Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

Diplomová práce

Omezení šumu signálu v genetických přepínačích

Plzeň, 2021 Bc. Lukáš Kuhajda

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta aplikovaných věd
Akademický rok: 2020/2021

ZADÁNÍ DIPLOMOVÉ PRÁCE
(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Bc. Lukáš KUHAJDA**
Osobní číslo:	**A19N0104P**
Studijní program:	**N3918 Aplikované vědy a informatika**
Studijní obor:	**Kybernetika a řídicí technika**
Téma práce:	**Omezení šumu signálu v genetických přepínačích**
Zadávající katedra:	**Katedra kybernetiky**

Zásady pro vypracování

1) Výchozí bod:
1a) Prozkoumání modelů a experimentálních výsledků v literatuře týkajících se genetických přepínačů.
1b) Popsání konkrétního designu genetického přepínače.
1c) Výzkum experimentálních výsledků pro potřebné biochemické prvky.
2) Analýza systému:
2a) Vypracování stochastického a deterministického modelu vybraného přepínače.
2b) Vytvoření instancí daných modelů s využitím relevantních hodnot pro příslušné parametry.
2c) Provedení analýzy systému. Vyhodnocení distribuce v ustáleném stavu, odhad spínacích časů a přibližné oblasti přitažlivosti. Vyvodit závěry týkající se fyzických omezení systému.
2d) Návrh experimentálních designů pro identifikaci modelu a jejich porovnání z hlediska praktické proveditelnosti a účinnosti.
3) Validace experimentů:
3a) Rozdělení daného systému na jednotlivé genové komponenty.
3b) Experimentální identifikace hodnot parametrů pro definované subsystémy použitím standardních experimentálních metod.
3c) Rekonstrukce genetického systému pomocí metod genetické rekombinace.
3d) Zavedení experimentálního návrhu pro automatizaci.
3e) Provedení navržených experimentů pomocí automatizované High Throughput Screening Platform.
3f) Identifikace modelu přepínače pomocí získaných experimentálních dat.

Rozsah diplomové práce: **40-50 stránek A4**
Rozsah grafických prací:
Forma zpracování diplomové práce: **tištěná**

Seznam doporučené literatury:

2012, Bistable responses in bacterial genetic networks: Designs and dynamical consequences (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3095517/)
2000, Construction of a genetic toggle switch in Escherichia coli (https://collinslab.mit.edu/files/gardner2001nature.pdf)
2017, Balancing a genetic toggle switch by real-time feedback control and periodic forcing (https://www.nature.com/articles/s41467-017-01498-0)

Vedoucí diplomové práce: **Doc. Daniel Georgiev, PhD.**
Katedra kybernetiky

Datum zadání diplomové práce: **1. října 2020**
Termín odevzdání diplomové práce: **24. května 2021**

L.S.

Doc. Dr. Ing. Vlasta Radová
děkanka

Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

V Plzni dne 1. října 2020

Prohlášení

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne .....

Poděkování

Tímto bych chtěl poděkovat panu Doc. M.Sc. et M.Sc. Danielu Georgievovi, Ph.D. za příkladné vedení mé diplomové práce, kdy mi byl vždy ochoten poskytnout odborné rady a cenné poznámky, které vedly k úspěšnému dokončení práce. Další mé poděkování patří panu Ing. Janu Švecovi, Ph.D., se kterým jsem mohl konzultovat vše v oblasti umělé inteligence. Poděkování patří také společnosti XENO Cell Inovations s.r.o. vedené panem Doc. M.Sc. et M.Sc. Danielem Georgievem, Ph.D. za umožnění vypracování laboratorních experimentů pro diplomovou práci a všem jejím zaměstnancům, kteří mi byli při procesu sestavování experimentu nápomocni.

Abstrakt

Pomocí neuronových sítí s nově navrženou strukturou jsou v diplomové práci nahrazeny nespolehlivé expertní metody pro vkládání regulačních míst do promotorů. Jedná se o první využití umělé inteligence k takové úpravě promotorů. Došlo k vytvoření nově strukturovaných modelů neuronových sítí. Pro vybraný model byl poté na základě výstupů pro testovací data vypracován experiment v laboratoři. Dle návrhu neuronové sítě vzniklo vkládáním regulačních míst do promotorů deset konstruktů. Téměř ve třetině případů bylo dosaženo znatelné regulace. S drobnou úpravou experimentu by však mohlo dojít k validaci další sady konstruktů. Odhadovaný výsledek by pak znamenal úspěšnost regulace u šedesáti procent návrhů.

Klíčová slova: biokybernetika, neuronové sítě, umělá inteligence, LSTM, syntetická biologie, genetické inženýrství, DNA, promotor, genetická regulace, Kvasinka pivní, Saccharomyces cerevisiae

Abstract

Using neural networks with a newly designed structure, the master thesis replaces unreliable expert methods for inserting regulatory sites into promoters. This is the first use of artificial intelligence to modify promoters in this way. Newly structured models of neural networks were created. Based on the outputs of the test data a laboratory experiment was conducted. According to the neural network design, ten constructs were created by inserting a regulatory site into the promoter. Significant regulation has been achieved in almost a third of cases. However, with a slight modification of the experiment, another set of constructs could be validated. The estimated result would then mean the success of regulation on sixty percent of proposals.

Keywords: biocybernetics, neural networks, artifical inteligence, LSTM, synthetic biology, genetic engineering, DNA, promoter, gene regulation, Brewer's yeast, Saccharomyces cerevisiae

Obsah

Kapitola 1

Úvod

Diplomová práce se věnuje propojení umělé inteligence s návrhem genetických úprav, kdy změnami v genomu má být dosaženo regulace procesů v organismech. Téma genetické regulace je zde spojeno s úpravami promotorů, které řídí tvorbu proteinů. V promotorech se mohou přirozeně vyskytovat vazebná místa interagující s regulačními proteiny. Při navázání regulačního proteinu na promotor dochází k ovlivnění exprese cílového proteinu. Díky znalosti tohoto mechanismu je možné navrhovat úpravy v promotoru tak, aby v něm vznikaly logické brány a v celém organismu pak složitější regulační obvody. Úpravy promotorů byly dosud expertní záležitostí, přičemž spolehlivost takovýchto zásahů není zpravidla stabilní. Tato práce formuluje problém vkládání regulačních míst do promotorů pro oblast strojového učení, k čemuž byly navrhnuty nové struktury neuronových sítí. Téma omezení šumu signálu v genetických přepínačích je zde zúženo na snahu o umístění represibilního místa do promotoru s vytvořením co největšího rozdílu ve stavech 'zapnuto'/'vypnuto'. Práce je rozdělena do tří hlavních bloků.

První část se věnuje teoretickým informacím potřebným pro vypracování a porozumění tématu diplomové práce. Jsou zde postupně vysvětleny a popsány potřebné termíny a mechanismy, které se v práci využívají. Následuje sekce zabývající se genetickou regulací se zaměřením na známé logické brány a motivy, které se vyskytují přírodně v organismech. Dále je pozornost věnována expertně navrženým regulačním obvodům popsaných v literatuře. Poté je v pořadí oblast soustředící se na expertní návrhy změn v DNA pro dosažení požadované regulační schopnosti v promotorech a náhled do problémů, které tyto metody přinášejí. Dále jsou představeny základní moduly, které byly využity pro trénování modelů neuronových sítí. Poslední část kapitoly je věnována metodám použitým pro experimentální ověření získaných výsledků z neuronových sítí.

V druhé kapitole jsou uvedeny inovace, které tato diplomová práce přináší. Zprvu jsou vypsány myšlenkové cesty vedoucí k možnému řešení vkládání regulačních sekvencí do promotorů. Následuje sekce s popisem dat, kde jsou charakterizovány vytvořené datasety využité pro trénování neuronových sítí a vzniklá databáze. Ta kombinuje několik veřejně dostupných databází se zaměřením na organismus druhu *Saccharomyces cerevisiae*, který byl využit pro laboratorní experimenty. Poté přichází sekce věnovaná neuronovým sítím a strukturám modelů využitých pro řešení úlohy.

Poslední kapitolou diplomové práce je analýza získaných výsledků. Nejprve se zde nachází seznámení s výstupy natrénovaných neuronových sítí pro popsané struktury a datasety z předchozí části práce. Spolu s tím je v sekci analýza promotorů a regulačních prvků, na kterých byly experimenty prováděny. Následuje popis použitých materiálů a metod, na základě kterých byl celý experiment vypracován. Dále jsou uvedeny získané experimentální výsledky z laboratoře. Nakonec dochází k diskuzi nad získanými daty ve spojení s využitými postupy.

Kapitola 2

Background

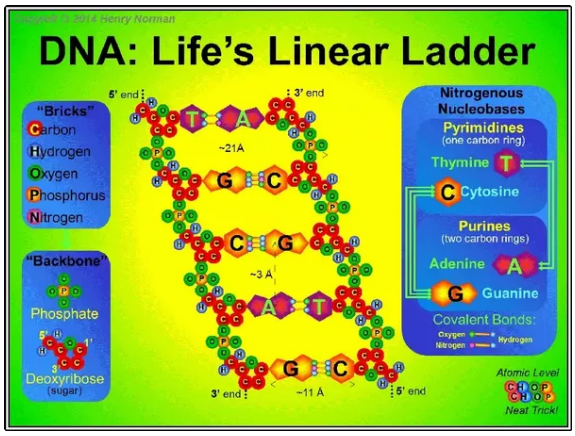
Tato kapitola má za cíl seznámit čtenáře s teoretickým pozadím pro praktickou část. V první sekci jsou nadefinovány biologické pojmy, které jsou následně používány v celém rozsahu práce. Spadá sem funkční popis mechanismů kolem DNA (transkripce na mRNA, translace na proteiny, apod.) a jednotlivé části DNA, které tyto mechanismy ovlivňují. Další v pořadí je sekce zabývající se genetickou regulací se zaměřením na motivy v regulačních sítích a jejich využití v praktických příkladech. Poté jsou popsány metody vkládání regulačních míst do promotorů a důvody, proč je zde snaha tyto dosud používané metody nahradit. Následuje sekce věnující se seznámení s moduly pro strojové učení, které byly v práci využity a nakonec jsou zde zmíněny metody použité pro sestavení experimentů a ověření výsledků.

2.1 Biologické pojmy

Sekce obsahuje vysvětlení biologických pojmů, které jsou v práci použity. Aby se termíny daly zpětně jednodušeji dohledat, jsou zvýrazněné tučně. Ve zbytku textu už poté nebudou nijak označeny.

