

Obsah

1	Úvod	5
2	Základní pojmy z molekulární genetiky	7
2.1	DNA – kyselina z jádra buněk	7
2.2	Elektroforéza	9
2.3	Mikrosatelity	9
2.4	Polymerázová řetězová reakce (PCR)	11
2.5	Multiplex PCR	12
2.6	Genotypizace	13
3	Optimalizace primerů pro Multiplex PCR	14
3.1	Matice délkových kompatibilit	14
3.2	Matice chemických kompatibilit	15
3.3	Sestavování multiplexů	15
4	Chemická kompatibilita primerů	17
4.1	Výpočet parametrů ΔG_T° , ΔH° a ΔS° metodou NN	20
4.2	Závislost ΔG_T° , ΔH° a ΔS° na množství $[\text{Na}^+]$	21
4.3	Ukázka výpočtu termálních parametrů	22
5	Popis programového řešení – MultiPCR	25
5.1	Implementační požadavky	25
5.2	Knihovny a moduly	26
5.3	Řešení délkové kompatibility	28
5.4	Řešení chemické kompatibility	30
5.5	Sestavování multiplexů	34
5.6	Složitost algoritmu	35
6	Dosažené výsledky	39
7	Závěr	44
A	Příloha	46
A.1	Vytvoření seznamu všech překrytí	46
A.2	Obecná metoda výpočtu termálního parametru	47
A.3	Implementace datových tabulek	48

B	Výsledky testovaných sad	50
B.1	Test sady human_multiplex	50
B.2	Test sady str_database	52
	Literatura	55

Přehled nejdůležitějších pojmů

Pro účely pochopení dalších kapitol by bylo vhodné si shrnout a krátce popsat některé často používané výrazy či pojmy. Jejich bližší popis následuje v kapitole č. 2.

nukleotid základní stavební jednotka DNA (popř. RNA) – jednotlivé nukleotidy v DNA jsou: adenin, thymín, cytozin a guanin. RNA navíc obsahuje uracil.

DNA deoxyribonukleová kyselina – nukleová kyselina nesoucí genetickou strukturu buněčného organismu. Molekula DNA je tvořena dvěma spojenými vlákny, která jsou stočena do šroubovice.

ssDNA jednovláknová DNA – jedno z vláken DNA molekuly, které bylo odděleno např. denaturací

RNA ribonukleová kyselina – molekulární struktura, která slouží k přepisu části DNA na bílkovinu. Její struktura je velmi podobná DNA, obsahuje však některé odlišné nukleotidy.

PCR polymerázová řetězová reakce – metoda, při které dochází k replikaci molekul DNA

polymeráza pomocný enzym používaný při PCR, na jednoduchá vlákna DNA váže komplementární nukleotidy

amplikon produkt PCR – replikovaný úsek DNA

mikrosatelit jednoduchá sekvence opakujících se nukleotidů

gen základní jednotka dědičnosti nebo také úsek DNA nesoucí dědičnou informaci

alela forma genu – od jiné formy stejného genu se může lišit např. počtem opakování krátké sekvence nukleotidů (repeticí)

počet repetice počet opakování sekvence nukleotidů v mikrosatelitu

elektroforéza proces, který vizualizuje délkové zastoupení molekul DNA ve vzorku

báze viz nukleotid

bázová dvojice dvojice komplementárních nukleotidů spojená vodíkovým můstkem

bp zkr. *base pair* – délková jednotka DNA, která určuje počet bázových dvojic

chromozóm – organizovaná struktura DNA a bílkovin obsažená v buňčném jádře

lokus pozice na chromozómu

vodíkový můstek slabá molekulární vazba mezi nukleotidy, která bývá za některých okolností úmyslně přerušena – např. při replikaci DNA

molekulární marker molekulární struktura označující místo v DNA

duplex propojená dvojice komplementárních vláken

oligonukleotid krátké vlákno DNA zpravidla o délce několika jednotek až desítek nukleotidů

primer oligonukleotid používaný k započetí reakce při PCR

vlásenkování nežádoucí jev během PCR, kdy se část primeru spojí sama se sebou

hybridizace nežádoucí jev, při kterém se vzájemně spojí dva primery

multiplex sada skupin vzájemně kompatibilních mikrosatelitů, kde počet skupin odpovídá počtu elektroforézních kanálů

1 Úvod

Genotypizace je metoda identifikace druhu nebo jedince pomocí DNA sekvencí. Používá se v běžném lékařství a experimentální biologii, nebo také bývá hojně využívána v soudním lékařství např. při sporech o rodičovství, identifikaci podezřelého, nebo při hledání příbuzných obětí katastrof. Posledním detekčním krokem genotypizace je elektroforéza, během níž zkoumáme genetické otisky subjektů na speciálním gelu. Jednotlivé otisky jsou vyjádřeny délkami alel vybraných mikrosatelitů.

Vymezení cílů

Forenzní laboratoře mají obrovský detekční potenciál – mohou např. určit barvu očí pachatele trestného činu, od něhož mají k dispozici pouze malé množství genetického materiálu (zbytky slin, krev, vlas...). Takové testy se doposud prováděly jen ve výjimečných případech kvůli vysokým nákladům. Velkou část nákladů tvoří nákup elektroforézního gelu, na kterém se provádí určování délek alel mikrosatelitů.

Tato práce si klade za cíl návrh a implementaci sady originálních matematických metod založených na nejnovějších poznatcích molekulární biologie a bioinformatiky, které jsou schopny rozdělit zvolené mikrosatelity do skupin, ve kterých se délky jejich alel nebudou překrývat. V rámci těchto skupin musí být zároveň zajištěna vzájemná chemická kompatibilita primerů vázaných k mikrosatelitům, aby nedocházelo k jejich hybridizaci. Zkombinováním více mikrosatelitů do společné skupiny umožníme jejich současnou amplifikaci metodou PCR. Zároveň můžeme provádět pro každou skupinu detekci na elektroforézním gelu ve společném kanálu, aniž bychom omezili detekční schopnosti. Tím navýšíme celkovou propustnost a efektivitu genotypizace, urychlíme amplifikaci a omezíme spotřebu materiálu. Reálné zkrácení detekční doby významným způsobem přispívá např. ke včasnému odhalení pandemického šíření nebezpečných virů.

Nalezené metody jsou implementovány v nově vytvořené desktopové aplikaci MultiPCR. Důraz je kladen na přesnost výpočtu, podporu standardních formátů, rychlost, nezávislost na platformě a v neposlední řadě i na přehledné grafické rozhraní. Nasazení aplikace se předpokládá v genetických laboratořích forenzních expertů, kde bude sloužit jako nástroj pro efektivní snižování nákladů a zkracování detekční doby mikrosatelitů. Využití však může nalézt i u národních referenčních laboratoří při SZU, na Slovensku pak Štátné zdra-

votné ústavy. Aplikace je v provozu v Laboratoři experimentální medicíny při Dětské klinice LF UP a FN Olomouc.

MultiPCR disponuje metodami určování kompatibility primerů na základě výpočtu sil jejich intramolekulárních vazeb. Oproti konkurenčním programům (Primer3 [12], Autodimer [5], FastPCR [13]) navíc také umožňuje, díky unikátní metodě testování délkové kompatibility a heuristickému prohledávání stavového prostoru, sestavovat celé multiplexy z kompatibilních mikrosatelitů. Autoři Autodimeru tuto schopnost záměrně neimplementovali kvůli příliš vysoké časové složitosti výpočtu, jde totiž o NP-těžký problém: „*Due to relatively computationally intensive nature of the algorithm we decided not employ the screening algorithm upstream in the primer selection process.*” [5] Jak si později ukážeme, časovou složitost se nám povedlo dostat do přijatelných mezí díky pokročilým optimalizacím.

Hlavní cílovou skupinu této práce tvoří potenciální uživatelé MultiPCR, např. molekulární genetici, experti ve forenzní genetice, virologové a podobně. Tito uživatelé se zde seznámí s výpočetním pozadím aplikace, aby mohli výsledky správně interpretovat v rámci svého výzkumu. Do druhé cílové skupiny spadají bioinformatičtí, kteří se podílejí na vývoji obdobných biologicky zaměřených aplikací.

Na práci je možno nahlížet ze dvou pohledů – z biologického a z informatického. Snaha propojit tyto dvě značně metodologicky odlišné oblasti neustále naráží na bariéry v podobě nesourodého jazyka obou stran. Vzhledem k multioborovému průřezu tématem by tyto bariéry mohly být alespoň částečně narušeny. Bioinformatika je poměrně mladým oborem postrádajícím dostatečné množství multioborových pracovníků. Nezbyvá než doufat, že tato práce kromě svého hlavního účelu také přitáhne pozornost dalších studentů, kteří svými schopnostmi přispějí k dalšímu rozvoji bioinformatiky.

Struktura práce

V úvodu práce je čtenář nejprve seznámen se stručnými základy molekulární genetiky (kap. 2). Dále pak následuje matematické vyjádření kompatibility primerů (kap. 3 a 4). Kapitola 5 je věnována programové implementaci MultiPCR a zhodnocení časově-prostorové složitosti. V kapitole 6 jsou uvedeny provedené testy na skutečných datech a jejich srovnání s předpoklady. Přílohy obsahují dodatečné výsledky testů a kusy programového kódu, které nebylo pro jejich rozsah umístit do textu. Součástí příloh je také DVD, na kterém nalezneme aplikaci MultiPCR ve spustitelné verzi.

2 Základní pojmy z molekulární genetiky

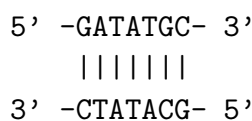
V této kapitole se podrobněji dozvíme, jak fungují některé biologické procesy. Popíšeme si DNA, její strukturu a způsoby replikace. Vysvětlíme si, co to jsou mikrosatelity a jaký mají vztah k této práci. Dozvíme se také o procesu zvaném genotypizace a vysvětlíme si, jak by šla zvýšit její propustnost pomocí multiplexování. Pokud čtenáři tato kapitola nestačí pro pochopení základních principů, může si znalosti prohloubit v [1].

2.1 DNA – kyselina z jádra buněk

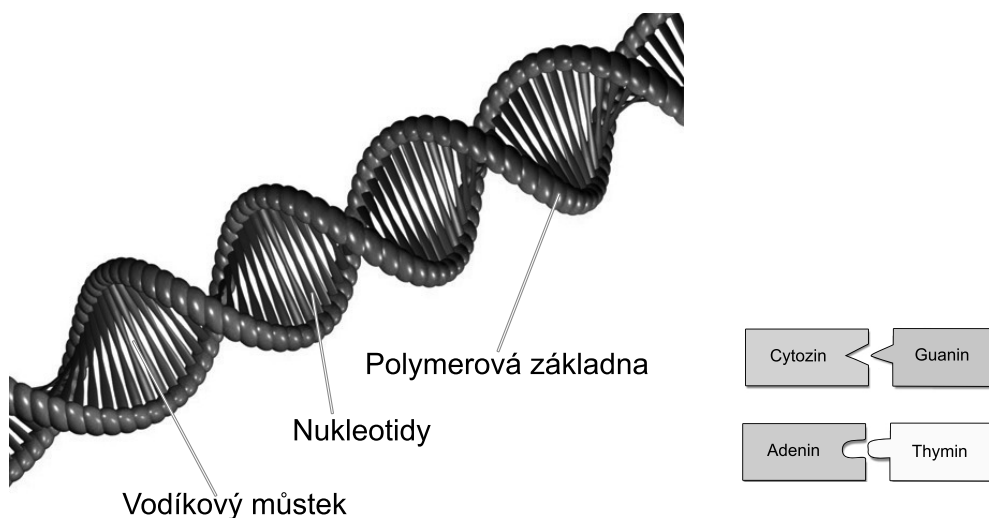
DNA je nukleová kyselina obsahující genetické informace o vývoji, stavbě a funkci živého organismu. Nukleová proto, že ji můžeme nalézt především v buněčném jádře. Jejím hlavním účelem je dlouhodobé a bezpečné uchovávání informací o daném živočichu, popřípadě viru. Hlavní kostru molekuly tvoří dvě polymerové základny (viz obr. 1) s navázanou sekvencí menších molekul – zvaných nukleotidy, které jsou nositeli genetické informace [14, 15]. Rozlišujeme čtyři různé nukleotidy: Adenin [A]¹, Cytosin [C], Thymin [T], Guanin [G].

Vzhledem ke zdvojené struktuře DNA musejí být zdvojeny i nukleotidy. Na jednom místě DNA se nacházejí vždy dva nukleotidy spojené tzv. vodíkovým můstkem. Říkáme, že jsou k sobě komplementární. K adeninu se váže pouze thymin: $A = T$, k cytosinu se váže pouze guanin: $C \equiv G$ – viz orientační obr. 2. Adenin je s thyminem spojen dvojitou vodíkovou vazbou, zatímco cytosin s guaninem vazbou trojitou. Trojitá vodíková vazba je pevnější, vyžaduje tak vyšší teploty na její přerušení. Komplementarita spolu se zdvojenou povahou DNA zajišťuje vysokou odolnost proti poškození informace. Chybějící nukleotid může být automaticky nahrazen samoopravnými mechanismy na základě znalosti jeho komplementu. Nukleotidy dělíme do dvou skupin. Purinové báze jsou A, T a pyrimidinové báze jsou C, G.

Abychom rozlišili jednotlivé konce DNA vláken v zápisech, označujeme je symboly 3' a 5' (čteme tři-konec a pět-konec). Vlákná jsou vůči sobě orientována antiparalelně, takže začíná-li jedno 5', pak na stejném místě druhé začíná 3' takto:



¹Při zápisech DNA sekvence používáme vždy první písmeno z názvu daného nukleotidu.



Obrázek 1: Zjednodušený model DNA molekuly.

Obrázek 2: Vazby nukleotidů.

Gen

Vyšší logickou strukturu DNA tvoří geny. Gen je posloupnost nukleotidů se speciálním významem. Na základě genové informace jsou nepřímo produkovány bílkoviny. Nepřímo proto, že se geny podílejí na tvorbě RNA, která řídí syntézu bílkovin. Geny zásadním způsobem ovlivňují vlastnosti jedince (barvu očí, fyzické dispozice) a jsou přenášeny i na potomky. Struktura genu se dělí na dvě oblasti – kódující a nekódující. Kódující oblasti určují, co daný gen ovlivňuje, zatímco nekódující oblasti určují, kdy je aktivován [16, 17]. Význam některých nekódujících oblastí ovšem nebyl doposud prokázán. Může se jednat buď o jakousi výplň kódu pro zajištění jeho bezpečnosti, nebo pozůstatek předchozího vývoje.

Alela

Jeden gen může mít více forem. Tyto formy nazýváme alely. Abychom mohli alely mezi sebou vzájemně porovnávat, musíme brát v potaz jejich umístění na lokusu. Jednotlivé formy genu se od sebe liší délkou repetice. Repetice je několikanásobné opakování krátké posloupnosti nukleotidů, nacházíme ji v mikrosatelitech (viz níže) [18, 19].

2.2 Elektroforéza

Elektroforéza je soubor analytických metod sloužících k separaci látek s odlišnou pohyblivostí ve stejnosměrném elektromagnetickém poli. V molekulární biologii a genetice využíváme elektroforézu k rozdělení DNA fragmentů podle jejich délky. Delší fragmenty mají větší hmotnost, menší pohyblivost a proto putují pomaleji. Pohyb probíhá od záporné elektrody ke kladné, neboť je polymerová základna DNA nabitá záporně.

Vzorky umísťujeme do speciálního gelu, jehož volba závisí na druhu zkoumaných vzorků. Pro separaci DNA se nejčastěji používá tzv. agarosový gel. Délku působení elektromagnetického pole a koncentraci gelu určujeme podle délky nejdelších fragmentů ve zkoumané směsi. Délka fragmentů bývá předem omezena pomocí restrikčních enzymů. Tyto enzymy přerušují DNA v určitém místě. Velikost elektrického napětí nám ovlivňuje rozlišovací schopnost experimentu. Vyšší napětí snižuje rozlišovací schopnost, ale zkracuje délku experimentu [20, 21].

Barviva

Agarosový gel je čirý a vzorek bývá také čirý. Aby byl výsledek pokusu patrný lidským okem, je třeba vzorek nějakým vhodným způsobem obarvit. Nejčastěji používanou barvou bývá ethidium bromid. Toto karcinogenní barvivo fialově fluoreskuje pod UV zářením. Přidáme-li barvivo na gel před započítím pokusu, můžeme sledovat postupný vývoj. Avšak přítomnost tohoto barviva negativně ovlivňuje přesnost experimentu. U vzorků, kde známe přibližnou velikost DNA fragmentů (a tedy i přibližnou dobu působení elektromagnetického pole) bývá výhodnější aplikovat barvivo až v závěru pokusu [26, 27].

2.3 Mikrosatelity

Mikrosatelity, nebo také STR's (Short Tandem Repeats) jsou krátké úseky nacházející se v DNA, které jsou složené z opakujícího se motivu nukleotidů. Délka opakujících se úseků bývá 2–4 báze dvojice, avšak může být i větší. Tyto úseky se většinou opakují 10 až 100×. V molekulární mikrobiologii jsou využívány jako markery. Existence konkrétní alely na určitém místě genomu jedince může naznačovat přítomnost specifické choroby. Mikrosatelity jsou obvykle označovány alfanumerickými kódy jako např. „D8S1179“. Mikrosatelit s opakovaným motivem AC a délkou repetice 8 zapíšeme $(AC)_8$.

Druhy mikrosatelitů

- Úplné – Opakující se sekvence není nijak narušena.
Např.: ACACACACACAC
- Neúplné – Opakující se motiv je přerušen jednou bází.
Např.: ACACATACACAC
- Složené – Jsou tvořeny dvěma a více po sobě jdoucími mikrosatelity s různými bázovými motivy.
Např.: ATCATCACACACCTCTCTCTCTCT

Mikrosatelity mohou být amplifikovány (namnoženy) metodou zvanou polymerázová řetězová reakce (zkr. PCR). K jejich amplifikaci je zapotřebí dvojice primerů, tzv. levý a pravý primer. Levý primer se váže na 3' konec mikrosatelitu, pravý primer na jeho 5' konec. Aby bylo možné určit identitu jedince pomocí některé z metod genotypizace, potřebujeme mít k dispozici dostatečné množství jeho genetického materiálu (resp. vybraných úseků DNA). Ne vždy soudní lékaři obdrží vzorek, který by zcela kvantitativně vyhovoval (např. vlas, zbytky kůže apod.) a právě proto je prováděna PCR. Touto reakcí jsou pak schopni i z několika málo molekul DNA namnožit požadované množství. Navíc neprobíhá množení molekul v celé jejich délce, replikují se pouze vybrané úseky o kterých je známo, že svou specificitou mohou jednotlivé jedince od sebe odlišit.

Jednou z důležitých informací o mikrosatelitu je jeho délka. Délku určíme součinem délky opakovaného motivu a počtu opakování. Např. mikrosatelit $(AGC)_{24}$ bude mít délku $3 \times 24 = 72$ bp. V praxi bude délka amplikonu (namnoženého mikrosatelitu) o něco větší, jelikož levý a pravý primer navazujeme na úsek k mikrosatelitu přilehlý, ne však vždy přímo sousedící. Skutečnou délku nám prozradí elektroforéza.

Vektor délek vybraných mikrosatelitů z genomu identifikuje jedince. Zjednodušeně řečeno - čím více jsou si dva vektory podobné, tím blíže jsou dvě osoby v příbuzenském vztahu. Čím větší máme počet vybraných mikrosatelitů, tím přesnější rozlišíme dva jedince od sebe. Větší vektory znamenají více provedených testů a tudíž vyšší náklady.

2.4 Polymerázová řetězová reakce (PCR)

Polymerázová řetězová reakce je metoda pro rychlé množení stejného úseku DNA. Duplikování probíhá ve třech fázích, které se periodicky opakují. Tato metoda využívá předem připravené primery (viz níže). Maximální délka duplikovaného úseku je přibližně 10 000 nukleotidů. Množství duplikovaného DNA roste exponenciálně s počtem provedených cyklů. Metodu lze uplatnit i na velmi malé vzorky obsahující pouze několik molekul DNA [28, 29].

Primer

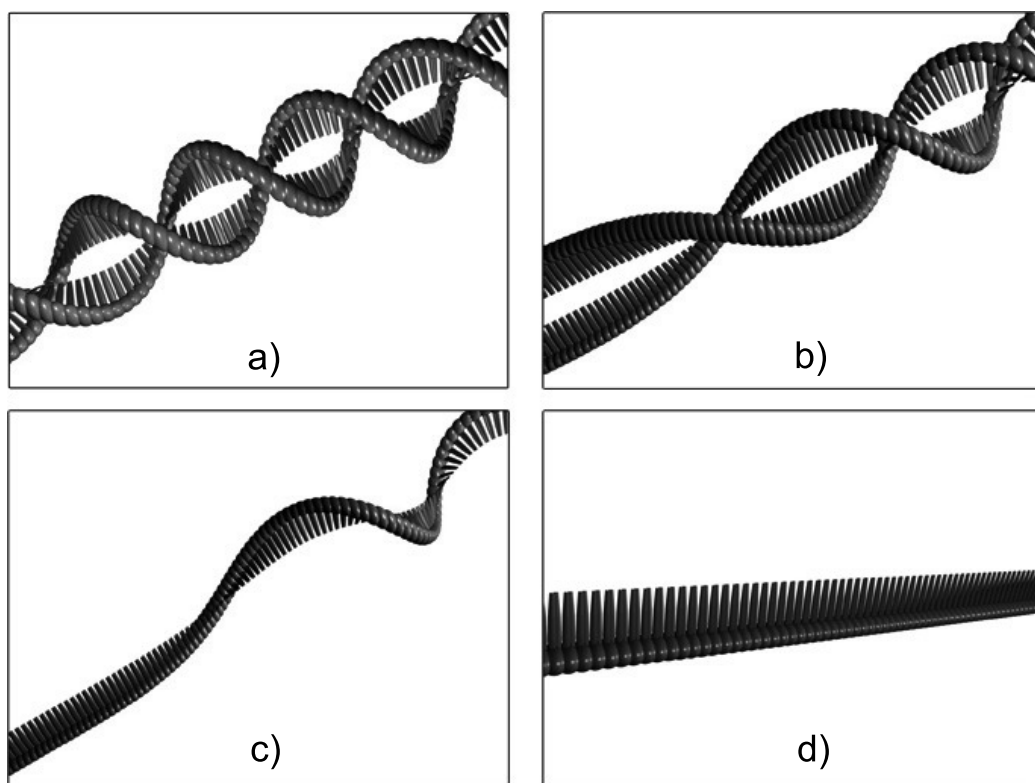
Primer je krátký, jednoduchý oligonukleotid, který slouží jako počáteční místo replikace DNA. Tento řetězec musí být komplementární k místu, kde chceme replikaci započít. Použití primerů je při replikování vyžadováno, jelikož enzymy vázající nukleotidy (tzv. polymeráza) jsou schopny přidat nové nukleotidy pouze k již existujícímu úseku DNA. V dnešní době bývá poměrně snadné získat libovolný primer. Pokud si nevybereme z tržní nabídky, můžeme si nechat primer vyrobit na zakázku ve specializovaných laboratořích [24, 25].

Jednotlivé fáze PCR

1. Denaturace – Po dobu 20–30 sekund působíme na DNA teplotou 94–98 °C. Působením tepla dojde k rozpadu vodíkových můstků dvoušroubovice. Vznikají nám tak dvě jednoduchá vlákna DNA. Viz obr. 3 a–d.
2. Nasednutí primerů (annealing) – Teplotu snížíme na 50–65 °C podle použitých primerů. Správně navržený primer by měl obsahovat vyvážený poměr $A = T$ a $C \equiv G$ párů, být dlouhý cca 18–24 párů a nevytvářet sekundární struktury. Sekundární struktura vznikne např. tím, že špatně navržený primer obsahuje vzájemně inverzní části, které se k sobě naváží.

Obvyklá teplota při nasedání primerů bývá 3–5 °C pod jejich teplotou tání (T_m). Při této teplotě se začnou primery obsažené ve směsi vázat na DNA vlákna, tzv. nasedat. Vznikají tak poměrně pevné vazby, které umožní v další fázi volným nukleotidům pomocí polymerázy pokračovat v rekonstrukci komplementárního vlákna.

3. Prodlužovací fáze – Teplota závisí na použité polymeráze. Na primery se postupně začínají vázat jednotlivé nukleotidy. Navazování pokračuje.



Obrázek 3: Postupná denaturace DNA v důsledku působení vysokých teplot.

čuje tak dlouho, dokud není opět zrekonstruována původní dvoušroubovice, nebo není-li vyčerpána polymeráza a nebo dokud nezačneme opět denaturovat. Dotvoří se nám tak chybějící komplementární část vlákn. Výsledkem je dvojnásobný počet řetězců DNA než byl v předchozím cyklu.

2.5 Multiplex PCR

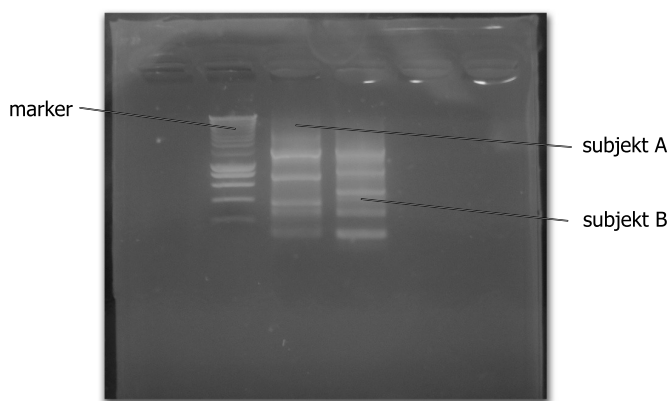
Existuje řada variant PCR. Nás bude pro účely této práce zajímat varianta označovaná jako Multiplex-PCR: Namísto jednoho mikrosatelitu jich množíme více naráz. Tím docílíme celkově vyšší propustnosti PCR a zároveň urychlíme proces genotypizace, který je na PCR závislý. S každým novým mikrosatelitem ve směsi nám ale přibývají dva nové primery. Tyto primery mohou za určitých okolností spolu reagovat – nasedat vzájemně na sebe (hybridizovat), popřípadě vlásenkovat. Další problém nastává při elektroforéze, která není schopna odlišit dva různé mikrosatelity identické délky. Všechny

tyto problémy je třeba eliminovat důkladnou přípravou před započítím experimentu a to tím, že do jedné sady testovaných subjektů vybíráme pouze kompatibilní multiplexy. Optimalizační algoritmus pro výběr vhodné sady nalezneme v kapitole 3.

2.6 Genotypizace

Souhrn genetických znaků daného jedince nazýváme genotyp. Metody a procesy, které vedou k určení genotypu nazýváme genotypizace. Genotypizaci používáme v experimentální biologii i v soudním lékařství. Genotypizační metody jsou schopny na základě porovnávání genetických vzorků určit např. druh živočicha, případně rozhodnout vzájemnou příbuznost srovnávaných jedinců (např. při sporech o mateřství/otcovství). Mezi prostředky určování genotypu patří elektroforézní zkoumání mikrosatelitů.

Na obr. č. 4 můžeme vidět výsledný produkt elektroforézy – fotografii agarosového gelu nasvíceného ultrafialovým světlem. Na tomto gelu byly využity tři tzv. kanály (sloupce, do kterých umísťujeme zkoumané vzorky). První kanál bývá zpravidla referenční – obsahuje marker, což je směs mikrosatelitů (resp. jejich alel) různých, avšak předem známých délek. Podle poloh jednotlivých alel v markeru můžeme určit délky alel v jiných kanálech na stejném řádku. Vzájemným porovnáním kanálů lze určit genotyp zkoumaného jedince.



Obrázek 4: Agarosový gel se třemi využitými kanály.

3 Optimalizace primerů pro Multiplex PCR

Prvním krokem při hledání kompatibilních multiplexů je sestavení matice délkových kompatibilit jednotlivých mikrosatelitů. Délková kompatibilita nám zajistí rozlišitelnost jednotlivých alel na elektroforézním gelu. Sestavení matice délkové kompatibility popisuje sekce 3.1. Druhým krokem je vytvoření matice chemické kompatibility. Chemická kompatibilita slouží k vyloučení nežádoucích interakcí mezi primery použitými pro namnožení mikrosatelitů. Vytvoření této matice je popsáno v sekci č. 3.2. V našem programu jsou k dispozici celkem dva moduly k určení chemické kompatibility. Autorem prvního je Dr. Ruslan Kalendar, druhý – standardní – vychází z článku prof. Santa-Lucii Jr. [3]. Popis kritérií pro určení chemické kompatibility „standardního“ modulu včetně používaných vztahů nalezneme v kapitole 4. Dalším krokem je sestavení všech možných kompatibilních multiplexů ze vstupní množiny mikrosatelitů. Poslední fází algoritmu je pak prohledávání pole kompatibilních multiplexů (které jsou uspořádány podle pořadí, v jakém byly uživatelem zadány) a seskupení požadované sady tak, aby se v ní žádný mikrosatelit neopakoval dvakrát. Sestavení multiplexů a vytvoření požadované sady je popsáno v sekci 3.3.

3.1 Matice délkových kompatibilit

Dva mikrosatelity jsou vůči sobě délkově kompatibilní tehdy, když neexistuje žádná alela prvního mikrosatelitu taková, která by měla shodnou délku s libovolnou alelou druhého mikrosatelitu. Délku určujeme vzhledem k pozicím primerů.

Matice délkových kompatibilit je sestavena následovně: její prvek na pozici (i, j) je nastaven na 1 pokud jsou mikrosatelity i a j vůči sobě délkově kompatibilní. Abychom docílili diagonální symetrie matice, nastavujeme zároveň prvek na pozici (j, i) také na hodnotu 1. Výsledná matice (pro názornost i s názvy mikrosatelitů) pak může vypadat např. takto:

	vWa	TPOX	TH01
vWa	0	1	1
TPOX	1	0	0
TH01	1	0	0

Tabulka 1: Ukázková matice délkové kompatibility.

3.2 Matice chemických kompatibilit

Ve druhém kroku algoritmu vytváříme matici chemických kompatibilit (pokud uživatel v programu zvolil testování chemické kompatibility). Tato matice je analogií výše uvedené matice délkových kompatibilit. Sestavíme ji následovně: její prvek na pozici (i, j) je nastaven na 1 pokud je levý primer mikrosatelitu i chemicky kompatibilní s levým i pravým primerem mikrosatelitu j a zároveň je pravý primer mikrosatelitu i chemicky kompatibilní s levým i pravým primerem mikrosatelitu j . Abychom docílili diagonální symetrie matice, nastavujeme zároveň prvek na pozici (j, i) také na hodnotu 1. Výsledná matice může vypadat např. takto:

	vWa	TPOX	TH01
vWa	0	0	1
TPOX	0	0	0
TH01	1	0	0

Tabulka 2: Ukázková matice chemické kompatibility.

Kombinací obou matic pomocí logické operace AND získáme matici celkové kompatibility. Tu sestavíme následovně: její prvek na pozici (i, j) je nastaven na 1 pokud platí $A(i, j) = 1$ a zároveň $B(i, j) = 1$. V opačném případě je nastaven na 0.

3.3 Sestavování multiplexů

Velikost multiplexů je určena jako podíl požadovaného počtu mikrosatelitů ve výsledné sadě a počtu elektroforézních kanálů. Pokud tento podíl není celočíselný, zaokrouhlí se nahoru s tím, že multiplexy pak mohou obsahovat i jistý počet „prázdných“ míst.

Příklad: Požadujeme-li 16 mikrosatelitů ve třech kanálech, program se pokusí sestavit multiplexy o velikosti 6-5-5 nebo 6-6-4 mikrosatelitů. V tomto kroku zpravidla dochází, kvůli požadavkům na vzájemnou kompatibilitu všech mikrosatelitů v multiplexu, k dramatické redukci počtu výsledných kompatibilních multiplexů oproti celkovému počtu možných kombinací (např. u výše zmíněného příkladu by šlo o kombinace všech 6-tic z 16 prvků, se započtením „prázdných“ míst pak z 18 prvků).

Vytvoření sady

Ve výše uvedeném příkladě potřebujeme vybrat celkem 3 multiplexy, a to tak, aby celkový počet mikrosatelitů v nich byl 16. Zde je použito standardní prohledávání do hloubky (depth-first search). Uživateli je předloženo první nalezené řešení a pokud s ním není spokojen, jsou postupně hledána řešení další. Pokud se nepodaří nalézt žádné řešení s požadovaným počtem mikrosatelitů, program se pokusí opakovat celý postup s menším výsledným počtem (v našem případě se začíná s 16 mikrosatelity, následně se postupuje na 15, 14...).

4 Chemická kompatibilita primerů

Molekulární polymerové struktury (DNA, primery, mikrosatelity) jsou stabilní jen při určitém rozsahu teplot, který vyplývá z jejich vnitřního uspořádání. Tohoto faktu si můžeme povšimnout i v běžném životě. Budeme-li zahřívat vaječný bílek, po určité době se nám působením tepla srazí. Ze stejného důvodu i lidské tělo obtížně zvládá působení teplot nad 42 °C.