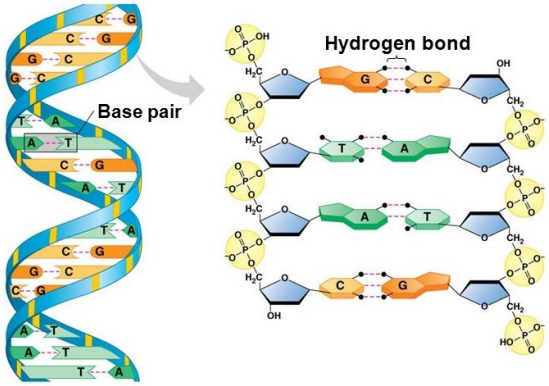
První skupina termínů se týká fyzické skladby deoxyribonukleové kyseliny (dále jako **DNA**), která ve většině případů nese celou genetickou informaci organismů a je umístěna v jádru buňky. Jedná se o dvojšroubovici[1] složenou ze čtyř typů **nukleových bází** - adenin (`A`), guanin (`G`), cytosin (`C`) a thymin (`T`) [1]. Každá nukleová báze obsahuje fosfátovou skupinu, přes kterou jsou báze spojeny v rámci jedné šroubovice (cukr-fosfátová kostra). Kostra vzniká propojením 5‘ uhlíku na jednom cukerném zbytku s 3‘ uhlíkem následujícího cukerného zbytku fosfodiesterovou vazbou. Dle tohoto řetězení se poté určují dva směry šroubovic (3‘→ 5‘, 5‘→ 3‘), přičemž ve dvojšroubovici má vždy protější šroubovice opačný směr (obr. 2.1) [2].

[1] Dvojšroubovice je složena ze dvou navzájem spletených jednovláknových šroubovic.



Obrázek 2.1: Vyobrazení 3‘→ 5‘ a 5‘→ 3‘ řetězců (z [3]).

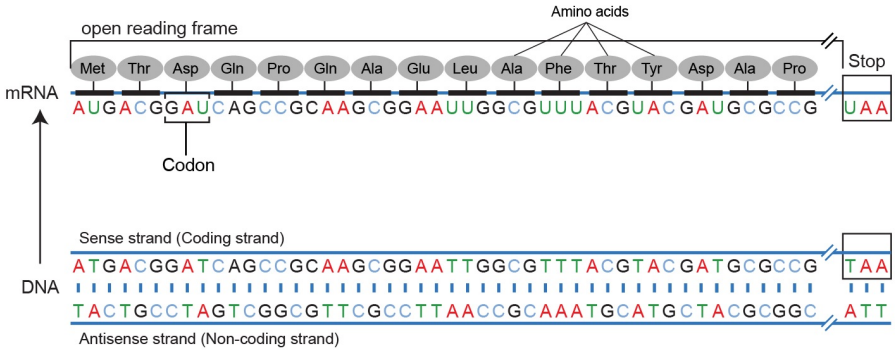
Spoje vzniklé mezi šroubovicemi jsou vazby vodíkové. Tyto spoje jsou omezeny na dvě možné pevně dané dvojice, jedná se o takzvanou **komplementaritu**: `A` = `T`, `C` ≡ `G` (obr. 2.2) [4]. Jak napovídá použité značení, vazby mezi dvojicemi jsou rozdílné. Adenin (`A`) se s thyminem (`T`) páruje přes dvě vodíkové vazby (=), zatímco guanin (`G`) s cytosinem (`C`) přes tři vodíkové vazby (≡). Spojení přes tři vodíkové vazby je silnější. V souvislosti pro danou úlohu se tak v promotorech vyhledávají potenciální změny spíše v oblastech, kde je nižší zastoupení dvojic `C` ≡ `G`. Obecně jsou promotory převážně složeny z nukleových bází `A`, `T` a oblastem s vyšší koncentrací `CG` je jako v článku (Hengge-Aronis 5) přiřazována funkční vlastnost, kterou je dobré v promotoru zachovat.



Obrázek 2.2: Zobrazení podoby dvojšroubovice DNA (vlevo), naznačení vazeb v DNA (vpravo) (z [6]).

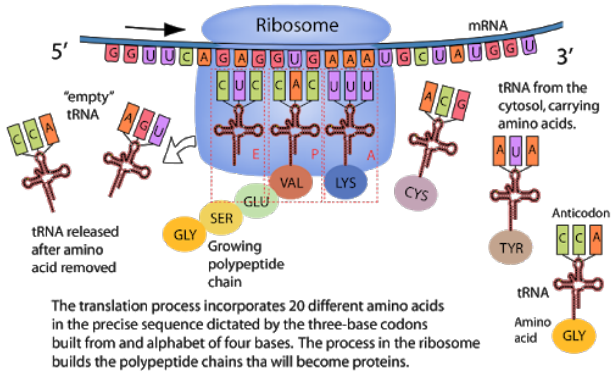
Další popsané termíny jsou také z oblasti fyzické skladby, jdou ale ruku v ruce s funkčními vlastnostmi DNA. Dojde zde tedy k jejich provázání. V DNA se nacházejí oblasti nazývané otevřené čtecí vzorce (v originálu open reading frames, dále bude používána zkratka ORF nebo termín **kódující sekvence**). **ORF** je úsek DNA, který

je transkribován na messenger RNA (dále **mRNA**). Při **transkripci** na DNA nasedlý enzym RNA polymeráza rozděluje DNA na dva oddělené řetězce a po směru cesty skládá volné nukleové báze na řetězec ve směru 3‘→ 5‘ s tím rozdílem, že thymin (`T`) je nahrazen uracilem (`U`). Řetězec mRNA je tím pádem jednovláknovou kopií šroubovice 5‘→ 3‘ v oblasti ORFu se zaměněným thyminem (`T`) za uracil (`U`) (obr. 2.3) [2].

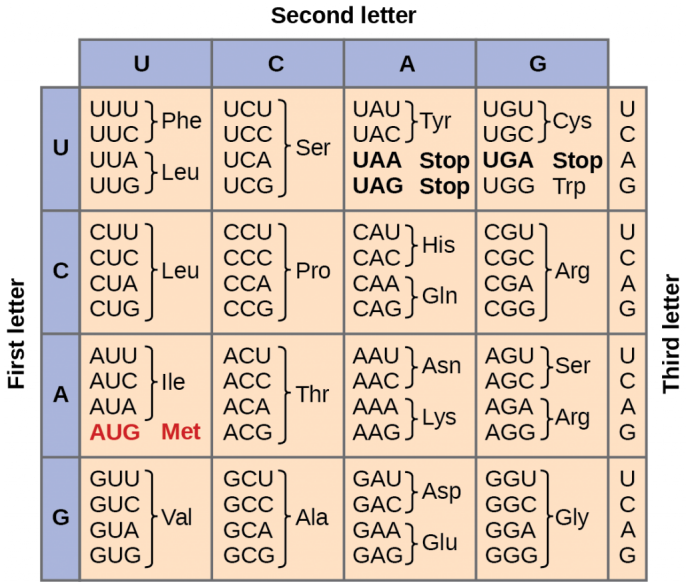


Obrázek 2.3: Ukázka transkripce DNA na mRNA s vyobrazeným rozsahem ORF a ilustrací, co je kodon a jak se skládají jednotlivé aminokyseliny do proteinů (z [7]).

Hranicemi ORFu jsou start a stop kodon. **Kodon** neboli nukleotidový triplet, je trojice nukleových bází určující aminokyselinu, do které se následně kodon překládá (translace) (obr. 2.5). Prvním kodonem ORFu je start kodon `AUG` odpovídající aminokyselině *methionine*. ORF končí před stop kodonem, který může mít podobu `UAA`, `UAG` nebo `UGA`. Ze sekvence bází odpovídajících ORFu se vytváří pomocí RNA polymerázy již zmíněná mRNA. Ta se následně dostává z jádra buňky do cytoplazmy a na základě posloupnosti kodonů se pomocí ribozomu překládá (**translace**) na řetězec z dvaceti typů aminokyselin, který se poté složí v cílový protein (obr. 2.4) [8].



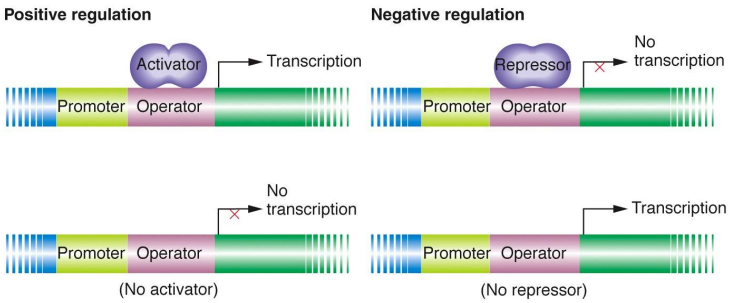
Obrázek 2.4: Ilustrace procesu translace mRNA na protein (z [9]).



Obrázek 2.5: Zobrazení genetické informace pro translaci každého kodonu v mRNA na aminokyselinu nebo terminační signál ve vznikajícím se proteinu (z [10]).

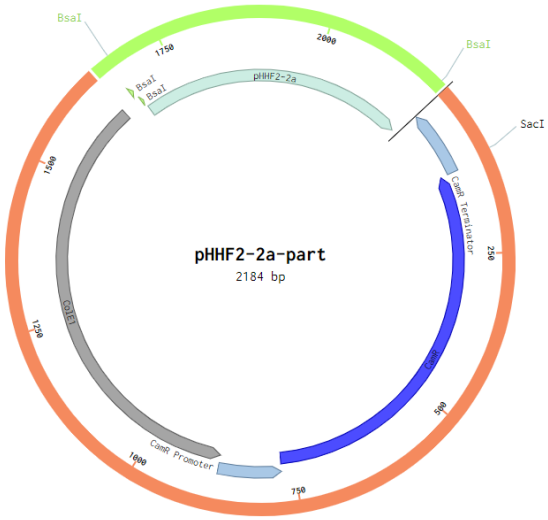
Výše jsou stručně shrnuty mechanismy vzniku proteinu. Tato práce je zaměřena především na fázi transkripce a na to, co jí předchází. ORF je obklopen promotorem a terminátorem a celý komplex promotor-ORF-terminátor se nazývá **gen**. Ve směru 5‘→ 3‘ se před ORFem nachází oblast zvaná **promotor**. Jedná se o sto až tisíc bází dlouhou sekvenci, kam nasedá RNA polymeráza a kde začíná transkripce. V eukaryotách RNA polymeráza nasedá na promotor v oblasti zvané **core promotor**, která v sobě kromě vazebné sekvence obsahuje počáteční místo transkripce (v originálu transcription start site, dále jako **TSS**) a může obsahovat TATA box. **TATA box** je sekvence bází pojmenovaná na základě opakujících se bází `T` a `A` [11]. Na TATA box se váže TATA-vazebný protein, který pomáhá umístit RNA polymerázu nad TSS [12]. Z hlediska funkčnosti existují různé typy promotorů: konstitutivní a regulované. **Konstitutivní promotory** jsou permanentně zapnuty a neustále tak dochází k transkripci kódující sekvence. Regulované promotory již mají dané závislosti, při kterých k transkripci dochází a při nesplnění požadavků je exprese genu vypnutá. Tyto regulační prvky se nazývají **transkripční faktory** a jedná se o aktivátory, nebo represory. U **aktivátorů** je pro expresi zapotřebí, aby byl regulační protein přichycen na odpovídající **vazebné místo** (neboli **operátor**) na promotoru. Naopak u **represorů** je pro transkripci třeba, aby regulační protein na promotoru přichycen nebyl, neboť transkripci přerušuje (obr. 2.6). **Terminátor** se poté ve směru 5‘→ 3‘ nachází za ORFem a slouží k ukončení transkripce.

Poslední skupinou pro definování jsou pojmy týkající se průběhu návrhu a reálného vytvoření genetických změn. Jedná se o primery, plazmidy, restrikční enzymy, overhangy, cutting-sites, DNA ligázu a part-plazmid. **Primer** je oligonukleotid (krátká syntetická jednovláknová sekvence DNA nebo RNA), který slouží k amplifikaci určité oblasti DNA (obr. 2.8) a zároveň může vytvářet genetické

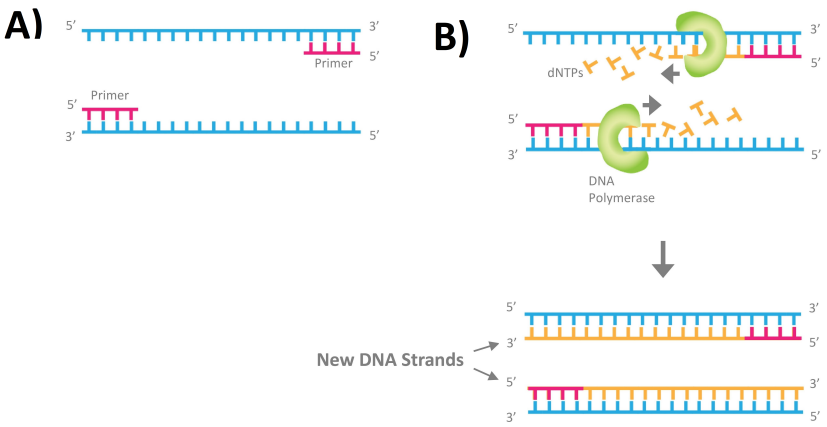


Obrázek 2.6: Znázornění kdy probíhá transkripce vzhledem k přítomnosti aktivátoru nebo represoru pro daný konstrukt (z [13]).

změny. **Plazmid** je menší DNA kruhovitého tvaru (obr. 2.7), jež se hojně využívá v syntetické biologii. Plazmid lze snadno dostat do bakterií, ve kterých se rychle namnoží tak, jak je a není třeba ho nijak integrovat do hlavní DNA dvojšroubovice. **Restrikční enzymy** jsou enzymy štěpící DNA na specifických místech (**cutting-sites**) svým specifickým způsobem, při kterém vznikají **overhangy**, které se vyznačují převisem jednoho z vláken dvojšroubovice (obr. 2.30). Spojení dvou overhangů poté obstarává **DNA ligáza**, která se snaží napravovat poškození DNA. **Part-plazmid** je plazmid obsahující informace potřebné pro namnožení v bakteriích a v ohraničeném úseku restrikčními místy nese část DNA, která má být dále použita například pro složení genu [14].



Obrázek 2.7: Ilustrativní podoba plazmidu v nástroji Benchling.



Obrázek 2.8: A) Ukázka vazby primeru na rozdělená vlákna DNA. B) Znázornění doplnění vlákna při procesu PCR (z [15]).

2.2 Genetická regulace

Lidské tělo se skládá z padesáti šesti typů buněk [16]. Genom jednotlivých buněk nese informaci o více než čtyřiceti šesti tisíci genech [17]. Buňky dle typu potřebují pro svoji funkčnost určité geny, které jiné typy buněk nepotřebují. Například mozkové buňky nemusí vytvářet hemoglobin a červené krvinky nepotřebují receptory acetylcholinu. Výroba genů stojí buňky energii. Některé geny jsou pro buňky dokonce toxické, je tedy třeba, aby byl gen exprimován pouze pokud je opravdu žádoucí. Regulace transkripce je tak nezbytná i pro jednobuněčné organismy [18].

Genetická regulace je soubor mechanismů, které řídí expresi genů na základě přítomnosti transkripčních faktorů. Gen, jehož promotor je regulován transkripčním faktorem, může sám být regulačním činitelem pro jiný gen. V buňkách tak vznikají složité sítě a kaskády navzájem se ovlivňujících faktorů [19]. Z časového hlediska je navázání transkripčního faktoru k náležícímu regionu DNA otázkou sekund, transkripce a následná translace zabere minuty a hromadění proteinového produktu v prostředí minuty až hodiny. Díky znalosti těchto mechanismů a parametrů jednotlivých prvků, které se na nich podílejí, je možné navrhovat obvody plnící stanovenou funkci. Přestože požadovaný obvod může být fyzikálně realizovatelný a jednoduše sestavitelný, v biologickém světe, kde se projevuje silná míra stochastiky, to může být aktuálně neřešitelná záležitost [20]. Dalším zásadním problémem je také limitovaný počet možných interakcí s buňkou, kdy pro návrh složitějších obvodů nemusí být dostatek vyhovujících prvků. Použití podobného automatického nástroje jako je vyvíjený zde, by tak mohlo zmíněnou bariéru odstranit.

V nadcházejících podsekcích jsou rozebrány genetické transkripční sítě, které si z evolučního hlediska našly v genomu svoji funkci a také experimentálně vytvořené obvody plnící různé úlohy.

2.2.1 Motivy a logické brány v transkripčních sítích

Interakce mezi transkripčními faktory a geny vytvářejí dohromady transkripční síť, která sleduje prostředí v buňce a okolí a produkuje potřebné proteiny s vysokou přesností. Na základě rozeznávání tepla, tlaku, signálů z okolních buněk, vnitřního poškození nebo přítomnosti prospěšných či toxických látek v okolí buňka vytváří transkripční faktory, které tak odpovídají stavové reprezentaci prostředí [19].

Transkripční síť popisuje všechny regulační transkripční interakce v buňce. Jejími prvky jsou **uzly** (geny), hrany a signály. **Hrany** představují regulaci genu regulačním proteinem a jejich typ odpovídá buď aktivaci nebo represi. Sítě obvykle obsahují větší množství aktivačních hran (60-80 %). Většina aktivátorů za určitých podmínek funguje také jako represor, což platí i naopak. Transkripční faktory mají tendenci používat stejný mód regulace pro všechny cílové geny, přičemž hrany vstupující do uzlu nebývají takto korelované. **Signály** jsou poté vstupy do sítě, molekuly přicházející z vně buňky, které modifikují protein a ovlivní tak jeho transkripční aktivitu. Například represor *TetR* po navázání *tetracyklinu* ztrácí schopnost represe cílového genu [21].

Míra regulace

Síla transkripčního faktoru ovlivňující míru transkripce cílového genu je popsána vstupní funkcí, která odpovídá Hillově funkci (vizualizace na obr. 2.9). Míra produkce cílového proteinu Y za jednotku času je funkcí transkripčního faktoru X:

$$Y = f(X^*), \tag{2.1}$$

kdy X^* je aktivní forma koncentrace X.

Vztah pro aktivátor je dán jako:

$$f(X^*) = \frac{\beta X^{*n}}{K^n + X^{*n}}, \tag{2.2}$$

kde K je aktivační koeficient, který se udává v koncentraci X nutné pro aktivaci exprese. Hodnota souvisí s afinitou transkripčního faktoru k vazebnému místu na DNA. β je maximální hodnota exprese ($X^* >> K$), kdy při vysoké koncentraci X^* se váže promotor s vysokou pravděpodobností a stimuluje RNA polymerázu k produkci velkého množství mRNA za jednotku času. Hillův koeficient n poté odpovídá potřebnému počtu transkripčních faktorů k navázání na cílový promotor a řídící tak šikmost vstupní funkce (obvykle hodnoty v intervalu mezi jednou a čtyřmi), viz obrázek 2.9.

Vztah pro represor je dán jako:

$$f(X^*) = \frac{\beta}{1 + (\frac{X^*}{K})^n}. \tag{2.3}$$

Zde mají parametry stejný význam jako pro aktivační funkci. V obou rovnicích se vyskytují tři proměnné, které jsou v organismech upravovány mutacemi transkripčního

faktoru a cílového promotoru.

Hillova funkce je vhodná pro detailní modely, pro zjednodušení je možné využít aproximace na skokovou funkci, kdy je gen buď zapnutý (ON: $f(X^*) = \beta$) nebo vypnutý (OFF: $f(X^*) = 0$) [22, 23]. Aktivační práh K poté určuje stav regulace.

Vztah pro aktivaci je tak dán jako:

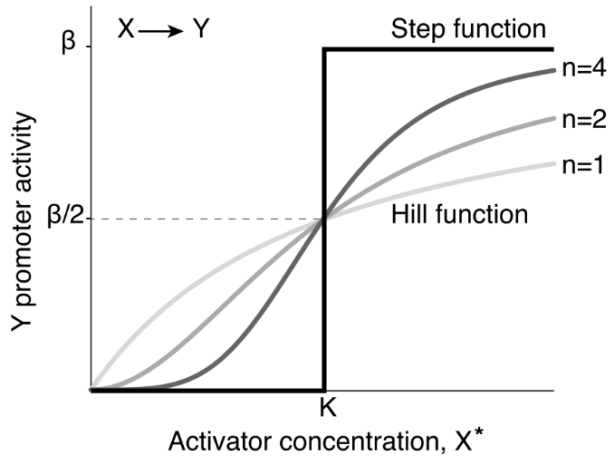
$$f(X^*) = \beta \cdot \theta(X^* > K), \tag{2.4}$$

kde θ je funkce nabývající hodnoty 1 při splnění vnitřní podmínky a hodnoty 0 při nesplnění.

Vztah pro represi je po aproximaci dán jako:

$$f(X^*) = \beta \cdot \theta(X^* < K), \tag{2.5}$$

kdy θ má stejný význam jako v rovnici 2.4.