V této sekci si určíme parametry, které popisují tepelné vlastnosti dvojic primerů v závislosti na jejich molekulární struktuře. Naším cílem bude nalezení takových podmínek, které nám naznačí, že by potenciální vazby mezi dvěma danými primery mohly být dostatečně slabé na to, aby nevznikly. O takových dvou primerech pak říkáme, že jsou kompatibilní.

U zadaných primerů předpokládáme, že jsou dobře navrženy. Neprovádí se tedy kontrola vlásenkování, protože počítáme s tím, že primery dodané uživatelem netvoří sekundární struktury. V programu nicméně tato možnost kontroly je a lze ji kdykoli aktivovat. Do budoucna je možno rovněž pracovat s návrhem primerů „de novo“.

Termodynamický systém

Při studiu termodynamických vlastností určitého systému rozlišujeme mezi systémem a jeho okolím. V našem případě jsou hranice okolí dány jednoznačně. Předmětem zkoumání je termodynamický systém nacházející se uvnitř PCR aparátu. Okolí představuje vše mimo aparát. Aparátem rozumíme nádobku se zkoumaným vzorkem a potřebnými chemikáliemi k provedení experimentu.

Pro nalezení rovnovážného stavu systému a popis probíhajících změn zavádíme veličiny ΔH° , ΔS° a ΔG_T° . Dosažením rovnovážného stavu dojde k uvolnění tepla ze systému do okolí. Tomuto stavu předchází vzájemné interakce jednoduchých vláken, které se na sebe váží a vytvářejí dvojítá vlákna. V izobarickém systému nazýváme změnu tepla nutnou pro dosažení rovnovážného stavu entalpie, značíme ji ΔH . V idealizovaném případě máme připravené jednotlivé primery v koncentraci 1 M. Pro označení ideálního případu používáme tzv. nulový symbol „°“. Změnu entalpie značíme ΔH° . Základní jednotkou je cal/mol.

Působením tepla dochází v našem systému ke změnám, které snižují jeho neuspořádanost. Je zjevné, že dimer vzniklý spojením dvou primerů vykazuje větší uspořádanost (tzn. menší neuspořádanost) než dva volně se pohybující

primery. Při amplifikaci DNA jsou na vlákna vázány jednotlivé nukleotidy, čímž se také zvyšuje uspořádanost systému. Míru neuspořádanosti systému vyjadřuje veličina zvaná entropie[6]. Změnu entropie značíme ΔS° a udáváme ji v jednotkách kcal/mol K (neboli počet kilo-kalorií na kelvin-mol).

S pomocí parametrů ΔH° a ΔS° můžeme vypočítat změnu gibbsovy volné energie. Změna gibbsovy volné energie odpovídá efektivní práci vykonané při reverzibilním procesu za konstantního tlaku a teploty[7]. Veličinu zapisujeme ΔG_T° . Teplota T je udávána v Kelvinech. Vztah mezi ΔG_T° , ΔH° a ΔS° definuje rovnice č. (1). Odvozením a převodem na společné jednotky získáme požadovaný vztah č. (2).

$$G_T = H - T \times S \quad (1)$$

$$\Delta G_T^\circ = \frac{\Delta H^\circ \times 1000 - T \times \Delta S^\circ}{1000} \quad (2)$$

Pokud jsme schopni zjistit ΔH° a ΔS° , pak můžeme takto vypočítat ΔG° při libovolné teplotě T . Následující rovnice vyjadřuje vztah mezi rovnovážnou konstantou K při teplotě T a změnou gibbsovy volné energie ΔG_T° kde R je plynová konstanta odpovídající 1,9872 cal/mol K:

$$\Delta G_T^\circ = -RT \times \ln K \quad (3)$$

Teplota tání T_m

Teplota tání T_m je definována jako teplota, při které je polovina vláken ve stavu tzv. statistického klubka (random coil) a druhá polovina vláken v duplexním stavu[3, 8]. Vyjádříme-li si T z rovnic (2) a (3), dostaneme následující vztah:

$$T = \frac{\Delta H^\circ \times 1000}{\Delta S^\circ - R \ln K} \quad (4)$$

Vycházíme-li z předpokladu, že koncentrace jednotlivých vláken (primerů) ve vzorku jsou shodné, pak pro rovnovážnou konstantu K platí $K = Ct/4$. Odvození rovnovážné konstanty K můžeme nalézt v [8], odstavec „Computation of T_m using ΔH° and ΔS° .“ Ct odpovídá koncentraci vláken v mol. Dosazením hodnoty K do rovnice (4) získáme:

$$T_m = \frac{\Delta H^\circ \times 1000}{\Delta S^\circ + R \ln \frac{Ct}{4}}. \quad (5)$$

Pokud by byla vlákna vůči sobě komplementární, pak $K = Ct$ a vztah pro výpočet T_m vypadá takto:

$$T_m = \frac{\Delta H^\circ \times 1000}{\Delta S^\circ + R \ln Ct}. \quad (6)$$

Výše uvedené vztahy nalezneme podrobněji popsané v [3, 8].

Podmínky kompatibility primerů

Dva primery budeme považovat za kompatibilní tehdy, když se nebudou na sebe při dané teplotě vázat. Teplotu, při které potenciální vazby zanikají (popř. nemohou vůbec vznikat) udává právě T_m . Při výpočtech si zavedeme prahovou hodnotu T_m^* , což může být např. teplota, při které bude probíhat v PCR nasedání primerů. Tuto teplotu si bude volit uživatel podle svých potřeb. Je třeba mít na paměti, že některé primery spolu nebudou tvořit duplexy při žádné přijatelné teplotě (např. když $T_m \leq 0$). Z matematického pohledu vypadá podmínka kompatibility vzhledem k T_m takto:

$$T_m \in (-\infty; 0) \cup (T_m^*; \infty). \quad (7)$$

Druhá podmínka nám bude vymezovat maximální změnu gibbsovy volné energie ΔG_T° . Opět si zavedeme uživatelsky nastavitelnou prahovou veličinu $\Delta G_T^{\circ*}$. Tentokrát požadujeme, aby byla energie ve dvojici primerů vyšší, než uživatelem nastavený práh. Podmínku zapíšeme takto:

$$\Delta G_T^\circ > \Delta G_T^{\circ*}. \quad (8)$$

Podmínky (7) a (8) využijeme v našem programu. Abychom mohli počítat ΔG_T° a T_m , potřebujeme znát termální parametry ΔH° a ΔS° . Hodnota těchto parametrů závisí na molekulární struktuře dvou zkoumaných primerů. Výpočet si ukážeme v následující sekci.

4.1 Výpočet parametrů ΔG_T° , ΔH° a ΔS° metodou NN

Metoda zvaná nearest-neighbour (zkr. NN), neboli metoda „nejbližší soused“ využívá empirických poznatků, které byly získány pozorováním chování krátkých úseků DNA za různých podmínek. Následující klasické výpočty známe díky výzkumu prof. SantaLucia, Jr. Jeho výzkum byl prezentován v [3]. Sjednocením termodynamických studií ze šesti různých laboratoří byly odvozeny vztahy pro výpočet ΔG_{37}° , ΔH° a ΔS° . Tyto vztahy byly zároveň empiricky dokázány.

Výpočetní model „nejbližší soused“ pro DNA vychází z předpokladu, že stabilita dané báze dvojice závisí na identitě a orientaci sousedních bázevých dvojic. Uvažujeme-li duplexy o délce 2 bp, počet vzájemných povolených kombinací každý-s-každým je 16. Některé z kombinací jsou totožné, proto celkový počet jedinečných kombinací je 10. Jednotlivé kombinace budeme zapisovat ve formátu TC/AG, což znamená, že 5'-TC-3' párujeme s 3'-AT-5'. Párování tedy probíhá antiparalelně, stejně jako tomu je u DNA. Celkové ΔG_{37}° počítáme takto:

Sekvence	ΔH° kcal/mol	ΔS° cal/mol K	ΔG_{37}° (NN) kcal/mol
AA/TT	-7,90	-22,20	-1,00
AT/TA	-7,20	-20,40	-0,88
TA/AT	-7,20	-21,30	-0,58
CA/GT	-8,50	-22,70	-1,45
GT/CA	-8,40	-22,40	-1,44
CT/GA	-7,80	-21,00	-1,28
GA/CT	-8,20	-22,20	-1,30
CG/GC	-10,60	-27,20	-2,17
GC/CG	-9,80	-24,40	-2,24
GG/CC	-8,00	-19,90	-1,84
<i>inic</i> (C · G)	0,10	-2,80	0,98
<i>inic</i> (A · T)	2,30	4,10	1,03
Symetrická korekce	0	-1,4	0,43

Tabulka 3: Unifikované oligonukleotidové parametry v 1M NaCl.

$$\Delta G_{37}^{\circ} = \Delta G_{inic}^{\circ}(\text{term}) + \sum_i n_i \Delta G_{37}^{\circ}(i) + \Delta G_{\text{sym}}^{\circ}, \quad (9)$$

kde jsou hodnoty nezávislé na sekvenci shrnuty v inicializačním parametru $\Delta G_{inic}^{\circ}(A \cdot T)$ případně $\Delta G_{inic}^{\circ}(C \cdot G)$ podle toho, začíná-li sekvence terminály $A \cdot T$ nebo $C \cdot G$. Hodnota $\sum_i n_i$ odpovídá počtu výskytů dané sekvence (např. pro $i = 1$ uvažujeme sekvenci AA/TT). Jednotlivé sekvence včetně odpovídajících hodnot inicializačních parametrů ΔG_{inic}° a $\Delta G_{37}^{\circ}(\text{term})$ nalezneme v tab. č. 3. Dále započítáváme entropickou korekci $\Delta G_{\text{sym}}^{\circ} = +0,43 \text{ kcal/mol}$ pro komplementární vlákna[9]. Pro nekompl. vlákna počítáme $\Delta G_{\text{sym}}^{\circ} = 0$.

V mnoha případech budeme počítat ΔG° při jiné teplotě než 37°C , ačkoliv je uvedený výpočet pro teplotu 37°C nejpřesnější. K tomu nám slouží hodnoty $\Delta H^{\circ}(i)$ a $\Delta S^{\circ}(i)$ uvedené opět v tab. č. 3. Celkové ΔH° a ΔS° počítáme:

$$\Delta H^{\circ} = \Delta H_{inic}^{\circ}(\text{term}) + \sum_i n_i \Delta H^{\circ}(i) + \Delta H_{\text{sym}}^{\circ}, \quad (10)$$

$$\Delta S^{\circ} = \Delta S_{inic}^{\circ}(\text{term}) + \sum_i n_i \Delta S^{\circ}(i) + \Delta S_{\text{sym}}^{\circ}. \quad (11)$$

Dosazením ΔH° a ΔS° do rovnice č. (2) získáme požadovaný vztah pro výpočet ΔG° při libovolné teplotě T . Pokud bychom potřebovali velkou extrapolaci teploty od původních 37°C , bylo by zapotřebí započítávat tepelnou kapacitu[11].

4.2 Závislost ΔG_T° , ΔH° a ΔS° na množství $[\text{Na}^+]$

Parametry ΔG° , ΔH° a ΔS° jsou ovlivňovány množstvím monovalentní soli Na^+ ve směsi. Abychom tuto závislost zahrnuli do výpočtu, sestavíme korekční formule[3]. Korekce pro $[\text{Na}^+]$ nezávisí na prvcích sekvence, závisí však na její délce[10]. Uvažovaná závislost má logaritmický charakter. Množství Na^+ počítáme v jednotkách Mol. Délkový parametr N odpovídá počtu báзовých dvojic duplexu. Do tohoto počtu nezapočítáváme terminální dvojici. Např. pro duplex o délce 20 bp bude $N = 18$.

$$\Delta G_{37}^{\circ}([\text{Na}^+]) = \Delta G_{37}^{\circ} - 0,114 \times N \times \ln [\text{Na}^+], \quad (12)$$

$$\Delta S^\circ([\text{Na}^+]) = \Delta S^\circ + 0,368 \times N \times \ln [\text{Na}^+], \quad (13)$$

$$\Delta H^\circ([\text{Na}^+]) = \Delta G^\circ([\text{Na}^+]) + \frac{310,15 \times \Delta S^\circ([\text{Na}^+])}{1000}. \quad (14)$$

Dosazením $\Delta G_{37}^\circ([\text{Na}^+])$ a $\Delta S^\circ([\text{Na}^+])$ do rovnice č. (2) získáme celkovou hodnotu $\Delta G_T^\circ([\text{Na}^+])$, kterou můžeme přímo použít k rozhodnutí první podmínky termální kompatibility:

$$\Delta G_T^\circ([\text{Na}^+]) = \Delta G_{37}^\circ([\text{Na}^+]) + \frac{(310,15 - T) \times \Delta S^\circ([\text{Na}^+])}{1000}. \quad (15)$$

Analogicky získáme dosazením $\Delta H^\circ([\text{Na}^+])$ a $\Delta S^\circ([\text{Na}^+])$ do rovnice č. (4) hodnotu $T_m([\text{Na}^+])$ určenou k rozhodnutí druhé podmínky termální kompatibility:

$$T_m([\text{Na}^+]) = \frac{\Delta H^\circ([\text{Na}^+]) \times 1000}{\Delta S^\circ([\text{Na}^+]) + R \ln K}. \quad (16)$$

Dodatečné korekce

Výše uvedené vztahy nám vyjadřují termální parametry duplexu v případech, kdy jsou jednotlivé nukleotidy v bázeové dvojici vůči sobě komplementární. V našem případě by byl ideální duplex takový, kde není vůči sobě žádný nukleotid komplementární a to proto, že je zde minimální šance na vznik pevných vazeb. Nekomplementarita nám ovlivňuje sílu vazby a proto s ní musíme počítat. Vzhledem k tomu, že by úplný výpočet vazebné korekce při nekomplementaritě přesahoval rozsah této práce, zvolíme stejný (konzervativní) přístup jako [5]. K hodnotě $\Delta G_T^\circ([\text{Na}^+])$ budeme započítávat +0,438 cal/Mol za každou neshodu. Hodnota M odpovídá počtu neshod:

$$\Delta G_T^\circ([\text{Na}^+], M) = \Delta G_T^\circ([\text{Na}^+]) + 0,438 \times M. \quad (17)$$

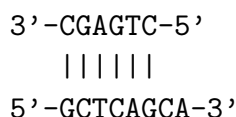
4.3 Ukázka výpočtu termálních parametrů

Příklad: Určete termální parametry primerů GCTCAGCA a CGAGTC při $[\text{Na}^+] = 0,085 \text{ mol}$ a teplotě $T = 45^\circ \text{C}$. Koncentraci vláken uvažujte $1 \mu\text{mol}$.

Postup řešení: Nejprve přiložíme zadané primery k sobě (podle obr. č. 5) a pomocí tab. č. 3 s využitím vzorce č. (9) spočítáme ΔG_{37}° .

Postupujeme takto: Na začátku zkoumáme první sousední dvojici (první dva bázevé páry). První dvojice je CG/GC . Nalezneme odpovídající hodnotu v tabulce. Druhou dvojicí bude GA/CT . Pro ní máme v tabulce také odpovídající hodnotu. Třetí dvojice bude AG/TC . Tato dvojice v tabulce není obsažena, avšak můžeme nalézt její ekvivalent CT/GA (psaný zrcadlově). Dále pokračujeme obdobně i při výpočtu ΔS° .