Obrázek 2.9: Hillova funkce pro aktivátor s vyobrazeným vlivem hodnoty Hillova koeficientu (z [19]).

Logické brány

Mnoho uzlů má více než jednu vstupní hranu. Aktivita promotoru má poté vícerozměrnou vstupní funkci různých transkripčních faktorů [24, 25, 26]. Zde dochází k možnosti aproximace logickými funkcemi, například AND branou, OR branou nebo SUM.

Při funkci odpovídající AND bráně je třeba, aby všechny transkripční faktory regulovaly pozitivně (aktivátory na promotor navázané, represory nenavázané), výsledný vztah pro regulaci dvěma transkripčními faktory je ve tvaru:

$$f(X^*, Z^*) = \beta \cdot \theta(X^* > K_x) \cdot \theta(Z^* > K_z) \sim X^* \; AND \; Z^*, \tag{2.6}$$

pro OR bránu pak stačí pozitivní regulace pouze jednoho z transkripčních faktorů:

$$f(X^*, Z^*) = \beta \cdot \theta(X^* > K_x \; OR \; Z^* > K_z) \sim X^* \; OR \; Z^* \tag{2.7}$$

a při verzi SUM je funkce daná jako suma vstupů:

$$f(X^*, Z^*) = \beta_x X^* + \beta_z Z^*, \tag{2.8}$$

kde X a Z jsou vstupní regulační proteiny.

Chování vstupní logiky je poměrně citlivé a například několika mutacemi v *Lac* promotoru je možné ze základní vstupní funkce vytvořit AND nebo OR bránu [27]. Celkově jsou hrany a vstupní funkce pod neustálým tlakem přirozeného výběru. Nevyužívané hrany zanikají mutacemi, kdy pro zrušení hrany $X \to Y$ stačí změna jedné nebo několika bází v oblasti vazby na DNA [19].

Dynamika a časová odezva jednoduché regulace genu

V tomto případě se uvažuje regulace uzlu jednou hranou ($X \to Y$). Bez vstupu je X neaktivní a Y není produkováno. Při objevení signálu S_x dochází k okamžité přeměně X na transkripčně aktivní formu X^*, která se naváže na promotor genu Y a dochází k produkci cílového proteinu v konstantní míře β. K vyvážení produkce proteinu přispívají nové dva faktory a to degradace a ředění. **Degradace** je specifická destrukce proteinu a její míra je označována α_{deg}. **Ředění** je redukce koncentrace proteinu způsobená růstem buňky (označení α_{dil}). Výsledná míra degradace a ředění je v následujícím tvaru:

$$\alpha = \alpha_{deg} + \alpha_{dil}. \tag{2.9}$$

Změna koncentrace Y v čase je tedy:

$$\frac{dY}{dt} = \beta - \alpha Y, \tag{2.10}$$

kdy v ustáleném stavu (Y_{st}), tedy při splnění $dY/dt = 0$, Y_{st} odpovídá hodnotě:

$$Y_{st} = \beta/\alpha. \tag{2.11}$$

Při následném odstranění aktivačního signálu S_x dochází ke konci produkce proteinu ($\beta = 0$), jehož molekuly v prostředí postupně degradují:

$$Y(t) = Y_{st} e^{-\alpha t}. \tag{2.12}$$

Časová odezva se měří pomocí času dosažení poloviny počáteční a koncové úrovně v dynamickém procesu ($Y(t) = Y_{st}/2$):

$$T_{1/2} = log(2)/\alpha. \tag{2.13}$$

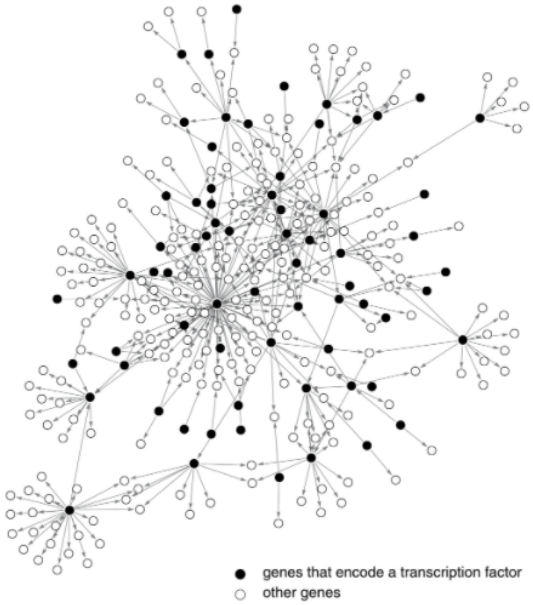
Stabilní proteiny nejsou aktivně degradovány v rostoucích buňkách a produkce je vyvažována ředěním způsobeným zvětšováním objemu rostoucí buňky. Pro tyto proteiny je tak $T_{1/2}$ rovno jedné generaci buněk. Před rozdělením doroste buňka dvojnásobného objemu a tím se koncentrace sníží na 50%:

$$T_{1/2} = log(2)/\alpha_{dil} = \tau, \tag{2.14}$$

kde τ je doba odpovídající jedné generaci buněk. Vzhledem k tomu, že bakterie mají generační čas třicet minut až několik hodin a eukaryoty ještě více, doba odezvy tak může být limitujícím prvkem tvorby efektivních regulačních obvodů [19].

Síťové motivy

Při navrhování genetických obvodů může být v buňce požadovaný typ funkce již přítomný a je možné se jím tedy inspirovat. V transkripční síti se nachází nepřeberné množství **vzorů**, které jsou tvořeny propojenými uzly pomocí hran (viz obr. 2.10). Mezi vzory se poté hledají ty důležité, které se nazývají **síťové motivy**. Jako síťové motivy jsou označeny vzory vyskytující se výrazně častěji v reálných transkripčních sítích, než v náhodně vygenerovaných[2] [28, 29]. V následujících odstavcích budou popsány vybrané základní motivy smyslových sítí.



Obrázek 2.10: Ukázka propojení uzlů v transkripční síti (z [19]).

Pro náhodnou transkripční síť je využíván model Erdos & Renyi, kde je generován stejný počet hran a uzlů [30]. Počet možných hran je tak:

$$N(N-1) + N = N^2 \tag{2.15}$$

a pravděpodobnost hrany je:

$$p = E/N^2. \tag{2.16}$$

Motivy smyslových sítí

Motivy nacházející se ve smyslových sítích mají obecně vlastnost rychlé reakce. Jedním z těchto motivů vypozorovaných v transkripční síti bakterie *Escherichia coli* (dále zkráceně *E.coli*) je autoregulační smyčka, kdy produkce genu ovlivňuje expresi sama sebe. Z hlediska pravděpodobnosti by v náhodné síti měl být daný motiv jednou až dvakrát, přičemž v transkripční síti *E.coli* se vykytuje 40krát. Ze čtyřiceti výskytů je poté třicet dva případů regulace negativní, která má schopnost urychlovat

[2]Náhodné transkripční sítě se generují se stejným počtem uzlů a hran.

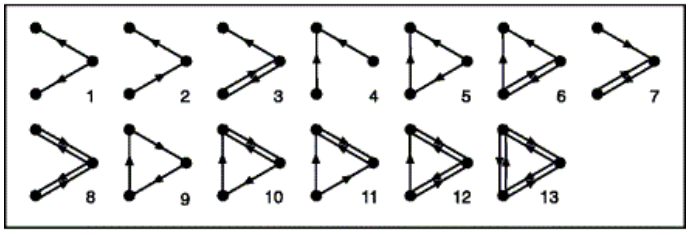
odezvu systému. Ve chvíli, kdy se začne gen exprimovat, koncentrace kódovaného proteinu se zvyšuje a tím represuje vlastní exprimaci vazbou na promotor. Dynamika produkce proteinu je popsána rovnicí:

$$T_{1/2} = \frac{K}{2\beta}, \tag{2.17}$$

kde β je nerepresovaná aktivita a K je koeficient represe. Při negativní autoregulaci pomocí silného promotoru se může rychle nabýt vysoká produkce, která se vzápětí sama zastaví. Samotné parametry jsou pak evolucí jednoduše nastavitelné, kdy K je možno upravit například mutací v místě vázání transkripčního faktoru na promotor a parametr β mutacemi v místě vázání RNA polymerázy na promotor. Negativní autoregulace je tedy vhodná v případech, kdy je třeba rychlá produkce proteinu po omezený čas.

Opačným případem je pozitivní autoregulace, kdy translatovaný protein aktivuje a zesiluje vlastní produkci. Toto nastavení má pomalou dynamiku a je vhodné pro procesy trvající dlouhou dobu, například procesy vývojové. Při aktivaci gen zůstává aktivní i po zmizení původního spouštěcího signálu [31, 32]. Jedná se tak o jednoduchou paměť, která má využití ve vývojových transkripčních sítích, kde se stanovuje osud buňky (například rozhodnutí o typu tkáně, které bude buňka součástí).

Poslední zaměření je na vzory se třemi uzly. Ze všech třinácti možných kombinací (viz obr. 2.11) je motivem pouze jeden. Jedná se o dopřednou smyčku (v originálu feed-forward loop, dále zkráceně jako **FFL**), kdy se hrana nevrací do počátečního uzlu. Výskyt trojúhelníkových vzorů je v náhodných sítích velmi vzácný, přičemž v *E.coli* se dopředná smyčka nachází 42krát. Při kombinaci všech hran s možnostmi, že každá může být aktivační, nebo represibilní, existuje celkem osm možností, jak se kombinace hran v motivu poskládají. Z toho se v buňce vyskytují pouze dva typy častěji: verze se všemi aktivačními hranami a verze s represibilní hranou odpovídající spojení genů Y a Z (viz obr. 2.12-A,B).

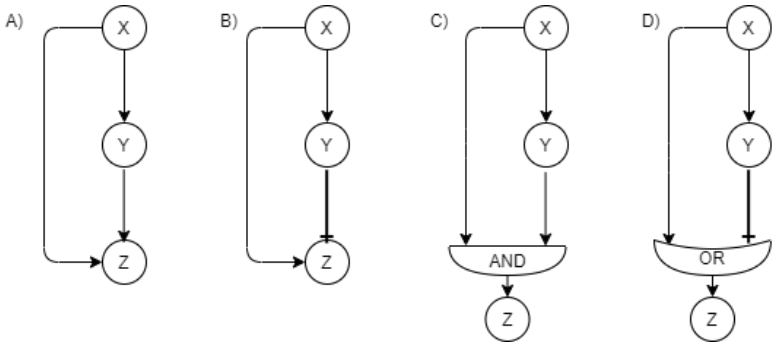


Obrázek 2.11: Možné kombinace propojení tří uzlů. Pouze pátý vzor je motivem v *E.coli*.

První zmíněný typ FFL (obr. 2.12-A) je koherentní, neboť nepřímá cesta má stejné znaménko jako přímá (dále motiv označován jako **C1-FFL**). Motiv C1-FFL s AND branou na vstupu do genu Z vytváří spínací zpožďovací mechanismus (obrázek 2.12-C). Vstupní signál S_x aktivuje $X \rightarrow X^*$ a tím se aktivuje produkce Y. K aktivaci exprese proteinu Z je pak zároveň zapotřebí překročení aktivačního

prahu K_{yz}, kdy se čeká na nahromadění aktivní formy genu Y (Y^*). Po přerušení vstupního signálu S_x se zastaví produkce X a celý proces tedy končí. Zpoždění tak v C1-FFL vzniká pouze při zapínání Z, nikoliv při vypínání, které je bez odezvy. Tento mechanismus zastává v buňce spíše ochranné funkce a zabraňuje zbytečnému kolísání filtrováním krátkých vstupních fluktuací.

Druhým motivem FFL je nekoherentní smyčka (dále **I1-FFL**) s opačným znaménkem nepřímé cesty oproti přímé (obr. 2.12-B). S OR logickou bránou na vstupu do uzlu Z vzniká mechanismus pulsního generátoru (obr. 2.12-D). Při vstupním signálu S_x se produkuje X^*, které aktivuje Y a Z. Zde stačí jeden pozitivní vstup do Z a dochází tak k jeho aktivaci. Zastavení exprese nastává bez odezvy při zmizení vstupního signálu S_x nebo po nahromadění Y^*, které represuje Z. Tato sestava tak s přítomným S_x exprimuje Z jen po určitou dobu a vytváří tak pulsní systém generování cílového proteinu.



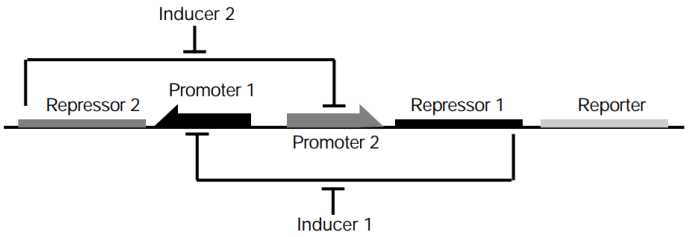
Obrázek 2.12: Dva typy dopředných smyček jako motivy v *E.coli*. A) Smyčka obsahující pouze pozitivní regulaci (C1-FFL). B) Smyčka obsahující represi mezi geny Y a Z (I1-FFL). C) C1-FFL smyčka s AND branou na vstupu do Z. D) I1-FFL smyčka s OR branou na vstupu do Z.

2.2.2 Expertně navržené regulační obvody

Předchozí podsekce 2.2.1 se zabývala regulačními obvody, které se přirozeně nacházejí v živých organismech a které naznačují možnosti regulačních obvodů pro umělé vkládaní do genetického kódu. V následujících odstavcích budou popsány reálné publikované aplikace uměle vytvořených genetických obvodů v jednobuněčných organismech.

Prvním obvodem pro seznámení je syntetický genetický přepínač konstruovaný v organismu E.coli (schéma zapojení viz obr. 2.13) [33]. Práce vycházela z tvrzení, že regulační obvody s prakticky jakoukoliv požadovanou vlastností mohou být konstruovány spojením jednoduchých regulačních prvků [34]. Podkladem pro vypracování bylo nalezení multi-stability a oscilací v genetických obvodech *bakteriofágu* λ a v sinicích [35, 36]. Vytvořený obvod (viz obr. 2.13) obsahuje dva vzájemně se represující geny. Transkribované proteiny jsou v použitém konstruktu *LacI* a *TetR*. Jedná se o dobře prostudované transkripční faktory, pro které jsou známé i jejich induk-

tory. **Induktor** je látka vázající se na cílový protein a zabraňující danému proteinu v nasednutí na vazebné místo v promotoru. Induktory se mohou přírodně vyskytovat v buňce nebo, pro dobře prozkoumané případy jako *LacI* a *TetR*, existují známé analogy napodobující funkci přirozených induktorů, které navíc nejsou buňkou samotnou nijak metabolizovány. V případě *LacI* je odpovídající induktor *Isopropyl β-D-1-thiogalactopyranoside* (zkráceně **IPTG**) a pro TetR se jedná o *anhydrotetracycline* (zkráceně **aTc**). Bez přítomnosti induktorů má daný obvod dva stabilní ustálené stavy, kdy je aktivní buď jeden, nebo druhý gen. V případě přidávání induktorů pak může dojít k narušení této stability a vytvoření pokročilejších funkcí obvodu [20]. Bi-stabilita byla v minulosti celkově studována s velkým zájmem, kdy se ve zkoumaných organismech nalezlo mnoho odpovídajících genetických obvodů s důležitými funkcemi [37].

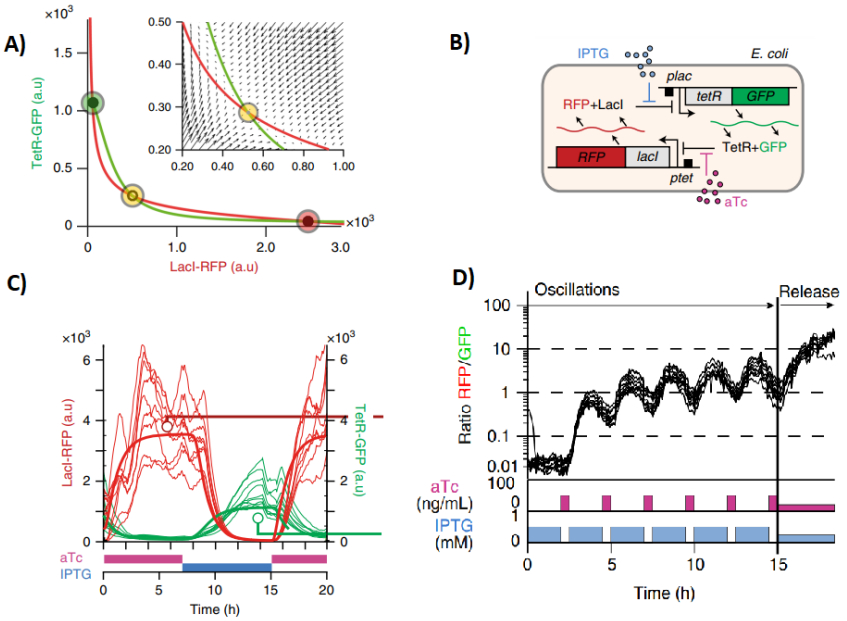


Obrázek 2.13: Genetický přepínač vytvořený v článku (Gardner et al. [33]).

Na popsaný přepínač z (Gardner et al. [33]) navázala studie zabývající se složitějším řízením biologického systému [20]. Do té doby sloužila regulace pouze k udržování exprese na konstantní úrovni [38]. Použité řízení udržující bistabilní genetický obvod v nestabilním rovnovážném stavu (obr. 2.14-A) je ekvivalentní s řízením inverzního kyvadla. Bistabilní obvody vykazující hysterezi se vyznačují specifickými problémy řízení, jedná se tak o ideální demonstraci pro použití kybernetiky [39]. Genetický obvod byl využit v podobné sestavě jako v (Gardner et al. [33]) s tím rozdílem, že zde se produkují dva signalizační proteiny místo jednoho (viz obr. 2.14-B). Za běžných okolností se obvod dostane do jednoho ze dvou rovnovážných stavů, ve kterém pak i přes perturbace zůstane. Laděním koncentrací propustných molekul IPTG a aTc se s touto základní funkčností dá pracovat na pokročilejší úrovni. K udržení obvodu mezi atraktory (dva stabilní stavy) je však třeba dynamické řízení, které je v biologii v real-time aplikacích velice náročné. Pro udržení jedné buňky v nestabilním stavu byl nejdříve použit PI regulátor, což pro reálnou aplikaci bylo zbytečně komplikované. Přešlo se tedy na takzvaný bang-bang regulátor, kdy se regulátorem aplikuje vždy maximální nebo minimální koncentrace induktoru na základě rozdílu cílové od pozorované fluorescence[3]. Tímto přístupem bylo stabilně dosaženo výborných výsledků (obr. 2.14-C) a potvrdila se tak možnost řídit buňku mimo stabilní stavy. Problémem však bylo omezení pouze

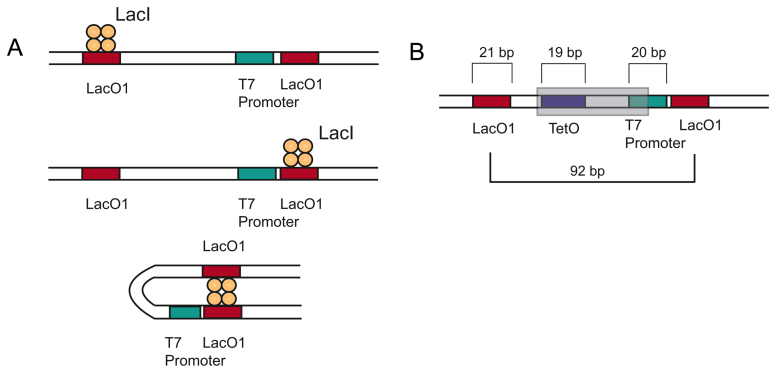
[3]Fluorescence je typ záření, které je vyvoláno účinkem dopadajícího záření nebo částic

na jednu buňku. Pro regulaci více buněk byl aplikován analogický princip s více kyvadly s rozdílnou vahou, který byl uřízen jednou mechanickou silou [40]. Při nalezení správné periody změn koncentrace IPTG a aTc poté došlo k udržení většího počtu buněk v nestabilním stavu najednou (obr. 2.14-D).