Obrázek 5: Ukázka přiložení primerů



$$\Delta G_{37}^\circ = \text{inic}(C \cdot G) + \Delta G_{37}^\circ(CG/GC) + \Delta G_{37}^\circ(GA/CT) + \Delta G_{37}^\circ(CT/GA) + \Delta G_{37}^\circ(GT/CA) + \Delta G_{37}^\circ(GA/CT) + \Delta G_{\text{sym}}^\circ$$

$$\Delta G_{37}^\circ = 0,98 - 2,17 - 1,30 - 1,28 - 1,44 - 1,30 + 0 = -6,51 \text{ kcal/mol}$$

$$\Delta S^\circ = \text{inic}(C \cdot G) + \Delta S^\circ(CG/GC) + \Delta S^\circ(GA/CT) + \Delta S^\circ(CT/GA) + \Delta S^\circ(GT/CA) + \Delta S^\circ(GA/CT) + \Delta S_{\text{sym}}^\circ$$

$$\Delta S^\circ = -2,8 - 27,2 - 22,2 - 21,0 - 22,4 - 22,2 + 0 = -117,8 \text{ cal/mol K}$$

Dále aplikujeme odpovídající korekce. Postupně si vypočteme ΔS° , ΔG_{37}° a ΔH° závislé na množství $[\text{Na}^+]$.

$$\Delta S^\circ([\text{Na}^+]) = \Delta S^\circ + 0,368 \times N \times \ln [Na^+]$$

$$\Delta S^\circ(0,085) = -117,8 + 0,368 \times 5 \times -2,47 = -122,34 \text{ cal/mol K}$$

$$\Delta G_{37}^\circ([\text{Na}^+]) = \Delta G_{37}^\circ - 0,114 \times N \times \ln [Na^+]$$

$$\Delta G_{37}^\circ(0,085) = -6,51 - 0,114 \times 5 \times -2,47 = -5,10 \text{ kcal/mol}$$

$$\Delta H^\circ([\text{Na}^+]) = \Delta G_{37}^\circ([\text{Na}^+]) + \frac{(310,15 \times \Delta S^\circ([\text{Na}^+]))}{1000}$$

$$\Delta H^\circ(0,085) = -5,10 + \frac{(310,15 \times -122,34)}{1000} = -43,04 \text{ kcal/mol}$$

Nyní vypočteme termální parametry ΔG_T° a ΔG_{45}° při zadaném množství $[\text{Na}^+]$, které lze rovnou využít v podmínce kompatibility.

$$\Delta G_T^\circ([\text{Na}^+]) = \Delta G_{37}^\circ([\text{Na}^+]) + \frac{(310,15 - T) \times \Delta S^\circ([\text{Na}^+])}{1000}$$

$$\Delta G_{45}^\circ(0,085) = -5,10 + \frac{(310,15 - [45 + 273,15]) \times -122,34}{1000} = -4,12 \text{ kcal/mol}$$

$$T_m([\text{Na}^+]) = \frac{\Delta H^\circ([\text{Na}^+]) \times 1000}{\Delta S^\circ([\text{Na}^+]) + R \ln \frac{C_t}{4}}$$

$$T_m(0,085) = \frac{-43,04 \times 1000}{-122,34 + 1,987 \times -15,20} = 282,15 \text{ K} = 9^\circ \text{ C}$$

5 Popis programového řešení – MultiPCR

Manuální provádění výpočtů z předchozí kapitoly by bylo neúměrně náročnou prací. Např. pro určení kompatibility dvou primerů o délce 24 nukleotidů potřebujeme celý výpočet provést $94 \times$. Výpočet kompatibility dvou mikrosatelitů s těmito primery bude dokonce ještě čtyřikrát náročnější. Vytvoření specializovaného software bylo v tomto případě nevyhnutelné a vyústilo ve vytvoření komplexní uživatelské aplikace v Javě o celkové velikosti kódu cca. 9 000 řádků (258 kB). Nejprve si shrneme požadavky, které byly na tento program kladeny.

5.1 Implementační požadavky

Primárním požadavkem návrhu byla nezávislost na použité platformě. Množství vědeckého software je psáno pro Linux, laboratoře bývají vybaveny počítači Apple Macintosh s Mac OS. Vědecké počítače s MS Windows také nejsou výjimkou. Obsazení přinejmenším těchto tří nejpoužívanějších platform bylo tedy klíčové. Další požadavek spočíval v modularitě – tedy možnosti snadného rozšíření aplikace o další vyhledávací algoritmy. Vývoj v oblasti návrhu multiplexů jde neadržitelně kupředu, proto může být sebelepší algoritmus kompatibility po čase překonán novým algoritmem, který reflektuje nejnovější poznatky z oblasti. Dále bylo nutné zajistit interkompatibilitu mezi podobnými programy použitím standardních, neproprietárních vstupně-výstupních formátů. Důvodem je fakt, že vstupem našeho programu může být výstup jiného, obdobného programu a naopak výstup našeho programu může být dále zpracováván dalším programem. I přesto však bude rozumné oddělit grafické rozhraní aplikace od jejího výpočetního jádra. Za výchozí jazyk byla zvolena angličtina. Implementační parametry aplikace, které splňují výše uvedené požadavky jsou shrnuty v tabulce č. 4.

Programovací jazyk	Java 5
Řešení modularity	implementací interface
Vstupní formáty	FASTA (data) XML (uložené výsledky)
Výstupní formáty	XML PDF

Tabulka 4: Implementační parametry MultiPCR.

5.2 Knihovny a moduly

V této sekci si popíšeme používané externí knihovny funkcí a ukážeme si, na jaké jednotlivé moduly je aplikace rozdělena. Za knihovny funkcí budeme považovat samostatné balíčky dodané do aplikace, které nezávisí na žádné z jejích součástí. Moduly jsou naše vlastní funkční celky. Na obr. č. 6 vidíme ukázkou uživatelského rozhraní (modul GUI).

Seznam použitých knihoven funkcí

BioJava – Balík BioJava umožňuje efektivní práci s mikromolekulárními strukturami v podobě řetězců. Optimalizované algoritmy prověřené léty zajišťují dostatečný výkon. Užitečné nástroje umožňují snadnou manipulaci s řetězci, přičemž minimalizují využití paměti a procesorového času použitím tzv. „pohledů“. Např. namísto zrcadlení dlouhého řetězce je vytvořen pohled, který obsahuje původní řetězec a informaci o směru zpracování.

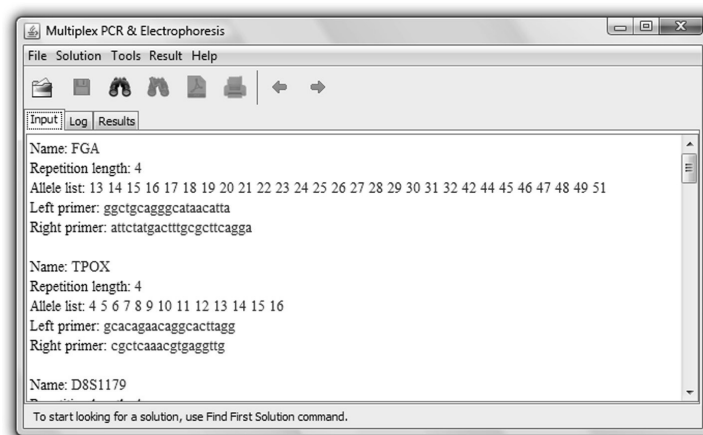
iText/PDF – iText je volně šiřitelná knihovna pro výstupy ve formátu PDF. V programu je využívána ke generování výsledků v tisknutelné podobě. Tisková sestava obsahuje grafické znázornění všech použitých kanálů a v nich umístění jednotlivých alel mikrosatelitů seřazených podle své délky. Autorem tiskových sestav je David Péres z Madridu.

Java Help System – Knihovna JHS slouží ke snadnému integrování nápovědy do aplikace. Využíváme jí především kvůli podpoře standardních formátů. Nápovědu můžeme tvořit ve formátu HTML, obsah ve formátu XML. Rozšiřování a údržbu nápovědy můžeme díky JHS provádět bez nutnosti měnit aplikační kód.

JUnit – JUnit je knihovnou pro sestavování automatizovaných aplikačních testů. Při sestavování nového algoritmu si nejprve vytyčíme výsledky, jakých chceme dosáhnout. Poté sestavíme test obsahující tyto výsledky. Algoritmus považujeme za úspěšný až tehdy, když projde naším testem.

Aplikační moduly

GUI – Modul GUI obstarává vše, co se týká vizuální komunikace s uživatelem. V tomto modulu jsou obsaženy jednotlivé dialogy a okna, vazby mezi akcemi uživatele a voláním metod výpočetního jádra. Je zde také



Obrázek 6: Ukázka hlavního okna aplikace s načteným vstupním souborem.

implementována HTML konzole, která je využívána jak pro zobrazení načteného seznamu mikrosatelitů, tak i pro logování a zobrazování nalezených výsledků. Jazyk HTML jsme zvolili proto, že umožňuje přehledně znázorňovat požadované informace. Stejně informace jako do konzole v záložce Log putují na standardní výstup. Pro lepší čitelnost jsou na standardním výstupu odfiltrovány HTML značky.

Chem – Modul Chem obsahuje komponenty výpočetního jádra zodpovědné za výpočet chemické kompatibility. V současné době modul Chem obsahuje komponentu pro původní výpočet od Dr. Ruslana Kalendara a naši vlastní komponentu pro výpočet pomocí termálních parametrů. Jsou zde také rozhraní *DimerTools* a *DimerData*. Implementováním *DimerTools* můžeme do programu přidat nový výpočetní algoritmus. Implementací rozhraní *DimerData* získáme datovou schránku, obsahující vstupní parametry nového algoritmu – tyto vstupní parametry se mohou od současných zcela diametrálně lišit. Není vyloučeno např. ani použití neuronové sítě či napojení na expertní systém.

Msat – Modul Msat obsahuje metody a datové struktury nutné pro práci s mikrosatelity. Jsou zde třídy definující primer, primerový pár nebo třeba amplikon. Metody modulu Msat umožňují načtení vstupu ve formátu FASTA. Je zde také obsažena komponenta toPDF sloužící k exportu výsledků ve formátu PDF. Také jsou tu implementovány datové třídy uchovávající seznam multiplexů v úsporném formátu.

Solver – Modul solver řídí úkony související s vyhledáváním řešení. Podle potřeby úkoluje příslušnou komponentu modulu Chem. Je také zodpovědný za výpočet délkové kompatibility. Pokrok ve výpočtu pak sděluje komponentě GUI, která odpovídajícím způsobem aktualizuje uživatelské prostředí. Tento modul je zároveň vhodný kandidát pro nový vstupní bod aplikace – v případě potřeby je možné drobnými úpravami kódu docílit podpory pro práci v příkazové řádce.

5.3 Řešení délkové kompatibility

Algoritmus sestavení matice délkové kompatibility nalezneme na začátku kapitoly 3 na straně 14. V této sekci si předvedeme způsob, jakým je uvedený algoritmus realizován programově.

Vstup:

Vstupem algoritmu je pole N mikrosatelitů (resp. amplikonů) a povolený délkový rozsah délek alel.

Výstup:

Výstupem algoritmu je matice délkové kompatibility o rozměrech $N \times N$.

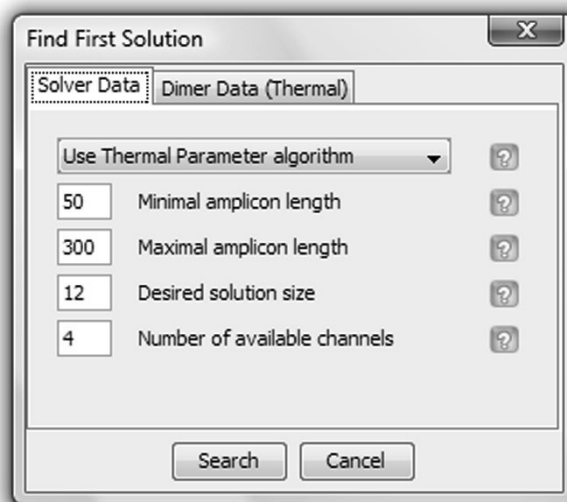
Postup řešení:

```
1 int i = -1;
2 for (STRamplicon amplicon: this) {
3     bitSets[++i] = new BitSet(maxLen-minLen+1);
4     for (float reps: amplicon.repList) {
5         bitSets[i].set(amplicon.amplLength(reps) - minLen);
6     }
7 }
8 for (i = 0; i < size(); i++) {
9     for (int j = i+1; j < size(); j++) {
10         if (! bitSets[i].intersects(bitSets[j])) {
11             matrix.set(i, j);
12             matrix.set(j, i);
13         }
14     }
15 }
```

Na řádce 1–7 procházíme všechny amplikony ze seznamu. Pro každý ampikon založíme bitové pole o velikosti jeho nejdelší alely (ř. 3) a projdeme délky všech jeho alel. Každou z délek poznamenejeme na příslušném místě bitového pole (ř. 5). Pozn.: Z paměťově-optimalizačních důvodů vytváříme bitové pole kratší o délku nejkratší alely a o tuto délku pak snižujeme indexy.

Řádky 8–15 slouží k porovnání všech bitových polí stylem „každý-s-každým“, přičemž kontrolujeme, zda se nepřekrývají některé bity v bitovém poli. Je-li vše v pořádku, nastavíme matice „matrix“ na pozici (i, j) a (j, i) na hodnotu 1.

Jednou z optimalizací, kterou provádíme ještě před započítáním výpočtu je odfiltrování všech ampikonů ze vstupního pole, které nesplňují zadané délky. Rozsah povolených délek určujeme v základním dialogu pro hledání řešení na záložce „Solver Data“. Dialog otevřeme buď klepnutím na ikonu , nebo stiskem klávesy F7. Rozsah délek máme označen jako „Minimal amplicon length“ a „Maximal amplicon length“. V tomto dialogu si také volíme použitý algoritmus chemické kompatibility. Podle volby algoritmu se přizpůsobí obsah na záložce „Dimer Data“. Parametry „Desired solution size“ a „Number of available channels“ využíváme až při sestavování multiplexů. Náhled dialogového okna můžeme vidět na obr. č. 7.



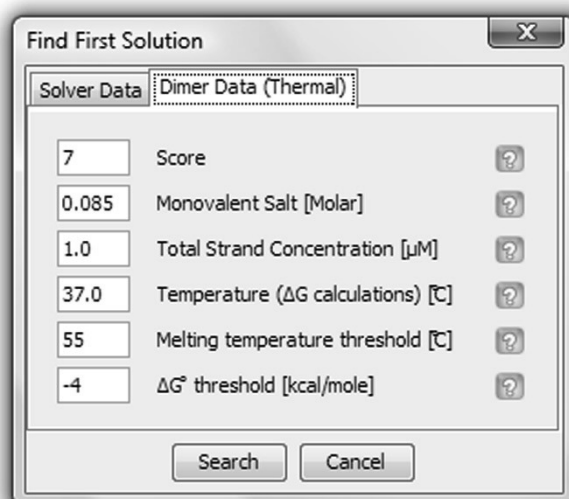
Obrázek 7: Ukázka dialogu pro hledání řešení.

5.4 Řešení chemické kompatibility

Výpočet chemické kompatibility dvou primerů pomocí termálních parametrů nalezneme v kapitole 4. Zde si popíšeme programový postup pro řešení kompatibility dvou mikrosatelitů, z nichž každý obsahuje dvojici primerů.

Vstup:

Vstupem algoritmu jsou dva vyšetřované mikrosatelity. Mezi nastavitelné parametry patří: Hodnota „Score“, množství monovalentní soli $[\text{Na}^+]$, koncentrace vláken, teplota T a dvě prahové hodnoty T_m^* a ΔG_T^* , které jsou popsány na straně 19. Dialog zadání výpočetních parametrů můžeme vidět na obr. č. 8.



Obrázek 8: Ukázka dialogu s parametry pro řešení chemické kompatibility.

Výstup:

Výstupem algoritmu je logická hodnota *true* nebo *false* podle toho, zda jsou dva zadané mikrosatelity kompatibilní.

5.4.1 Postup řešení kompatibility mikrosatelitů

Dotaz na řešení kompatibility mikrosatelitů provádíme zavoláním metody:

```
public boolean isCompatible(PrimerPair p1, PrimerPair p2);
```

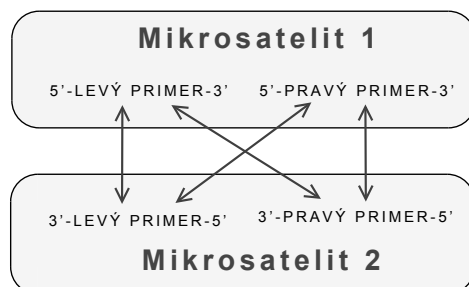
Tato metoda rozhoduje o tom, která porovnání kompatibilit jednotlivých primerů budou provedena. Primery si označíme *pp1.L*, *pp1.R*, *pp2.L* a *pp2.R* podle toho, který (pp) primerový pár uvažujeme a také podle toho, jestli se jedná o (L) levý nebo (R) pravý primer. Dotaz na řešení kompatibility primerů provedeme zavoláním:

```
public boolean isCompatible(Primer p1, Primer p2);
```

Pozn.: Provádíme zde přetěžování metody *isCompatible* – o tom, zda bude použita metoda kompatibility mikrosatelitů nebo primerů rozhoduje datový typ vstupních proměnných.

Podle schématu na obr. č. 9 sestavíme metodu pro mikrosatelity takto:

```
1 public boolean isCompatible(PrimerPair pp1, PrimerPair pp2) {
2     boolean cond;
3     cond =
4         isCompatible(pp1.L, pp2.L) &&
5         isCompatible(pp1.L, pp2.R) &&
6         isCompatible(pp1.R, pp2.L) &&
7         isCompatible(pp1.R, pp2.R);
8     return cond;
9 }
```



Obrázek 9: Schéma vzájemného porovnávání primerů.

5.4.2 Postup řešení kompatibility primerů

Postup řešení kompatibility primerů si rozdělíme do tří kroků: V prvním kroku nalezneme všechna možná vzájemná překrytí dvou primerů. Ve druhém kroku spočteme pro každé jednotlivé překrytí termální parametry podle popisu ze sekce 4.1 a 4.2. Ve třetím kroku rozhodneme podmínky kompatibility a vrátíme výsledek.