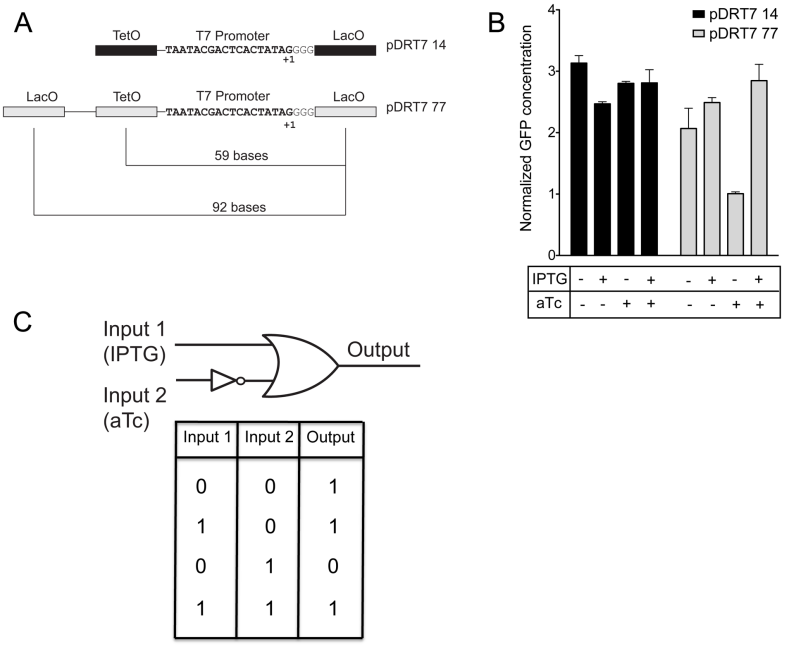


Obrázek 2.14: A) Ustálené stavy bistabilního obvodu, dva stabilní (červený, modrý), jeden nestabilní (žlutý). B) Schéma genetického obvodu použitého pro vytvoření oscilátoru. C) Výsledek řízení jedné buňky. D) Výsledek řízení většího počtu buněk periodickým buzením. (všechny obrázky z [20]).

Další popisovanou expertní úpravou je vytvoření logické brány IMPLIES pomocí *T7* promotoru a operátorů pro *LacI* a *TetR* [41]. *T7* promotor je sekvence osmnácti bází (5' – `TAATACGACTCACTATAG` – 3'), kterou rozpoznává *T7* RNA polymeráza a která nabízí jednoduchý ortogonální systém exprese pro použití v různých buněčných a i bezbuněčných systémech [42]. Transkripce navazující kódující sekvence může být potlačována transkripčními faktory *LacI* a *TetR*, což umožňuje tvorbu logických bran [43]. Podoba sestaveného promotoru se nachází na obrázku 2.15. Protein *LacI* se vyskytuje v buňce jako dimer dimerů (tetramer), kdy každý dimer se může navázat na jeden *Lac* operátor. Při nízké koncentraci *LacI* je tedy velká možnost, že se jeden tetramer naváže na oba vložené operátory *LacI* a vytvoří tak na DNA smyčku represující transkripci (obr. 2.15-A) [44]. Při správném vložení *Tet* operátoru poté dochází k potlačení smyčky, kdy přítomný aktivní *TetR* narušuje represi způsobenou aktivním transkripčním faktorem *LacI* a vytváří tak IMPLIES bránu: IPTG OR (NOT aTc). Výsledky pro danou logickou bránu jsou znázorněny na obrázku 2.16-B v pravé části.



Obrázek 2.15: A) Vizualizace vytvoření smyčky na DNA pomocí tetrameru *LacI*. B) Názorná ukázka umístění *Tet* operátoru mezi *Lac* operátory (z [41]).



Obrázek 2.16: A) Experimentální umístění operátorů pro *LacI* a *TetR*. B) Výsledky měření daných sestav s přidáváním induktorů IPTG a aTc. C) Logická brána IMPLIES vytvořená v sestavě s *LacI* a *TetR* (z [41]).

Poslední expertní přístup tvorby genetických obvodů, který bude zmíněn, je iniciativa snažící se o automatickou tvorbu regulačních sítí [45]. Byly vytvořeny brány pro druh *Saccharomyces c.* založené na minimálních (core) konstitutivních promotorech (cca sto dvacet párů bází), pro které byla vytvořena pravidla pro vkládání operátorů. Takto bylo vytvořeno devět NOT/NOR bran, na základě kterých byly pomocí programu Cello 2.0 vytvořeny obvody s až jedenácti regulačními proteiny. Jedná se o nástroj zjednodušující konstrukce regulačních sítí v eukaryotách.

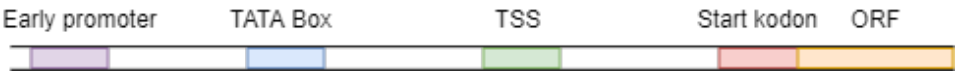
Nevýhodou však je, že pro použití je třeba využívat pevně danou knihovnu jednotlivých prvků. I tak se ale jedná o velice pokročilý a revoluční nástroj, který lze snadno využít pro návrh genetického obvodu s vlastní specifikovanou funkcí.

2.3 Expertní metody pro genetickou regulaci

Tématem vkládání regulačních míst do promotorů se zabývá celá řada publikací. Od prvních studií se však stále dodržují stejné postupy. Mezi prvními studiemi byl výzkum, který do promotorů vkládá *Lac* operátor [46]. Transkripční faktor *LacI* se ve studiích objevuje již od roku 1942 a je tedy do hloubky prozkoumán [47]. Dalším hojně používaným a také dobře prostudovaným transkripčním faktorem je *TetR*. Bližší seznámení s oběma proteiny je v sekci 4.1.3, neboť byly využity pro experimenty i v této diplomové práci. V následujících odstavcích bude pozornost věnována expertním přístupům pro vkládání operátorů do promotorů, jejich mutacím a mutacím v transkripčních faktorech s cílem dosažení co nejlepší genetické regulace.

Prvními pokusy o genetickou kontrolu byla snaha regulovat indukovatelné eukaryotické promotory. Jednalo se o reakce na ionty těžkých kovů (Mayo et al. [48]), tepelný šok (Nover [49]) a regulaci pomocí hormonů (Hynes et al. [50]). V následujících letech pak byla pozornost upřena na integraci *Lac* operátoru do promotorů.

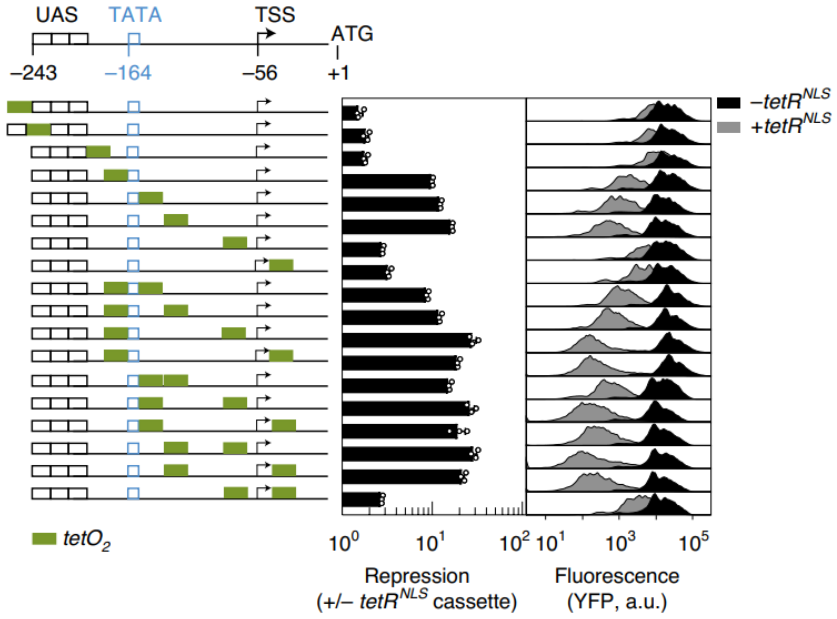
První vybraný článek je z roku 1987 (Hu [51]). Umisťování *Lac* operátoru zde probíhalo vkládáním operátoru mezi důležité oblasti v promotoru (oblasti zobrazeny na obr. 2.17). Úpravy promotorů probíhaly do dvou mateřských plazmidů, do kterých byly vždy vkládány jeden až tři operátory za sebou. Vkládání probíhalo do oblasti mezi SV40 early-promotor[4] a TATA-box, mezi TATA-box a TSS a mezi TSS a start kodon. Ze získaných výsledků dané studie vyplývá, že umístění jednoho až dvou operátorů před TATA-box vytváří silnou represi (80-90 %). Při umisťování operátorů mezi TATA-box a TSS je pro silnou represi potřeba dvou až tří operátorů (represe 70-89 %), kdy samostatný operátor v této oblasti represuje aktivitu v průměru jen o 25 %. Poslední možnost, tedy vložení operátorů mezi TSS a start kodon, je také silně závislá na insertovaném počtu operátorů. Při integraci jednoho má represe účinnost necelých 50 %, s počtem tří operátorů je však represe až 98 %, přičemž s vložením čtyř operátorů za sebou již míra represe opět klesá a to na necelých 70 %.



Obrázek 2.17: Posloupnost důležitých oblastí v promotoru vytvářející mezi sebou místa, která byla v (Hu [51]) využita pro vkládání regulačních míst.

[4]V případě, že gen má více promotorů a kódujících sekvencí, early-promotor je prvním promotorem genu, který náleží první kódující sekvenci [52].

Podobný přístup byl zvolen i v sekci 2.2 probíraném článku (Chen et al. [45]), kde knihovna jednotlivých částí pro skládání regulačních obvodů byla vytvářena různorodým vkládáním operátorů do promotorů (viz obr.2.18).



Obrázek 2.18: Porovnání umístění operátorů a případně jejich rozestupů s mírou represe, kterou jednotlivé konstrukty produkují (z [45]).

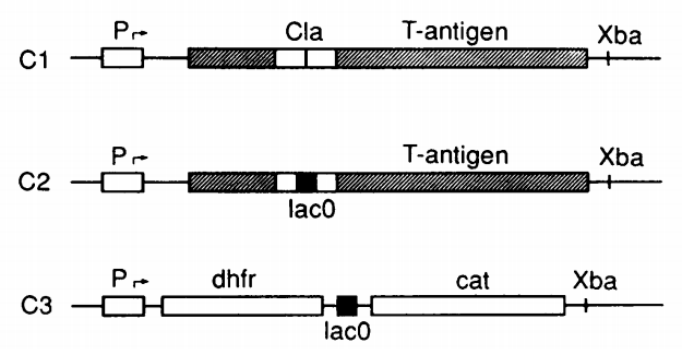
V dalším vybraném článku byl zkoumán vliv umístění *Lac* operátoru v těsné blízkosti kolem TATA-boxu v promotoru viru *Vaccinia* [53]. Výsledky jsou zde vztaženy k sestavě, kdy promotor končí TATA-boxem částečně obsahujícím již start kodon (obr. 2.19-první řádek tabulky). Vložení *Lac* operátoru pak bylo provedeno za TATA-box, za TATA-box s částečným přepsáním TATA-boxu a před TATA-box (obr. 2.19). Získané výsledky ukazují, že vložením operátoru do vybraných pozic je způsobena znatelná změna ve fungování promotoru. Umístěním operátoru za TATA-box byla snížena přirozená síla promotoru o 34 %. S přítomností represoru *LacI* pak došlo v podstatě k úplnému vypnutí a s přidáním induktoru IPTG opět k produkci cílového proteinu. Při částečném přepsání TATA-boxu operátorem došlo k citelnému narušení promotoru, kdy přirozená exprese dosahovala síly pouze jedné pětiny oproti nezměněnému promotoru. Systém s přítomným represorem a induktorem fungoval stejným způsobem jako u předchozí verze umístění, pouze s nižší produkcí cílového proteinu po přidání induktoru. Poslední verzí bylo umístění operátoru před TATA-box. Tento přístup však úplně rozbil promotor, který způsobenou změnou přestal fungovat.