Krok 1 – nalezení všech překrytí dvou primerů

Nejprve si připravíme datovou třídu, do které budeme ukládat všechna použitelná překrytí. Tuto třídu nazvěme *AlignmentResult*. Bude obsahovat údaj o posunutí (pos), počet kompl. nukleotidů (matches), počet nekompl. nukleotidů (mismatches) a řetězec překrytí (mString). Řetězec překrytí obsahuje v kompl. místě písmeno označující nukleotid, v nekompl. místě symbol 'N'.

Vzhledem k velkému počtu všech možných překrytí bylo zapotřebí zavést klasifikátor, který nám vyloučí některé snadno rozhodnutelné případy. Např. šance na hybridizaci dvou primerů, které nejsou vůči sobě v daném překrytí vůbec komplementární (tzn. že nemají žádný komplementární nukleotid) je minimální. Příznakem pro náš klasifikátor bude hodnota skóre. Tu vypočítáme jako rozdíl počtu kompl. a nekompl. nukleotidů. Práh si volí uživatel na záložce DimerData (řádek Score). Nižší skóre nám zpřisňuje požadavky, ale může prodloužit dobu výpočtu. Klasifikaci podle skóre používáme ve stejném smyslu jako autoři softwaru Autodimer. Ti při svých experimentech došli k závěru, že prahové skóre 7–8 funguje pro návrh PCR primerů poměrně spolehlivě[5]. Kód pro získání seznamu všech přijatelných překrytí nalezneme v příloze A, v sekci č. A.1.

Krok 2 – Výpočet termálních parametrů

V tomto kroku budeme procházet jedno překrytí za druhým a pro každé z nich spočítáme termální parametry. Pokud se ve kterémkoliv překrytí ukáže, že primery nejsou kompatibilní, končíme výpočet a vracíme *false*.

Třída *ThermalResult* uvedená (níže) na ř. 3 je pouze další datovou třídou. Slouží nám k uchování hodnot ΔG_T° a T_m . Na ř. 5 si vyžádáme seznam překrytí, abychom jej mohli na ř. 6 procházet cyklem. Příkaz *computeThermal(aResult)* na ř. 7 vypočte termální parametry. Stačí mu k tomu pouze řetězec překrytí, který máme pro každé jednotlivé překrytí uložen v *aResults* a vstupní parametry, které získá ze třídy *DimerData*. Výsledek pak uloží do *tResult*. Na ř. 8 už jen ověříme podmínku kompatibility zavoláním přetížené metody *isCompatible(ThermalResult tResult)*. Implementaci této metody si ukážeme ve třetím kroku.

```
1 public boolean isCompatible(Primer left, Primer right) {  
2     ArrayList<AlignmentResult> aResults;  
3     ThermalResult tResult;  
4     aResults = align(left, right);
```



```
5  if (aResults == null) return true;
6  for (AlignmentResult aResult:aResults) {
7      tResult = computeThermal(aResult);
8      if (!isCompatible(tResult)) return false;
9  }
10 return true;
11 }
```

Samotný výpočet termálních parametrů spočívá ve výpočtu tabulkových parametrů (níže, ř. 4–6), aplikaci korekcí (ř. 7–12) a výpočtu teploty tání (ř. 14). Konstrukce metody *computeThermal()* vypadá takto:

```
1 private ThermalResult computeThermal(AlignmentResult aResult) {
2     double g, s, salt, tempC, tempK;
3     tempK = dimerParams.getTemp() + 273.15;
4     g = deltaG(aResult.matchString);
5     s = deltaS(aResult.matchString);
6     h = deltaH(aResult.matchString);
7     double numPhos = (primer1.length() + primer2.length()) * 0.5 - 1;
8     double logSalt = Math.log(dimerParams.getSalt());
9     double gCorr = g - 0.114 * numPhos * logSalt;
10    double sCorr = s + 0.368 * numPhos * logSalt;
11    double hCorr = gCorr + (310.15 * sCorr) * 0.001;
12    double gtCorr = gCorr + ((310.15 - tempK) * sCorr) *
13        0.001 + 0.438 * (aResult.mismatches);
14    double tmCorr = meltingTemperature(hCorr, sCorr);
15    return new ThermalResult(tmCorr, gtCorr);
16 }
```

Implementace metod *deltaG()*, *deltaS()*, *deltaH()* a *meltingTemperature()* s použitím hash-mapy pro efektivní konstrukci tabulky parametrů jsou pro svůj větší rozsah uvedeny v příloze A.

Krok 3 – Rozhodnutí podmínek kompatibility

Na základě výsledků termální analýzy provedeme rozhodnutí kompatibility pro dané překrytí. K dispozici máme hodnoty ΔG_T° a T_m uložené v *tResult*. Prahové hodnoty jsou *dimerParams.getTm()* a *dimerParams.getDeltaG()*. Pokud je pravdivá alespoň jedna z podmínek na řádcích 2–3, pak primery prohlásíme za nekompatibilní. V opačném případě jsou kompatibilní.

```
1 private boolean isCompatible(ThermalResult tResult) {
2     cond1 = tResult.tm > dimerParams.getTm();
3     cond2 = tResult.deltaG < dimerParams.getDeltaG();
4
5     if (cond1 || cond2) return false;
6     return true;
7 }
```

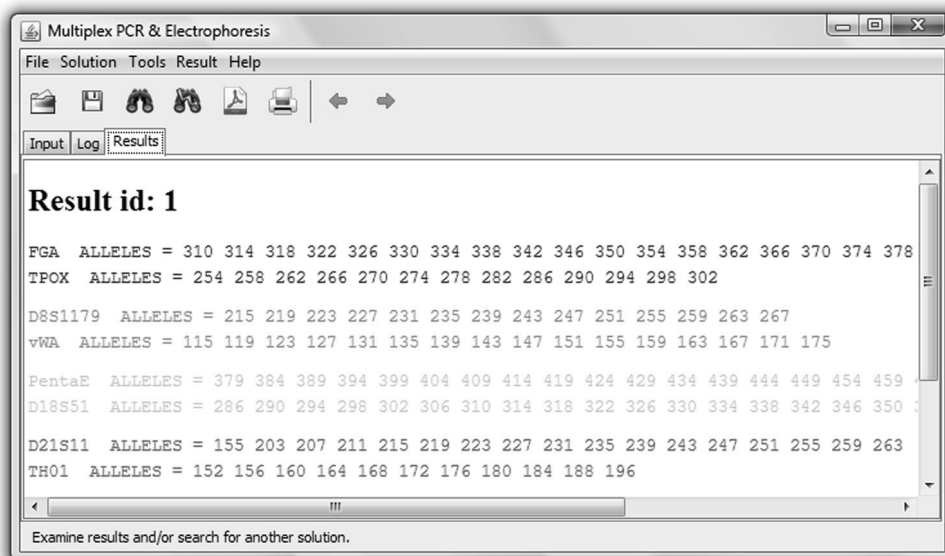
5.5 Sestavování multiplexů

Sestavování multiplexů spočívá ve výběru kompatibilních sad mikrosatelitů. Uživatel si volí požadovanou velikost řešení a počet elektroforézních kanálů, které budou během experimentu využity. Na základě těchto údajů jsou vygenerovány všechny existující kompatibilní sady požadovaných velikostí. Z nich jsou poté metodou *searchNext* sestavovány cílové multiplexy. Prostor řešení procházíme do hloubky („depth-first-search“). Implementace metody *searchNext* pro nalezení cílového multiplexu s použitím rychlé bytové indexace vypadá takto:

```
1 public void searchNext() {
2     if (size == goalSize) {
3         backtrack(); size--;
4     }
5     bestSize = 0;
6     while (size < goalSize && size >= 0) {
7         while (size < goalSize && findNextElement()) {
8             solution[size++] = forward();
9         }
10        if (size > bestSize) {
11            bestSize = size;
12            System.arraycopy(solution, 0, bestSolution, 0, size);
13        } else {
14            if (size > 0) backtrack();
15            size--;
16        }
17    }
18 }
```

Výše uvedená metoda *searchNext()* je obsažena ve třídě *BBSolver* (modul Solver). Cyklus na ř. 6 opakujeme tak dlouho, dokud nenalezneme požado-

vané řešení. Vnořený cyklus na ř. 7 postupně přidává kompatibilní sady, pokud jsou k dispozici. Tím navyšujeme velikost našeho výsledného multiplexu. Dosáhneme-li požadované velikosti, pak známe řešení a zkopírujeme jej z pole *solution* do výsledného pole *bestSolution* (ř. 12). V opačném případě jsme našli jen částečné řešení a zkusíme hledat znovu jinou cestou. Podmínka na ř. 2 rozhoduje, zda jsme v minulém hledání našli částečné a nebo úplné řešení. Podle toho pak začínáme s hledáním nového řešení a nebo hledáme řešení obdobné. Uživateli prezentujeme pouze úplné řešení. Metoda *findNextElement* na ř. 7 hledá použitelné sady, metoda *forward* na ř. 8 poznamenává mezivýsledek do pole řešení, metoda *backtrack* na ř. 15 naopak mezivýsledek z pole řešení odstraňuje. Na obr. č. 10 můžeme vidět, jak aplikace prezentuje sestavený kompatibilní multiplex.



Obrázek 10: Ukázková sada kompatibilních mikrosatelitů.

5.6 Složitost algoritmu

Tato sekce slouží k popsání časové náročnosti výpočetního algoritmu. Hledání výsledného multiplexu probíhá ve třech fázích – 1. určení kompatibility primerů, 2. vytvoření všech možných skupin kompatibilních mikrosatelitů, 3. sestavení multiplexu z několika kompatibilních skupin. Přesněji určit lze pouze časová složitost 1. fáze (viz odstavec „Časová složitost 1. fáze algoritmu“). Ve

2. fázi je teoreticky možno vygenerovat exponenciálně mnoho kompatibilních skupin (vzhledem k počtu N zpracovávaných mikrosatelitů). Ve 3. fázi pak vybíráme do výsledného multiplexu skupiny mikrosatelitů mezi všemi kompatibilními skupinami, výpočet tedy opět může mít exponenciální časovou složitost. Z toho vyplývá, že omezení počtu kompatibilních skupin mikrosatelitů vhodným nastavením kritérií kompatibilit v 1. fázi má klíčový význam pro efektivitu algoritmu. Proto je třeba mít na paměti, že doba zpracování velmi závisí na vstupních datech i na konkrétním nastavení vyhledávacích parametrů. V neposlední řadě pak pochopitelně na použitém hardwaru.

Pro testovací účely byl použit počítač s dvoujádrovým procesorem architektury Intel s taktem jádra 1.87 Ghz a pamětí 5 GB. Testování probíhalo v operačním systému Windows Vista x64. Použité spouštěcí prostředí bylo Java Runtime 6 Update 11.

Časová složitost 1. fáze algoritmu

Nejdříve si zkusme spočítat, jak rychle nám narůstá počet vzájemných porovnání primerů s narůstajícím počtem vstupních mikrosatelitů. Víme, že chceme porovnat každý mikrosatelit s každým kromě sebe sama. To nám dává $\frac{N^2}{2} - N$ kombinací. Každé porovnání dvou mikrosatelitů sestává ze čtyř porovnání primerů. Tím pádem dochází k $2N^2 - 4N$ porovnání primerů. Uvážíme-li, že během výpočtu chemické kompatibility bude maximální délka primeru L , pak při každém porovnání dochází maximálně k $2L - 1$ překrytí řetězců. Pro každé překrytí počítáme termální parametry. Pro určení chemické kompatibility N mikrosatelitů provedeme výpočet maximálně K -krát, kde K je rovno:

$$K(L, N) = (2L - 1)(2N^2 - 4N) \quad (18)$$

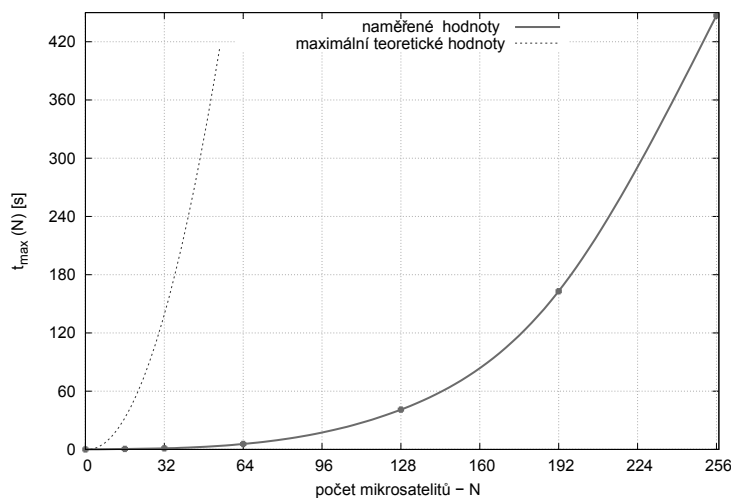
Tabulka č. 5 znázorňuje, jak rychle nám narůstá počet kombinací s rostoucím počtem porovnávaných mikrosatelitů. Délku L budeme uvažovat 24 nukleotidů.

Nyní si zkusíme porovnat, jaký bude časový rozdíl při použití optimalizačního parametru „Score“ a bez jeho použití. Měření na naší pokusné sestavě ukázalo, že průměrná doba pro určení kompatibility dvou primerů byla 1,54 ms. Vynásobíme-li touto dobou počet kombinací pro dané N , získáme teoretickou maximální dobu průběhu celého algoritmu výpočtu chemické kompatibility. Délku primerů budeme opět uvažovat 24 nukleotidů, jelikož to je průměrná délka v naší testovací skupině a zároveň uváděná optimální délka primeru.

Teoretickou maximální dobu zpracování $t_{\max}$ vypočítáme takto:

$$t_{\max}(N) = 47(2N^2 - 4N) \times 1,54 \times 10^{-3} \quad (19)$$

Pro srovnání také změříme skutečnou dobu výpočtu. Měření provedeme na testovacích množinách s 16, 32, 64, 128, 192 a 256 mikrosatelity. Výsledek shrnuje graf na obr. č. 11.



Obrázek 11: Srovnání teoretické maximální doby zpracování a naměřené doby zpracování chemické kompatibility pro skupiny N mikrosatelitů.

Na grafu vidíme, jak rychle by nám rostla časová náročnost kdybychom brali v potaz všechny existující kombinace (čárkovaná křivka). Za těchto podmínek by výpočet pro $N = 256$ trval více než 2,5h. My však uplatňujeme skóre,

N	Počet kombinací (K)
16	22 560
32	93 248
64	379 008
128	1 528 064
192	3 447 168
256	6 136 320

Tabulka 5: Nárůst složitosti při výpočtu chemické kompatibility mikrosatelitů, jejichž primery mají délku 24 nukleotidů.

čímž počet přijatelných případů rapidně snížíme. Navíc, nalezneme-li překrytí dvou primerů, při kterém by mohla vzniknout vazba, ihned primery označíme za nekompatibilní a dále je nezkoumáme. Tímto způsobem výpočet značně urychlíme. Reálně naměřené hodnoty ukazuje tučněji vyznačená křivka. Skutečně naměřené hodnoty jsou na křivce zvýrazněny, ostatní hodnoty byly dopočítány.

Hodnoty znázorněné v grafu musejí být pochopitelně brány jen orientačně, jelikož zobrazují dobu zpracování, která se bude lišit podle situace. Vidíme zde však významný vliv optimalizací. Operační složitost algoritmu podle rovnice č. (18) je $O(N^2)$.

Paměťové nároky aplikace

Měření paměťové náročnosti aplikace na naší testovací sestavě bylo opět provedeno pro vstupní množiny s 16, 32, 64, 128, 192 a 256 mikrosatelity. Výsledky ukazují, že až po $N = 32$ paměťové nároky výrazně rostou (do hodnoty 520MB při $N = 64$). Odtud paměťová náročnost roste jen pozvolna do 556MB při $N = 256$. Pro objektivní změření byla aplikace před novým měřením vždy restartována. Naměřené výsledky shrnuje tabulka č. 6.

N	16	32	64	128	192	256
Paměť [MB]	128	276	520	540	553	556

Tabulka 6: Naměřené paměťové nároky aplikace.

Na první pohled je patrné, že jsou paměťové nároky aplikace relativně vysoké. Je ovšem třeba zohlednit, že test byl spuštěn na sestavě s dostatkem operační paměti. V takových případech je pro urychlení výpočtu alokováno spíše více paměti a nepoužitá paměť uvolňována jen pozvolna. Počítače s menším množstvím operační paměti budou schopny program přesto spustit, avšak lze očekávat snížení jeho výkonu kvůli častým dealokacím nepoužívaných paměťových bloků. Jiné implementace Java Virtual Machine mohou využívat paměť odlišným způsobem.

6 Dosažené výsledky

Korespondence vypočtených výsledků se skutečností má zásadní význam pro použitelnost naší aplikace. Jedině tak může být nasazena v reálném provozu pro účely, ke kterým byla vytvořena – tedy pro zvýšení propustnosti elektroforézy optimalizací primerů. V této kapitole se pokusíme ověřit vypočtené výsledky porovnáním vůči známé sadě mikrosatelitů, která je určena pro lidskou identifikaci. Nejprve si shrneme základní informace o uvažované sadě.

Powerplex 16[®] BIO

Jedna z populárních, běžně používaných mikrosatelitových sad se nazývá Powerplex 16. Využíváme ji v soudním lékařství, při genotypizování (včetně ověřování rodičovství) a také pro identifikaci lidských jedinců. Jediněný výběr mikrosatelitů umožňuje provádět jednorázovou amplifikaci, aniž by hrozily vzájemné interakce použitých primerů. Analýza na elektroforézním gelu může probíhat v jediném kanálu při použití třech fluorescenčních barev, nebo ve třech kanálech při použití jedné barvy. Obsažené mikrosatelity splňují standardy FBI pro lidskou identifikaci v rámci celé populace. Třináct ze šestnácti obsažených mikrosatelitů využívá FBI ve svém vyhledávacím systému CODIS. Čtrnáctý mikrosatelit, Amelogenin slouží k určení pohlaví. Zbýlé dva mikrosatelity pak dále navyšují rozlišovací schopnost sady. Pravděpodobnost, že budou dva různí europoidní jedinci identifikováni touto sadou stejně je 1 ku $1,83 \times 10^{17}$, tedy minimální[30].

Sadu Powerplex 16 jsme pro ověření funkcí programu vybrali záměrně. Díky vysoké odolnosti zvolených primerů vůči hybridizaci můžeme provést testování přes celou škálu vstupních hodnot. Jakmile nalezneme parametry, při kterých nám program vyhodnotí mikrosatelity za nekompatibilní, provedeme zhodnocení, zda jsou tyto parametry ještě přípustné. Pokud by přípustné byly, chyba bude zřejmě v našem programu. V opačném případě lze algoritmus považovat za správný.

Bohužel jsme pro vytíženost konzultační laboratoře dosud nestihli provést praktický laboratorní test uzpůsobený specificky pro naše potřeby. Zcela průkazné ověření funkčnosti metody tedy není součástí této práce, lze se pouze spoléhat na silné teoretické podklady, ze kterých tato práce čerpá. Např. [3].

Parametry testu na sadě Powerplex 16

Test chemické kompatibility provedeme za použití výchozích vstupních parametrů. Tyto parametry na záložce SolverData jsou: Rozsah povolených délek amplikonů 1–999, požadovaná velikost řešení – 16 a tři pracovní kanály. Záložka DimerData bude obsahovat: Skóre – 7, množství Na^+ – 0,085 mol, koncentrace vláken – 1 μmol , teplota – 37° C, prahová teplota tání T_m – 64° C, ΔG° bude -10.

V průběhu testu budeme měnit vždy jen jeden parametr naráz a to do doby, než dosáhne nuly. Měnit budeme obě prahové hodnoty. T_m budeme snižovat po deseti a ΔG° budeme zvyšovat po jedné.

Výsledek a diskuze

Provedené testy chemické kompatibility ukazují (viz tabulky č. 7 a 8), že se první nekompatibilita projeví až při prahové hodnotě T_m rovno 4° C, tedy těsně před úplným minimem. Pro T_m menší než 0 už žádné vazby prakticky ani vznikat nemohou. Při změnách prahu ΔG° se nekompatibilita dvou mikrosatelitů projeví také až těsně před nulou.

Práh T_m [° C]	64	54	44	34	24	14	4
N	0	0	0	0	0	0	1

Tabulka 7: Výsledek testu odolnosti mikrosatelitů proti hybridizaci. Postupně měníme prahovou hodnotu pro teplotu tání a sledujeme, kolik mikrosatelitů bude označeno za nekompatibilní (řádek 'N').

Práh ΔG° [kcal/mol]	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
N	0	0	0	0	0	0	0	0	0	1	1

Tabulka 8: Výsledek testu při změnách prahové hodnoty ΔG° .

Tyto výsledky naznačují, že zkoumaná množina bude mít velkou odolnost proti hybridizaci. Parametry, při kterých byly mikrosatelity označeny za nekompatibilní jsou zcela mimo škálu použitelnosti. Např. při teplotě 4° C experimenty obvykle neprovádíme.

Rozdělení mikrosatelitů do skupin na základě délkových kompatibilit proběhlo také v pořádku. Žádné z nalezených řešení neobsahovalo zakázané překrytí.

Nalezené řešení s identifikačním číslem 148 navíc odpovídalo referenčnímu rozdělení do tří skupin, které můžeme nalézt na adrese: <http://www.cstl.nist.gov/div831/strbase/kits/PowerPlex16.htm> [27. 4. 2009].

Detailní výstup aplikace můžeme vidět na obr. 12–14. V levém sloupci je vždy uvedena délka alel platná pro daný řádek. Jednotlivé sloupce odpovídají elektroforézním kanálům. Z tohoto výstupu je jasné patrné, že se žádné dvě alely v jednom kanálu nepřekrývají.

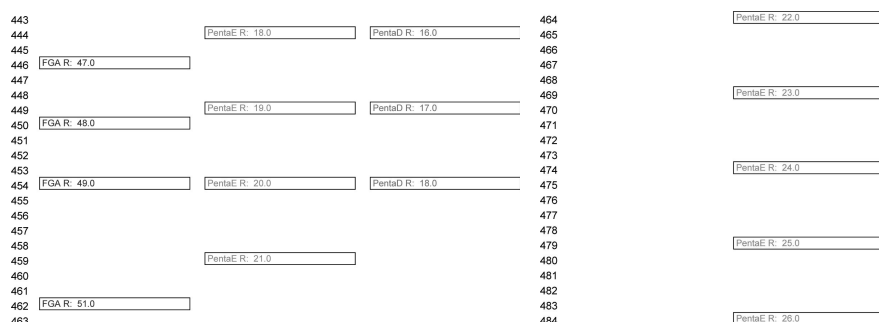
99		D3S1358 R: 8.0		174			
100				175	WVA R: 25.0		
101				176		D13S317 R: 7.0	TH01 R: 9.0
102				177			
103		D3S1358 R: 9.0		178			
104				179			
105				180		D13S317 R: 8.0	TH01 R: 10.0
106	Amelogenin R: 0.0			181			
107		D3S1358 R: 10.0		182			
108				183			
109				184		D13S317 R: 9.0	TH01 R: 11.0
110				185			
111		D3S1358 R: 11.0		186			
112	Amelogenin R: 1.0			187			
113				188		D13S317 R: 10.0	TH01 R: 12.0
114				189			
115	WVA R: 10.0	D3S1358 R: 12.0	D5S818 R: 6.0	190			
116				191			
117				192		D13S317 R: 11.0	
118				193			
119	WVA R: 11.0	D3S1358 R: 13.0	D5S818 R: 7.0	194			
120				195			
121				196		D13S317 R: 12.0	TH01 R: 14.0
122				197			
123	WVA R: 12.0	D3S1358 R: 14.0	D5S818 R: 8.0	198			
124				199			
125				200		D13S317 R: 13.0	
126				201			
127	WVA R: 13.0	D3S1358 R: 15.0	D5S818 R: 9.0	202			
128				203		D21S11 R: 24.0	
129				204		D13S317 R: 14.0	
130				205			
131	WVA R: 14.0	D3S1358 R: 16.0	D5S818 R: 10.0	206			
132				207		D21S11 R: 25.0	
133				208		D13S317 R: 15.0	
134				209			
135	WVA R: 15.0	D3S1358 R: 17.0	D5S818 R: 11.0	210			
136				211		D21S11 R: 26.0	D7S820 R: 5.0
137				212		D13S317 R: 16.0	
138				213			
139	WVA R: 16.0	D3S1358 R: 18.0	D5S818 R: 12.0	214			
140				215	D8S1179 R: 7.0	D21S11 R: 27.0	D7S820 R: 6.0
141				216		D13S317 R: 17.0	
142				217			
143	WVA R: 17.0	D3S1358 R: 19.0	D5S818 R: 13.0	218			
144				219	D8S1179 R: 8.0	D21S11 R: 28.0	D7S820 R: 7.0
145				220			
146				221			
147	WVA R: 18.0	D3S1358 R: 20.0	D5S818 R: 14.0	222			
148				223	D8S1179 R: 9.0	D21S11 R: 29.0	D7S820 R: 8.0
149				224			
150				225			
151	WVA R: 19.0		D5S818 R: 15.0	226			
152			TH01 R: 3.0	227	D8S1179 R: 10.0	D21S11 R: 30.0	D7S820 R: 9.0
153				228			
154				229			
155	WVA R: 20.0	D21S11 R: 12.0	D5S818 R: 16.0	230			
156			TH01 R: 4.0	231	D8S1179 R: 11.0	D21S11 R: 31.0	D7S820 R: 10.0
157				232			
158				233			
159	WVA R: 21.0		D5S818 R: 17.0	234			
160			TH01 R: 5.0	235	D8S1179 R: 12.0	D21S11 R: 32.0	D7S820 R: 11.0
161				236			
162				237			
163	WVA R: 22.0		D5S818 R: 18.0	238			
164			TH01 R: 6.0	239	D8S1179 R: 13.0	D21S11 R: 33.0	D7S820 R: 12.0
165				240			
166				241			
167	WVA R: 23.0			242			
168		D13S317 R: 5.0	TH01 R: 7.0	243	D8S1179 R: 14.0	D21S11 R: 34.0	D7S820 R: 13.0
169				244			
170				245			
171	WVA R: 24.0			246			
172		D13S317 R: 6.0	TH01 R: 8.0	247	D8S1179 R: 15.0	D21S11 R: 35.0	D7S820 R: 14.0
173				248			

Obrázek 12: Výsledné rozdělení sady Powerplex 16, alely délek 90–248 bp.

6 DOSAŽENÉ VÝSLEDKY

249				345		
250	D8S1179 R: 16.0	D21S11 R: 36.0	D7S820 R: 15.0	346	FGA R: 22.0	D18S51 R: 22.0
251				347		
252				348		
253				349	CSF1PO R: 13.0	
254	TPOX R: 4.0			350	FGA R: 23.0	D18S51 R: 23.0
255	D8S1179 R: 17.0	D21S11 R: 37.0	D7S820 R: 16.0	351		
256				352		
257				353		
258	TPOX R: 5.0			354	FGA R: 24.0	CSF1PO R: 14.0
259	D8S1179 R: 18.0	D21S11 R: 38.0		355		D18S51 R: 24.0
260		D16S539 R: 4.0		356		
261				357		
262	TPOX R: 6.0			358	FGA R: 25.0	D18S51 R: 25.0
263	D8S1179 R: 19.0	D21S11 R: 39.0		359		CSF1PO R: 15.0
264		D16S539 R: 5.0		360		
265				361		
266	TPOX R: 7.0			362	FGA R: 26.0	D18S51 R: 26.0
267	D8S1179 R: 20.0			363		
268		D16S539 R: 6.0		364		CSF1PO R: 16.0
269				365		
270	TPOX R: 8.0			366	FGA R: 27.0	D18S51 R: 27.0
271				367		
272		D16S539 R: 7.0		368		
273				369		
274	TPOX R: 9.0			370	FGA R: 28.0	
275		D16S539 R: 8.0		371		
276				372		
277	TPOX R: 10.0			373	FGA R: 29.0	
278				374		
279		D16S539 R: 9.0		375		
280				376		
281				377	FGA R: 30.0	
282	TPOX R: 11.0			378		PentaE R: 5.0
283				379		
284		D16S539 R: 10.0		380		
285				381	FGA R: 31.0	
286	TPOX R: 12.0		D18S51 R: 7.0	382		
287				383		PentaE R: 6.0
288		D16S539 R: 11.0		384		PentaD R: 4.0
289				385		
290	TPOX R: 13.0		D18S51 R: 8.0	386	FGA R: 32.0	
291				387		
292		D16S539 R: 12.0		388		
293				389		PentaE R: 7.0
294	TPOX R: 14.0		D18S51 R: 9.0	390		PentaD R: 5.0
295				391		
296		D16S539 R: 13.0		392		
297				393		
298	TPOX R: 15.0		D18S51 R: 10.0	394		PentaE R: 8.0
299				395		PentaD R: 6.0
300		D16S539 R: 14.0		396		
301				397		
302	TPOX R: 16.0		D18S51 R: 11.0	398		
303				399		PentaE R: 9.0
304		D16S539 R: 15.0		400		PentaD R: 7.0
305				401		
306			D18S51 R: 12.0	402		
307				403		
308		D16S539 R: 16.0		404		PentaE R: 10.0
309		CSF1PO R: 5.0		405		PentaD R: 8.0
310	FGA R: 13.0		D18S51 R: 13.0	406		
311				407		
312				408		
313				409		PentaE R: 11.0
314	FGA R: 14.0	CSF1PO R: 6.0	D18S51 R: 14.0	410		PentaD R: 9.0
315				411		
316				412		
317				413		PentaE R: 12.0
318	FGA R: 15.0		D18S51 R: 15.0	414		PentaD R: 10.0
319		CSF1PO R: 7.0		415		
320				416		
321				417		
322	FGA R: 16.0		D18S51 R: 16.0	418		
323				419		PentaE R: 13.0
324		CSF1PO R: 8.0		420		PentaD R: 11.0
325				421		
326	FGA R: 17.0		D18S51 R: 17.0	422		
327				423		PentaE R: 14.0
328				424		PentaD R: 12.0
329		CSF1PO R: 9.0		425	FGA R: 42.0	
330	FGA R: 18.0		D18S51 R: 18.0	426		
331				427		
332				428		PentaE R: 15.0
333				429		PentaD R: 13.0
334	FGA R: 19.0	CSF1PO R: 10.0	D18S51 R: 19.0	430		
335				431		
336				432		
337				433		
338	FGA R: 20.0		D18S51 R: 20.0	434	FGA R: 44.0	PentaE R: 16.0
339		CSF1PO R: 11.0		435		PentaD R: 14.0
340				436		
341				437		
342	FGA R: 21.0		D18S51 R: 21.0	438	FGA R: 45.0	
343				439		PentaE R: 17.0
344		CSF1PO R: 12.0		440		PentaD R: 15.0
				441		
				442	FGA R: 46.0	

Obrázek 13: Výsledné rozdělení sady Powerplex 16, alely délek 249–442 bp.



Obrázek 14: Výsledné rozdělení sady Powerplex 16, alely délek 443–484 bp.

Provedli jsme ještě obdobné testy na dvou dalších sadách mikrosatelitů, které nám poskytla k otestování naše konzultační laboratoř. Tyto sady se jmenují *human_multiplex* a *str_database*. Povedlo se nám z nich opět sestavit kompatibilní multiplexy. Detailní výstup aplikace z těchto testů, nastavení vstupních parametrů, matice kompatibilit a popis problematických primerů nalezneme v příloze B.

human_multiplex – Jedná se o menší sadu čítající 10 mikrosatelitů, která obsahuje markery pro určování lidského genotypu z databáze „The Cooperative Human Linkage Center“. Navíc ještě obsahuje mikrosatelit HUMTPOX, pomocí kterého jsme schopni měřit množství protilátek namířených proti hormonům štítné žlázy a odhalit tak její nesprávnou funkci.

str_database – Rozsáhlejší sada čítající směs celkem 27 různých lidských mikrosatelitů sloužících např. pro vyhledávání patogenů, nebo pro detekci Von Willebrandovy choroby².

²dědičná porucha srážlivosti krve

7 Závěr

Tato práce si kladla za cíl navržení a implementaci sady matematických metod, které urychlují laboratorní genotypizaci vhodným výběrem kompatibilních mikrosatelitů. Vytvořené algoritmy pro výpočet kompatibility primerů byly ověřovány na třech sadách mikrosatelitů. Ve všech případech byly výpočetní testy úspěšné, shodovaly se s předpokládaným rozdělením.

Implementace výpočetních metod v programu MultiPCR zpřístupňuje forenzním laboratorům metody výzkumu, které by jinak mohly být příliš nákladné. S pomocí tohoto nástroje jsou schopny snížit náklady na pořízování elektroforézního gelu a dalších preparátů tím, že umožní provést více experimentů najednou. Kontrola délkových kompatibilit umožní současné použití více mikrosatelitů v jediném kanálu, přičemž bude stále možné přesně identifikovat jednotlivé alely na elektroforézním gelu.