Předchozí dva zmíněné články se zaměřovaly na zabránění transkripce před jejím započetím [50, 53]. Další vybraná studie se však zabývala blokací již transkribující RNA polymerázy (Deuschle et al. [54]). Na obrázku 2.20 je naznačené

vložení *Lac* operátoru do kódující sekvence v konstruktech C2 a C3. Na základě přítomnosti induktoru IPTG pak byly pozorovány změny spojené s represí v oblasti operátoru. V přítomnosti induktoru docházelo k transkripci celého ORFu, zatímco v nepřítomnosti IPTG byla transkripce v místě operátoru zastavována.

Plasmids	Position of Lac-Operator Insertion	% Expression Virus: WT	vlacI	vlacI
		IPTG: -	-	+
pSC11	P11 - TAAATG - lacZ	100	100	100
placOZ-1	P11 - TAAATA GAATTGTGAGC:GCTCACAATTC TCGAGCATG - lacZ	66	<0.1	8.3
placOZ-2	P11 - TA AATTGTGAGC:GCTCACAATT TCGAGCATG - lacZ	20	<0.1	1.4
placOZ-3	P11 - GAATTGTGAGC:GCTCACAATTC TAAATACTCGAGCATG - lacZ	<0.1	<0.1	<0.1

Obrázek 2.19: Tabulka s výsledky umisťování *Lac* operátoru do promotoru viru *Vaccinia*. WT: wild-type virus, bez přítomnosti represoru. vlacI: měření s přítomností represoru *LacI*



Obrázek 2.20: Konstrukty C2 a C3 obsahující *Lac* operátor uvnitř kódující sekvence (z [54]).

V již probíraném článku (Iyer et al. [41]) v sekci 2.2 využili pro vytvoření logické brány IMPLIES řadu otestovaných sestav z jiných publikací. Z článku (Dubendorf and Studier [43]) byla použita vzdálenost 238 bází *Lac* operátoru od pomocného *Lac* operátoru k vytvoření co nejsilnější represibilní smyčky po navázání jednoho tetrameru *LacI*. Vzdálenost 238 bází se nakonec projevila jako příliš velká, neboť se ukázalo, že pro vytvoření silné smyčky je vhodná kratší vzdálenost [55]. V (Müller et al. [56]) hledali optimální vzdálenost *Lac* operátoru s pomocným *Lac* operátorem vzdálených od sebe v intervalu od 57 do 1493 bází. Vyšla z toho lokální maxima pro vzdálenost operátorů 70, 92 a 115 bází. V dané studii (Iyer et al. [41])

byla nakonec na základě testů zvolena vzdálenost 92 bází. Pro následné vložení *Tet* operátoru bylo zvoleno místo mezi oběma operátory pro *LacI*. Nejprve byly testovány kratší vzdálenosti od hlavního *Lac* operátoru, a to 21, 23, 25 a 27 bází, což nedostatečně ovlivňovalo represi působenou *Lac* operátory. Cílené chování brány IMPLIES bylo následně dosaženo umístěním *Tet* operátoru 59 bází od hlavního *Lac* operátoru.

2.3.1 Nespolehlivost

Allosterická[5] regulace se vyskytuje ve všech oblastech života, stále však chybí jednoduché prediktivní teorie, které přímo spojují experimentálně laditelné parametry systému s jeho odezvou výstupu na vstup [57]. To vyvolává obavu, že každá regulační architektura může vyžadovat jedinečnou analýzu, kterou nelze přenést na jiné systémy.

Všechny expertní metody a aplikace zmiňované v sekci 2.3 dosahovaly nakonec žádaných výsledků. Problémem však zůstává zmíněná nepřenositelnost získaných závěrů na jiné typy konstruktů. Výsledky jsou často vztažené k určitému promotoru (jako například v (Iyer et al. [41])). Zjištěné optimum může alespoň posloužit jako počáteční odhad při jeho záměně. Stejný problém se týká operátorů. Pokud ve vyladěném konstruktu funguje regulátor dle představ, záměnou operátoru za jiný typ (například *Tet* za *Lac*) může být získaná vysoká kvalita regulace ztracena.

Promotory jsou obecně velice citlivé na jakékoliv změny, v případě přemisťování operátorů stačí posun o několik bází a regulace může být násobně horší. Bodové mutace v operátorech nebo v důležitých oblastech transkripčních faktorů mohou také vést k výrazným změnám [58]. Dosud používané metody pro hledání ideálního nastavení operátorů vůči zbytku promotoru jsou často pouze lokálním optimem, kdy globální optimum může být lidskému vnímání skryto a náhodnými pokusy nenalezeno.

Takovéto prohledávání celého stavového prostoru je náročné jak finančně, tak časově. Vytváření nástrojů pomocí strojového učení, jako v této diplomové práci, by tak mohlo být cestou k nahrazení zdlouhavých procedur rychle získaným návrhem pro umístění regulačních míst do libovolných promotorů.

2.4 Strojové učení

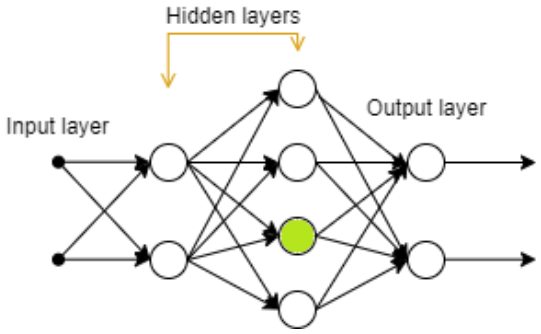
Strojové učení by mohlo výrazně zrychlit a celkově zefektivnit umisťování regulačních míst do promotorů a tím tak otevřít možnosti pro nárůst složitosti uměle vytvořených genetických obvodů. V této práci byly pro dané účely využity neuronové sítě. V následujících podsekcích je popis základních informací o fungování neuronových sítí se zaměřením na moduly, které byly použity při vytváření modelů. Sekce je rozdělena na dvě podsekce. První se zabývá přímo moduly pro se-

[5] Alosterické enzymy regulují rychlost metabolických drah.

stavení modelu sítě. Druhá je věnována systému SentencePiece, který je použit na předzpracování dat. Bližší seznámení s vytvořenou strukturou a funkčností je pak následně až v sekcích 3.1 a 3.3.

2.4.1 Neuronové sítě

Umělé neuronové sítě (dále jen neuronové sítě nebo NN) jsou vysoce nelineární modely inspirované biologickými neuronovými sítěmi v mozku [59]. Aproximací biologických neuronů tak vznikly umělé (formální) neurony, jejichž zapojení do umělé neuronové sítě vytváří takzvané fully-connected vrstvy (znázornění na obrázku 2.21). Postupem času začaly vznikat rozdílné sítě, které se odklonily od původní architektury a mají větší specializaci a schopnosti pro určité typy úloh. Řeč je například o konvolučních neuronových sítích (pokročilé zpracování obrazu) nebo o sítích s dlouhou krátkodobou pamětí (v originálu Long short-term memory, dále **LSTM**), které jsou určené primárně pro práci s textem [60, 61]. V této podsekci budou postupně popsány vybrané moduly, ze kterých byly sestaveny modely vytvořené pro diplomovou práci.



Obrázek 2.21: Schéma hluboké neuronové sítě se vstupní vrstvou, dvěma skrytými fully-connected vrstvami a jednou výstupní vrstvou.

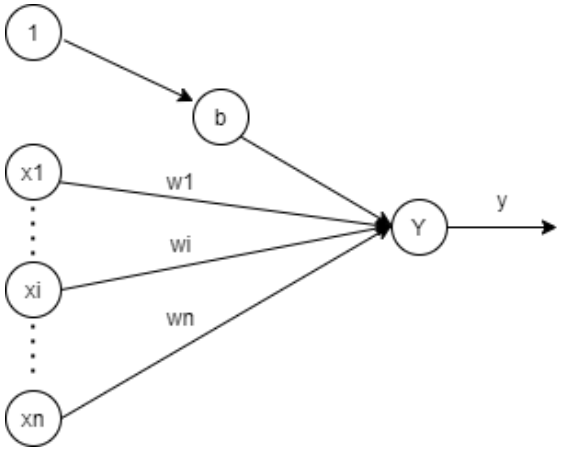
Biologický vs. formální neuron

Neurony jsou základní stavební funkční prvky nervové soustavy. Jedná se o buňky specializované na přenos, zpracování a uchování informací. Přenos informací je zprostředkováván vstupními (dendrit) a výstupními (axon) kanálky vystupujícími z těla neuronu (soma). Informace se poté přenáší pomocí synapsí, což je mezineuronové rozhraní. Synapse jsou z funkčního hlediska buď excitační, jež umožňují rozšiřování vzruchu, a nebo inhibiční, které naopak vzruch utlumují. Soma i axon jsou pokryty membránou, která je schopna generovat elektrické impulsy, jenž mohou být pomocí dendritů synaptickými branami přenášeny i na sousední neurony. V případě překročení aktivační hranice (prahu) neuron samotný generuje další impuls. Po průchodu signálu se mění synaptická propustnost, čemuž se přisuzuje paměťová schopnost neuronů [59].

Formální neuron (dále jen neuron) je základem matematického modelu neuronové sítě (viz obr. 2.22). Neuron Y má n reálných vstupů tvořících vstupní vektor $x = (x_1, ...x_n)$, který je ohodnocen váhovým vektorem $w = (w_1, ...w_n)$. Stejně jako u synapsí v biologickém neuronu, i zde mohou být váhy záporné. Vstupní potenciál neuronu je pak suma převážených vstupních signálů s prahem (b), který může být braný jako váha $w_0 = b$ pro vstup $x_0 = 1$:

$$y_{in} = \Sigma_{i=0}^{n} w_i x_i = \Sigma_{i=i}^{n} w_i x_i + b, \tag{2.18}$$

kdy vnitřní potenciál po dosažení hodnoty b indukuje výstup y. Nelinearita výstupu je dána aktivační (přenosovou) funkcí. Běžně používanými funkcemi jsou například skokové přenosové funkce, sigmoidy nebo hyperbolické tangenty.