Aplikace byla optimalizována pro vícejádrové procesory (resp. víceprocesorové systémy), takže dokáže plně využít jejich výkon. Používáme v ní unikátní heuristiky ořezávající velmi účinně strom možných řešení, takže kombinatorická exploze prostoru řešení, který prohledáváme do hloubky metodou „depth-first search“, je tím dramaticky omezena. Experimenty ukázaly, že vytvořený software dovoluje efektivně nacházet multiplexy vybrané ze sad desítek nebo i stovek (do počtu 256) mikrosatelitů, což je pro praktické nasazení více než dostatečné.

V případě odhalení chemické nekompatibility program poskytuje informace potřebné k identifikaci problematických primerů. Je-li k dispozici dostatečný počet kanálů, mikrosatelity obsahující problematické primery jsou od sebe ve výsledné sadě odděleny. Chemicky kompatibilní mikrosatelity jde amplifikovat společně, čímž se sníží nutný počet provedených polymerázových řetězových reakcí.

Seznámení s aplikací usnadňuje kontextová nápověda. V místech, kde je k dispozici detailnější popis (např. při zadávání vstupních parametrů) bývá umístěno tlačítko s jasně viditelným zeleným otazníkem. Kratší popisky jsou řešeny formou tooltipových textů, které se zobrazují při najetí myši na požadovanou položku. Nejpravděpodobněji další činnost uživatele je popsána ve spodní části okna (např. informace, že jsou již k dispozici výsledky a že by si je měl prostudovat).

Modulární podoba vytvořené aplikace otevírá možnosti pro její další rozvoj, obzvláště v oblasti řešení chemické kompatibility. Současný algoritmus lze vylepšit započtením tepelné kapacity ΔC_p° , popřípadě rozšířením na me-

todu „další nejbližší soused“ (NNN – Next Nearest Neighbour) s využitím neuronové sítě. Vstup aplikace by mohl být načítán z biologických databází, nebo být nahrazen generováním primerů „de-novo“ dle požadovaných vlastností.

MultiPCR oproti všem nám doposud známým konkurenčním programům (Autodimer[5], Primer3[12] a FastPCR[13]) poskytuje navíc unikátní metodu testování délkových kompatibilit mikrosatelitů. Také umožňuje díky heuristickému prohledávání stavového prostoru sestavovat automaticky celé multiplexy z kompatibilních mikrosatelitů. Výpočty intramolekulárních vazeb pro určování chemické kompatibility vycházejí z nejnovějších poznatků o chování primerů při PCR. Jako jediný může pracovat nativně pod libovolným operačním systémem s prostředím Java Runtime, čímž využívá procesorový čas mnohem efektivněji než ostatní aplikace spouštěné pod emulátory (Např. FastPCR musí být v Linuxu spouštěn pod Wine, v Mac OS pod emulátorem VirtualPC, Autodimer podporu OS neuvádí vůbec).

Věřím tomu, že by se brzy mohla aplikace MultiPCR zařadit do běžné softwarové výbavy forenzních genetiků. V současné době probíhá její pilotní provoz v Laboratoři experimentální medicíny při Dětské klinice LF UP a FN Olomouc. Pro účely této laboratoře byla prvotně navržena.

A Příloha

A.1 Vytvoření seznamu všech překrytí

```
1 public ArrayList<AlignmentResult> align(Primer p1, Primer p2) {
2     SymbolList sList1, sList2;
3     ArrayList<AlignmentResult> results;
4     int k1, k2, len, matches, mismatches;
5     StringBuffer mString;
6
7     mString = new StringBuffer();
8     results = new ArrayList<AlignmentResult>();
9     sList1 = DNATools.createDNA(p1.seqString());
10    sList2 = DNATools.createDNA(p2.seqString());
11    int slideCount = p1.length() + p2.length();
12
13    for (int i=1; i < slideCount; i++) {
14        k1 = Math.max(sList1.length() - i, 0);
15        k2 = Math.max(i - sList1.length(), 0);
16        len = Math.min(sList1.length() - k1, sList2.length() - k2);
17        matches = 0; mismatches = 0; mString.setLength(0);
18        for (int j=0; j<len; j++) {
19            if (DNATools.complement(sList1.symbolAt(j + k1 + 1))
20                .equals(sList2.symbolAt(j + k2 + 1))) {
21                matches++;
22                mString.append(sList1.subStr(j + k1 + 1, j + k1 + 1));
23            } else {
24                mismatches++; mString.append("N");
25            }
26        }
27        int score = matches - mismatches;
28        if (mString.equals("N")) continue;
29        if (score >= dimerParams.getScore()) {
30            results.add(new AlignmentResult(
31                mString.toString(), matches, mismatches, i));
32        }
33    }
34    return results;
35 }
```

Na ř. 9–10 (v bloku kódu uvedeném výše) převádíme naše primerové ře-

těžce do zpracovatelnějšího formátu – *SymbolList* z balíku BioJava. Ten nám umožní na ř. 19 provést porovnání komplementarity dvou symbolů. Hlavní cyklus na ř. 13 prochází všechna možná posunutí, vnořený cyklus na ř. 18 pro každé posunutí zkontroluje komplementaritu protilehlých symbolů. Zároveň zjišťujeme počet shod, počet neshod a vytváříme řetězec překrytí. Na ř. 28 odfiltrujeme ta překrytí, kde se kryje jen jeden nukleotid a ten navíc ani není komplementární. Na ř. 29 vyhodnotíme skóre. Do výstupního seznamu přidáváme všechna posunutí, která jsou rovna prahové hodnotě, nebo vyšší.

A.2 Obecná metoda výpočtu termálního parametru

Metoda *computeParam* vypočte na základě řetězce překrytí a zvolené tabulky požadovaný termální parametr. Např. pro výpočet ΔH° s řetězcem překrytí "ATTGA" bychom volali: `computeParam("ATTGA", tG)`; kde *tG* je hashovací tabulka vytvořená metodou *createTableH()*.

```
1 private cmdFloat computeParam(String mString,
2                               HashMap<String, Float> table) {
3     Float nn;
4     float param = 0;
5     float init = 0;
6
7     // Výpočet hodnot nejbližšího souseda
8     for (int i = 0; i < s1.length() - 1; i++) {
9         nn = table.get(s1.substring(i, i+2));
10        if (nn == null) continue;
11        param += nn;
12    }
13
14    // Výpočet inicializačního parametru
15    nn = table.get(String.valueOf(s1.charAt(s1.length()-1)));
16    if (nn != null) init += nn;
17
18    return init+param;
19 }
```

A.3 Implementace datových tabulek

Implementace tabulky pro výpočet ΔG_{37}°

```

1 private HashMap<String, Float> createTableG() {
2     tG = new HashMap<String, Float>(16);
3
4     // unified free energy at 37C
5     tG.put("aa", -1.00f); tG.put("at", -0.88f);
6     tG.put("ac", -1.44f); tG.put("ag", -1.28f);
7     tG.put("ta", -0.58f); tG.put("tt", -1.00f);
8     tG.put("tc", -1.30f); tG.put("tg", -1.45f);
9     tG.put("ca", -1.44f); tG.put("ct", -1.28f);
10    tG.put("cc", -1.84f); tG.put("cg", -2.17f);
11    tG.put("ga", -1.30f); tG.put("gt", -1.44f);
12    tG.put("gc", -2.24f); tG.put("gg", -1.42f);
13
14    // inic. korekce G-C
15    tG.put("g", 0.98f); tG.put("c", 0.98f);
16
17    // inic. korekce A-T
18    tG.put("a", 1.03f); tG.put("t", 1.03f);
19
20    tG.put("n", 0.00f);
21    return tG;
22 }
```

Implementace tabulky pro výpočet ΔH°

```

1 private HashMap<String, Float> createTableH() {
2     tH = new HashMap<String, Float>(16);
3
4     // unified delta H parameters (kcal/mol)
5     tH.put("aa", -7.90f); tH.put("at", -7.20f);
6     tH.put("ac", -8.40f); tH.put("ag", -7.80f);
7     tH.put("ta", -7.20f); tH.put("tt", -7.90f);
8     tH.put("tc", -8.20f); tH.put("tg", -8.50f);
9     tH.put("ca", -8.50f); tH.put("ct", -7.80f);
10    tH.put("cc", -8.00f); tH.put("cg", -10.60f);
11    tH.put("ga", -8.20f); tH.put("gt", -8.40f);
12    tH.put("gc", -9.80f); tH.put("gg", -8.42f);
```



```
13
14 // inic. korekce G-C
15 tH.put("g", 0.10f);
16 tH.put("c", 0.10f);
17
18 // inic. korekce A-T
19 tH.put("a", 2.30f);
20 tH.put("t", 2.30f);
21
22 tH.put("n", 0.00f);
23 return tH;
24 }
```

Implementace tabulky pro výpočet ΔS°

```
1 private HashMap<String, Float> createTableS() {
2     tS = new HashMap<String, Float>(16);
3
4     // unified delta S parameters (cal/k.mol)
5     tS.put("aa", -22.20f); tS.put("at", -20.40f);
6     tS.put("ac", -22.40f); tS.put("ag", -21.00f);
7     tS.put("ta", -21.30f); tS.put("tt", -22.20f);
8     tS.put("tc", -22.20f); tS.put("tg", -22.70f);
9     tS.put("ca", -22.70f); tS.put("ct", -21.00f);
10    tS.put("cc", -19.90f); tS.put("cg", -27.20f);
11    tS.put("ga", -22.20f); tS.put("gt", -22.40f);
12    tS.put("gc", -24.40f); tS.put("gg", -19.90f);
13
14    // inic. korekce G-C
15    tS.put("g", -2.80f);
16    tS.put("c", -2.80f);
17
18    // inic. korekce A-T
19    tS.put("a", 4.10f);
20    tS.put("t", 4.10f);
21
22    tS.put("n", 0.00f);
23    return tS;
24 }
```

B Výsledky testovaných sad

B.1 Test sady human_multiplex

Pro testování této sady jsme použili následující parametry vyhledávače: Rozsah délek ampliconů: 0–230, velikost řešení: 10, počet kanálů: 3.

Pro výpočet termálních parametrů jsme použili: Score: 7, množství soli: 0,085 mol, koncentrace vláken: 1 μ mol, teplota: 37°C, teplota tání: 25°C, ΔG° : -3 kcal/mol.

Všechny mikrosatelity ze sady mají délku repetice 4. Nejvhodnější nalezené rozdělení do tří kanálů kanály bylo 3-3-2. Celou sadu tudíž nelze použít dohromady kvůli překrývání některých alel. Najednou lze použít maximálně 8 z 10 mikrosatelitů sady. Chemická nekompatibilita nebyla žádná odhalena. Na obr. 16 vidíme matici délkových kompatibilit.

Name: 1_G08710_human_STS_CHL

Allele list: 7 8 9 10 11 12 13 14 15 16 17 18 19

Name: 2_D6S1017_AL035588.-

Allele list: 6 7 8 9 10 11 12 13

Name: 3_G08202_human_STS_CHL

Allele list: 16 17 18 19 20 21 22 23 24 25 26 27 28

Name: 4_D1S1677_AL513307.-

Allele list: 9 10 11 12 13 14 15 16 18

Name: 5_G08820_human_STS_CHL

Allele list: 9 10 11 12 13 14 15 16 17 18 19

Name: 6_G08085_human_STS_CHL

Allele list: 8 10 11 12 13 14 15 16 17 18 19

Name: 7_G08184_human_STS_CHL

Allele list: 9 10 11 11.3 12 12.3 13 13.3 14 14.3 15 16 17

Name: 8_G08921_human_STS_CHL

Allele list: 15 16 17 18 19 20 21 22 23 24 25 26

Name: 9_G07820_human_STS_CHL

Allele list: 9 10 11 12 13 14.3 15.3 16.3 17.3 18.3 19 19.3

Name: 10_HUMTPOX_Human_thyroi

Allele list: 5 6 7 8 9 10 11 12 13 14

B VÝSLEDKY TESTOVANÝCH SAD

79		4. D1S1677 AL513307 - R: 9.0			
80	2_D6S1017_AL035588 - R: 6.0				
81					
82		G08820 human_STS_CHL R: 9.0			
83		4. D1S1677 AL513307 - R: 10.0			
84	2_D6S1017_AL035588 - R: 7.0		G08184 human_STS_CHL R: 9.0		
85			G08184 human_STS_CHL R: 11.3		
86	G08710 human_STS_CHL R: 7.0	G08820 human_STS_CHL R: 10.0			
87		4. D1S1677 AL513307 - R: 11.0			
88	2_D6S1017_AL035588 - R: 8.0		G08184 human_STS_CHL R: 10.0		
89					
90	G08710 human_STS_CHL R: 8.0	G08820 human_STS_CHL R: 11.0			
91		4. D1S1677 AL513307 - R: 12.0			
92	2_D6S1017_AL035588 - R: 9.0		G08184 human_STS_CHL R: 11.0		
93					
94	G08710 human_STS_CHL R: 9.0	G08820 human_STS_CHL R: 12.0			
95		4. D1S1677 AL513307 - R: 13.0			
96	2_D6S1017_AL035588 - R: 10.0		G08184 human_STS_CHL R: 12.0		
97					
98	G08710 human_STS_CHL R: 10.0	G08820 human_STS_CHL R: 13.0			
99		4. D1S1677 AL513307 - R: 14.0	G08184 human_STS_CHL R: 12.3		
100	2_D6S1017_AL035588 - R: 11.0		G08184 human_STS_CHL R: 13.0		
101					
102	G08710 human_STS_CHL R: 11.0	G08820 human_STS_CHL R: 14.0	HUMTPOX_Human_thyroi R: 5.0		
103		4. D1S1677 AL513307 - R: 15.0	G08184 human_STS_CHL R: 13.3		
104	2_D6S1017_AL035588 - R: 12.0		G08184 human_STS_CHL R: 14.0		
105					
106	G08710 human_STS_CHL R: 12.0	G08820 human_STS_CHL R: 15.0	HUMTPOX_Human_thyroi R: 6.0		
107		4. D1S1677 AL513307 - R: 16.0	G08184 human_STS_CHL R: 14.3		
108	2_D6S1017_AL035588 - R: 13.0		G08184 human_STS_CHL R: 15.0		
109					
110	G08710 human_STS_CHL R: 13.0	G08820 human_STS_CHL R: 16.0	HUMTPOX_Human_thyroi R: 7.0		
111					
112			G08184 human_STS_CHL R: 16.0		
113					
114	G08710 human_STS_CHL R: 14.0	G08820 human_STS_CHL R: 17.0	HUMTPOX_Human_thyroi R: 8.0		
115		4. D1S1677 AL513307 - R: 18.0			
116		G07820 human_STS_CHL R: 9.0	G08184 human_STS_CHL R: 17.0		
117	G08710 human_STS_CHL R: 15.0	G08820 human_STS_CHL R: 18.0	HUMTPOX_Human_thyroi R: 9.0		
118					
119					
120					
121	G08202 human_STS_CHL R: 16.0	G07820 human_STS_CHL R: 10.0			
122	G08710 human_STS_CHL R: 16.0	G08820 human_STS_CHL R: 19.0	HUMTPOX_Human_thyroi R: 10.0		
123					
124					
125	G08202 human_STS_CHL R: 17.0	G07820 human_STS_CHL R: 11.0			
126	G08710 human_STS_CHL R: 17.0		HUMTPOX_Human_thyroi R: 11.0		
127					
128					
129	G08202 human_STS_CHL R: 18.0	G07820 human_STS_CHL R: 12.0			
130	G08710 human_STS_CHL R: 18.0		HUMTPOX_Human_thyroi R: 12.0		
131					
132	G08202 human_STS_CHL R: 19.0	G07820 human_STS_CHL R: 13.0			
133	G08710 human_STS_CHL R: 19.0		HUMTPOX_Human_thyroi R: 13.0		
134					
135					
136					
137	G08202 human_STS_CHL R: 20.0				
138					
139					
140					
141	G08202 human_STS_CHL R: 21.0				
142					
143					
144	G08202 human_STS_CHL R: 22.0	G07820 human_STS_CHL R: 14.3			
145					
146					
147					
148	G08202 human_STS_CHL R: 23.0	G07820 human_STS_CHL R: 15.3			
149					
150					
151					
152					
153	G08202 human_STS_CHL R: 24.0				
154					
155					
156					
157	G08202 human_STS_CHL R: 25.0	G07820 human_STS_CHL R: 16.3			
158					
159					
160					
161	G08202 human_STS_CHL R: 26.0	G07820 human_STS_CHL R: 17.3			
162					
163					
164					
165	G08202 human_STS_CHL R: 27.0	G07820 human_STS_CHL R: 18.3			
166		G07820 human_STS_CHL R: 19.0			
167					
168					
169	G08202 human_STS_CHL R: 28.0				
170		G07820 human_STS_CHL R: 19.3			

Obrázek 15: Výsledné rozdělení sady human_multiplex, alely délek 79–170 bp.

Dále si ukážeme matici délkových kompatibilit a matici celkové kompatibility. V našem případě budou obě matice shodné, jelikož jsou všechny uvedené mikrosatelity vzájemně chemicky kompatibilní.

Length compatibility matrix									
	0	1	2	3	4	5	6	7	8
0					x	x			x
1						x	x		
2								x	
3						x	x		
4	x					x			x
5	x	x		x	x		x		x
6		x		x		x			
7			x						
8	x				x	x			

Overall compatibility matrix									
	0	1	2	3	4	5	6	7	8
0					x	x			x
1						x	x		
2								x	
3						x	x		
4	x					x			x
5	x	x		x	x		x		x
6		x		x		x			
7			x						
8	x				x	x			

Obrázek 16: Matice kompatibilit pro sadu human_multiplex tak jak jsou zobrazeny programem MultiPCR.

B.2 Test sady str_database

Pro testování této sady jsme použili následující parametry vyhledávače: Rozsah délek ampikonů: 0–999, velikost řešení: 15, počet kanálů: 3.

Pro výpočet termálních parametrů jsme použili: Score: 3, množství soli: 0,085 mol, koncentrace vláken: 1 μ mol, teplota: 37°C, teplota tání: 55°C, ΔG° : -3 kcal/mol.

Mikrosatelity v sadě mají délky repetice 4–6. Nejvhodnější nalezené rozdělení do tří kanálů kanály bylo 5-5-5. Zároveň byla odhalena chemická nekompatibilita mezi mikrosatelity *gi938996gbG08446.1G0* a *gi938648gbG08098.1G0*, popis problému můžeme vidět na obr. 18. Tyto dva mikrosatelity nebyly do řešení zahrnuty, protože existovaly vhodnější kandidáti. Matice kompatibilit pro jejich větší rozměry neuvádíme. Mikrosatelity zařazené do výsledné sady jsou:

Name: gi4914674gbG42676.1G
Repetition length: 4
Allele list: 8 9 10 11 12 13 19

Name: gi938752gbG08202.1G0
Repetition length: 4
Allele list: 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Name: gi938648gbG08098.1G0
Repetition length: 4
Allele list: 8 9 10 11 12 13 14

Name: HUMUT1674Humanchromo
Repetition length: 4
Allele list: 14 16 17 18 19 20 21 22

Name: DYS434AC002992CTAT.m
Repetition length: 4
Allele list: 8 9 10 11

Name: gi296731embX71600.1H
Repetition length: 5
Allele list: 4 5 6 7 8 9 10 11 12

Name: gi939605gbG09055.1G0
Repetition length: 4
Allele list: 11 12 13 14 15

Name: gi939567gbG09017.1G0

Repetition length: 4

Allele list: 7 8 9 10 11 12 13 14 15

Name: gi604825gbL30761.1HU

Repetition length: 4

Allele list: 1 2 3 4 5 6 7

Name: gi31348embX06292.1HS

Repetition length: 4

Allele list: 8 9 10 11 12 13 14

Name: gi307519gbM68651.1HU

Repetition length: 4

Allele list: 5 6 7 8 9 10 11 12 13 14

Name: gi939370gbG08820.1G0

Repetition length: 4

Allele list: 9 10 11 12 13 14 15 16 17 18 19 20

Name: gi938475gbG07925.1G0

Repetition length: 4

Allele list: 5 8 9 10 11 12 13 14 15

Name: gi939290gbG08740.1G0

Repetition length: 4

Allele list: 5 6 7 8 9 10 11 12 13 14 15

Name: gi182293gbM21986.1HU

Repetition length: 4

Allele list: 3 4 5 6 7 8 9 10 11 12 13 14

B VÝSLEDKY TESTOVANÝCH SAD

83		gi938370gbG08820.1G0 R: 9.0	157			gi938475gbG07925.1G0 R: 11.0
84			158		gi31348embX06292.1HS R: 12.0	
85			159			
86			160	gi938752gbG08202.1G0 R: 15.0	gi604825gbL30761.1HU R: 3.0	
87		gi938370gbG08820.1G0 R: 10.0	161			gi938475gbG07925.1G0 R: 12.0
88			162	HUMUT1674Humachromo R: 14.0	gi31348embX06292.1HS R: 13.0	
89	gi4914674gbG42676.1G R: 8.0		163			
90			164	gi938752gbG08202.1G0 R: 16.0	gi604825gbL30761.1HU R: 4.0	
91		gi938370gbG08820.1G0 R: 11.0	165		gi604825gbL30761.1HU R: 5.0	gi938475gbG07925.1G0 R: 13.0
92			166		gi31348embX06292.1HS R: 14.0	
93	gi4914674gbG42676.1G R: 9.0		167	gi938752gbG08202.1G0 R: 17.0	gi604825gbL30761.1HU R: 6.0	
94			168		gi938567gbG09017.1G0 R: 8.0	gi938475gbG07925.1G0 R: 14.0
95		gi938370gbG08820.1G0 R: 12.0	169			
96	gi4914674gbG42676.1G R: 10.0		170	HUMUT1674Humachromo R: 16.0		
97			171			
98		gi296731embX71600.1H R: 4.0	172	gi938752gbG08202.1G0 R: 18.0	gi604825gbL30761.1HU R: 6.0	
99			173		gi938567gbG09017.1G0 R: 9.0	gi938475gbG07925.1G0 R: 15.0
100	gi4914674gbG42676.1G R: 11.0		174	HUMUT1674Humachromo R: 17.0		
101			175			
102		gi307519gbM68651.1HU R: 5.0	176	gi938752gbG08202.1G0 R: 19.0	gi604825gbL30761.1HU R: 7.0	
103		gi938370gbG08820.1G0 R: 14.0	177		gi938567gbG09017.1G0 R: 10.0	
104		gi296731embX71600.1H R: 5.0	178	HUMUT1674Humachromo R: 18.0		
105	gi4914674gbG42676.1G R: 12.0		179			gi182283gbM21986.1HU R: 3.0
106		gi307519gbM68651.1HU R: 6.0	180	gi938752gbG08202.1G0 R: 20.0	gi938567gbG09017.1G0 R: 11.0	
107		gi938370gbG08820.1G0 R: 15.0	181			
108			182	HUMUT1674Humachromo R: 19.0		
109	gi4914674gbG42676.1G R: 13.0	gi296731embX71600.1H R: 6.0	183			gi182283gbM21986.1HU R: 4.0
110	DYS434AC002992CTAT-m R: 8.0	gi307519gbM68651.1HU R: 7.0	184	gi938752gbG08202.1G0 R: 21.0		
111		gi938370gbG08820.1G0 R: 16.0	185		gi938567gbG09017.1G0 R: 12.0	
112			186	HUMUT1674Humachromo R: 20.0		
113	DYS434AC002992CTAT-m R: 9.0	gi296731embX71600.1H R: 7.0	187			gi182283gbM21986.1HU R: 5.0
114		gi307519gbM68651.1HU R: 8.0	188	gi938752gbG08202.1G0 R: 22.0	gi938567gbG09017.1G0 R: 13.0	gi938280gbG08740.1G0 R: 5.0
115		gi938370gbG08820.1G0 R: 17.0	189			
116			190	HUMUT1674Humachromo R: 21.0		gi182283gbM21986.1HU R: 6.0
117		gi307519gbM68651.1HU R: 9.0	191			
118	DYS434AC002992CTAT-m R: 10.0	gi938370gbG08820.1G0 R: 18.0	192	gi938752gbG08202.1G0 R: 23.0	gi938605gbG09055.1G0 R: 11.0	gi182283gbM21986.1HU R: 7.0
119		gi307519gbM68651.1HU R: 10.0	193		gi938567gbG09017.1G0 R: 14.0	gi938280gbG08740.1G0 R: 6.0
120		gi938370gbG08820.1G0 R: 19.0	194	HUMUT1674Humachromo R: 22.0		
121			195	gi938648gbG08098.1G0 R: 8.0		gi182283gbM21986.1HU R: 8.0
122	DYS434AC002992CTAT-m R: 11.0	gi938370gbG08820.1G0 R: 19.0	196	gi938752gbG08202.1G0 R: 24.0	gi938605gbG09055.1G0 R: 12.0	gi182283gbM21986.1HU R: 9.0
123			197		gi938567gbG09017.1G0 R: 15.0	gi938280gbG08740.1G0 R: 7.0
124		gi296731embX71600.1H R: 9.0	198			
125			199	gi938648gbG08098.1G0 R: 9.0		gi182283gbM21986.1HU R: 8.0
126		gi307519gbM68651.1HU R: 11.0	200	gi938752gbG08202.1G0 R: 25.0	gi938605gbG09055.1G0 R: 13.0	gi938280gbG08740.1G0 R: 8.0
127		gi938370gbG08820.1G0 R: 20.0	201			
128			202			gi182283gbM21986.1HU R: 9.0
129		gi296731embX71600.1H R: 10.0	203	gi938648gbG08098.1G0 R: 10.0	gi938605gbG09055.1G0 R: 14.0	gi938280gbG08740.1G0 R: 9.0
130		gi307519gbM68651.1HU R: 12.0	204	gi938752gbG08202.1G0 R: 26.0		gi182283gbM21986.1HU R: 10.0
131			205			gi938280gbG08740.1G0 R: 10.0
132			206			
133	gi4914674gbG42676.1G R: 19.0	gi938475gbG07925.1G0 R: 5.0	207	gi938648gbG08098.1G0 R: 11.0	gi938605gbG09055.1G0 R: 15.0	gi182283gbM21986.1HU R: 11.0
134		gi307519gbM68651.1HU R: 13.0	208	gi938752gbG08202.1G0 R: 27.0		gi938280gbG08740.1G0 R: 11.0
135			209			gi182283gbM21986.1HU R: 12.0
136			210			gi938280gbG08740.1G0 R: 12.0
137		gi307519gbM68651.1HU R: 14.0	211	gi938648gbG08098.1G0 R: 12.0		gi182283gbM21986.1HU R: 13.0
138			212	gi938752gbG08202.1G0 R: 28.0		gi938280gbG08740.1G0 R: 13.0
139		gi296731embX71600.1H R: 12.0	213			gi182283gbM21986.1HU R: 14.0
140			214			gi938280gbG08740.1G0 R: 14.0
141		gi31348embX06292.1HS R: 8.0	215	gi938648gbG08098.1G0 R: 13.0		
142			216			
143			217			
144			218			
145		gi938475gbG07925.1G0 R: 8.0	219	gi938648gbG08098.1G0 R: 14.0		
146	gi31348embX06292.1HS R: 9.0		220			
147			221			gi938280gbG08740.1G0 R: 14.0
148		gi938475gbG07925.1G0 R: 9.0	222			gi182283gbM21986.1HU R: 15.0
149			223			
150	gi31348embX06292.1HS R: 10.0		224			
151			225			
152	gi604825gbL30761.1HU R: 1.0	gi938475gbG07925.1G0 R: 10.0	226			
153			227			
154	gi31348embX06292.1HS R: 11.0		228			
155			229			
156	gi604825gbL30761.1HU R: 2.0					

Obrázek 17: Výsledné rozdělení sady str.database, alely délek 83–229 bp.

incompatible: gi938996gbG08446.1G0 [id: 15] vs gi938648gbG08098.1G0 [id: 2]
 Problematic primers: aacaggtcaaacctcctgtg VS ggggtgattttcctcttgggt
 $\Delta G^\circ = -3.64 \text{ kcal/mol}$ [under threshold]
 score = 6

Obrázek 18: Popis nalezené chemické nekompatibility tak jak ji zobrazuje aplikace MultiPCR.

Literatura

- [1] CLAVERIE, Jean-Michel, NOTREDAME, Cedric. *Bioinformatics For Dummies*. [s.l.] : For Dummies, 2006. 436 s. 2.
- [2] VALLONE, Peter M., BENIGHT, Albert S. Melting studies of short DNA hairpins containing the universal base 5-nitroindole. In *Nucleic Acids Research*. [s.l.] : Oxford University Press, 1999. Vol. 27, no. 17. s. 3589-3596. ISSN 0305-1048.
- [3] SANTALUCIA, JR., John. *A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics*. In Biochemistry. USA : Proc. Natl. Acad. Sci, 1998. Vol. 95, no 4. s. 1460-1465.
- [4] CHAVALI, Sreenivas, et al. *Oligonucleotide properties determination and primer designing : a critical examination of predictions*. In Bioinformatics : Original Paper. [s.l.] : Oxford University Press, 2005. Vol. 21, no. 20. s. 3918-3925.
- [5] VALLONE, Peter M., BUTTLER, John M. *AutoDimer: a screening tool for primer-dimer and hairpin structures*. In BioTechniques : Short Technical Reports. [s.l.] : [s.n.], 2004. Vol 37, no. 2. s. 226-231.
- [6] DAINTITH, John. *Oxford Dictionary of Physics*. [s.l.] : Oxford University Press, 2005. 586 s. 5. ISBN 0-19-280628-9
- [7] KITTEL, Charles. *Thermal physics*. USA : Twenty-first printing, 2000. 418 s. ISBN 0-7167-1088-9
- [8] SANTALUCIA, JR., John *Physical Principles and Visual-OMP Software for Optimal PCR Design*. In Methods in Molecular Biology : PCR Primer Design [s.l.] : [s.n.], 2007. Vol. 402. s. 3-34
- [9] CANTOR, C. R., SCHIMMEL, P. R. *Biophysical Chemistry Part III: The Behavior of Biological Macromolecules*. San Francisco : W. H. Freeman, 1980. 624 s. ISBN 0716711923
- [10] Record, M. T., Jr., Lohman, T. M. (1978) *Biopolymers* 17. s. 159–166
- [11] B. CHAIRES, Jonathan. Possible origin of differences between van't Hoff and calorimetric enthalpy estimates. In COOPER, A., DI CERA, E., WINTER, R. *Biophysical Chemistry*. [s.l.] : [s.n.], 1997. s. 15-23. ISSN 0301-4622

- [12] Steve Rozen, Helen J. Skaletsky (1998) *Primer3*. Dostupný z WWW: http://www-genome.wi.mit.edu/genome_software/other/primer3.html
- [13] *FastPCR* [online]. 2009, 30. 4. 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://www.biocenter.helsinki.fi/bi/Programs/fastpcr.htm>
- [14] *DNA* [online]. 2009, 16. 2. 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Dna>
- [15] *DNA* [online]. 2009, last modified on 7 March 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Dna>
- [16] *Gen* [online]. 2009, 24. 2. 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Gen>
- [17] *Gene* [online]. 2009, last modified on 7 March 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Gene>
- [18] *Allele* [online]. 2009, last modified on 6 March 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Allele>
- [19] *Alela* [online]. 2009, 5. 1. 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Alela>
- [20] *Elektroforéza* [online]. [2008], 2008-11-21 [cit. 2009-03-07]. Dostupný z WWW: <http://wapedia.mobi/cs/Elektroforéza>
- [21] *Agarose Gel Electrophoresis* [online]. c2007, last updated (Tuesday, 04 September 2007) [cit. 2009-03-07]. Dostupný z WWW: <http://alturl.com/imv>
- [22] *Agarose gel electrophoresis* [online]. 2009, last modified on 6 March 2009 [cit. 2009-03-07]. Dostupný z WWW: http://en.wikipedia.org/wiki/Agarose_gel_electrophoresis
- [23] *Restriction enzyme* [online]. c2009, last modified on 17 February 2009 [cit. 2009-03-07]. Dostupný z WWW: http://en.wikipedia.org/wiki/Restriction_enzyme
- [24] *Primer* [online]. 2008, 29. 10. 2008 [cit. 2009-03-07]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Primer>

- [25] *Primer (molecular biology)* [online]. 2009, last modified on 22 January 2009 [cit. 2009-03-07]. Dostupný z WWW: <[http://en.wikipedia.org/wiki/Primer_\(molecular_biology\)](http://en.wikipedia.org/wiki/Primer_(molecular_biology))>
- [26] *Ethidium bromide* [online]. 2009, last modified on 19 February 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Ethidium_bromide>
- [27] RACLAVSKÝ, Vladislav, MUDr., Ph.D. *Výhody a nevýhody přidávání ethidium bromidu do gelu* [online]. c2003 [cit. 2009-03-07]. Dostupný z WWW: <<http://biologie.upol.cz/metody/Slovník/Vyhody%20a%20nevahody%20EtBr%20v%20gelu.htm>>
- [28] *Polymerase chain reaction* [online]. 2009, last modified on 4 March 2009 [cit. 2009-03-07]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Polymerase_chain_reaction>
- [29] *Polymerázová řetězová reakce* [online]. 2008, 19. 12. 2008 [cit. 2009-03-07]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Polymerázová_řetězová_reakce>
- [30] *PowerPlex 16* [online]. c2009 [cit. 2009-04-21]. Dostupný z WWW: <<http://alturl.com/famr>>