Obrázek 2.22: Vizualizace formálního neuronu s prahem.

Backpropagation

Backpropagation je nejrozšířenější adaptační algoritmus zpětného šíření chyby ve vícevrstvých neuronových sítích, kdy je používán přibližně v osmdesáti procentech všech aplikací [59]. Algoritmus je založený na gradientní metodě, kdy změna gradientu udává rozdíl chybovosti neuronové sítě se změnou vah synaptických spojů. Cílem učení je úprava vah spojů mezi neurony tak, aby klesal gradient a tím byly minimalizovány chyby sítě. Algoritmus je rozdělený na tři fáze: dopředné šíření, zpětné šíření chyby a aktualizace vah.

Při dopředném šíření dochází pomocí excitace neuronů k postupnému průchodu dat sítí. Výstupem neuronové sítě je poté odezva na daný vstupní podnět, který je srovnán se vstupní informací od učitele. V biologii se analogicky provádějí stejné procesy, kdy vstupní vrstvu mohou tvořit například zrakové buňky.

Při zpětném šíření (takzvaná adaptace) dochází k postupu informace směrem od vrstev vyšších k nižším. Vypočítané aktivace y_k jsou srovnávány se stanovenými

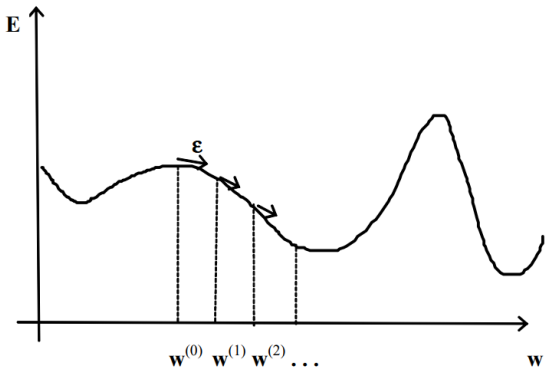
trénovacími výstupními hodnotami t_k pro každý neuron ve výstupní vrstvě a pro každý trénovací vzor. Srovnáním je získána chyba neuronové sítě:

$$E(w) = \Sigma_{l=1}^{q} E_l(w), \tag{2.19}$$

jež je daná součtem parciálních chyb $E_l(w)$ pro jednotlivé trénovací vzory:

$$E_l(w) = \frac{1}{2}\Sigma_{k\epsilon Y}(y_k - t_k)^2. \tag{2.20}$$

Poté se chyba částečně šíří do vrstvy předcházející (δ_k pro $k = 1, ...m$ - pro výstupní neuron Y_k). Úprava vah w_{jk} pak závisí na faktoru δ_k a aktivačních funkcích neuronů, do kterých se chyba zpětně šíří. Takto prováděná optimalizace chyby není na nelineárním systému triviální úkol. Pro aplikaci je však třeba diferencovatelnost chybové funkce. Se zvolenou konfigurací $w^{(0)}$ se v tomto bodě chybové funkce sestrojí tečný vektor (gradient) $\frac{\delta E}{\delta w}(w^{(0)})$ a dojde k posunu o δ ve směru gradientu (viz obr. 2.23). Tím je získána nová konfigurace, která se stejným způsobem mění až do nalezení lokálního minima chybové funkce.



Obrázek 2.23: Ukázka vývoje chyby na základě zvolené konfigurace v gradientní metodě (z [59]).

Dense vrstvy

Dense vrstvy jsou pojmenováním pro fully-connected vrstvy ve frameworku Tensorflow-keras, který zde byl využit pro trénování neuronových sítí. Dense vrstvy jsou hojně využívaným modulem pro neuronové sítě, kdy jsou všechny neurony vrstvy propojeny s neurony vrstvy předcházející. Jednotlivé neurony tak mají tolik vstupů, kolik je v předchozí vrstvě neuronů a jeden výstup, který je dále vstupem pro neurony v následující vrstvě. Například v obrázku 2.21 označený neuron má dvě hodnoty na vstupu a jeho výstupní funkce je vstupem pro dva neurony v nadcházející vrstvě. Dense vrstvy jsou obecně nelineární[6], přičemž v základu mají lineární předpis $wx+b$,

[6]Aktivační funkce může být lineární, v tu chvíli však nemá smysl takto skládat za sebe více vrstev, neboť složením lineárních funkcí vzniká opět funkce lineární. Jednalo by se tak o zbytečné plýtvání výpočetními zdroji.

který poté prochází nelineární funkcí (aktivační funkce):

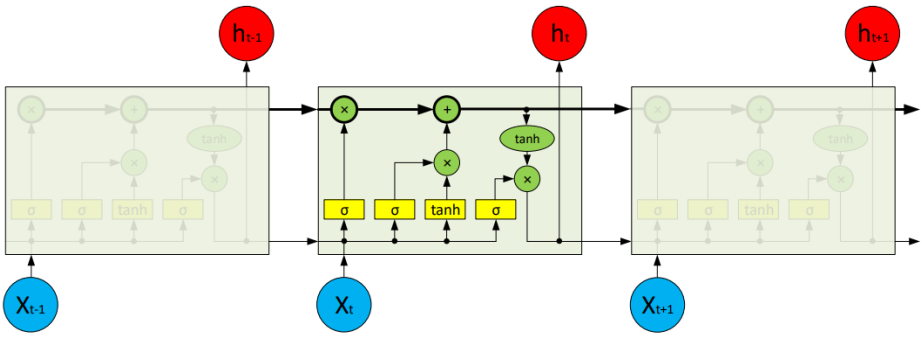
$$y = f(wx + b). \tag{2.21}$$

Při vytvoření více vrstev (viz obr. 2.21) je tak možné tyto nelineární funkce za sebou kumulovat a modelovat komplexnější matematické funkce. Dense vrstvy však mají omezení z hlediska možnosti zachycení opakování v čase nebo získávání různého výstupu na stejný vstup. Pro tyto účely vznikly sítě s rekurentní strukturou.

LSTM

Velice důležitým prvkem pro trénování neuronových sítí v této práci byla architektura LSTM, v originálu Long short-term memory. LSTM má rekurentní strukturu. Na rozdíl od klasických rekurentních sítí, je schopna určit dlouhodobé závislosti. Architektura je tak vhodná například na rozpoznávání řeči, ručně psaných textů a klasifikaci textů [62].

Strukturou LSTM jsou zřetězené opakující se moduly s třemi typy bran (vstupní, výstupní, zapomínající), které regulují tok informací skrz daný modul (obr. 2.25). Vnitřní stav modulu pak prochází celým řetězcem a je ovlivňován lineárními interakcemi, procházející informace tak mohou být beze změn.



Obrázek 2.24: Zřetězení modulů v architektuře LSTM (z [62]).

Postupným průchodem skrz strukturu modulu (obr. 2.25) zapomínající brána rozhoduje o uchovávání vnitřního stavu buňky C_{t-1}. Srovnáním současného vstupu x_t a předchozího výstupu h_{t-1} je vytvořen vektor f_t, který dosahuje hodnot od nuly do jedné, jež odpovídají míře zachování předchozího stavu:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \tag{2.22}$$

kdy hodnota jdoucí k nule je pro zahození, hodnota jdoucí k jedné pro plné zachování.

Vstupní brána se stará o ukládání nových informací do vnitřního stavu modulu C_t. V této fázi nejprve dochází k rozhodnutí pomocí sigmoidální funkce i_t, jaké hodnoty budou aktualizovány:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \tag{2.23}$$

Tangent funkcí vytvořený list kandidátů $\tilde{C}_t$ pak udává možné kandidáty na přidání do vnitřního stavu C_t:

$$\tilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c). \tag{2.24}$$

Aktualizace stávajícího stavu C_{t-1} na C_t vypadá následovně:

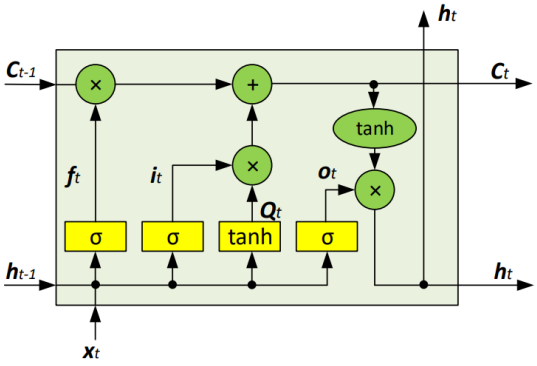
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t. \tag{2.25}$$

Poslední brána v pořadí je výstupní. Zodpovídá za tvorbu výstupní hodnoty, která je závislá na aktuálním upraveném stavu C_t pomocí funkce hyperbolického tangentu. Na základě vstupu a předchozího výstupu je rozhodnuto funkcí o_t o složkách vnitřního stavu, které budou přivedeny na výstup modulu:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o). \tag{2.26}$$

Výstupní funkce je poté ve tvaru:

$$h_t = o_t \cdot tanh(C_t). \tag{2.27}$$

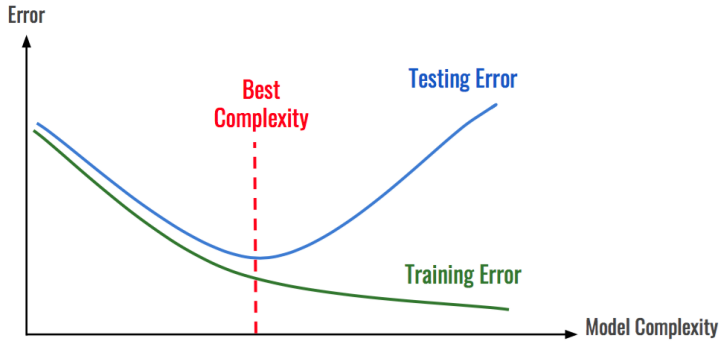


Obrázek 2.25: Modul LSTM sítě (z [62]).

Dropout vrstvy

Metoda Dropout se snaží bojovat s fenoménem zvaným overfitting neboli přetrénování (viz obr. 2.26). V tomto případě dochází k události, kdy se síť přizpůsobuje na speciální trénovací data a ztrácí svoji kvalitu na obecných validačních datech. Většinou

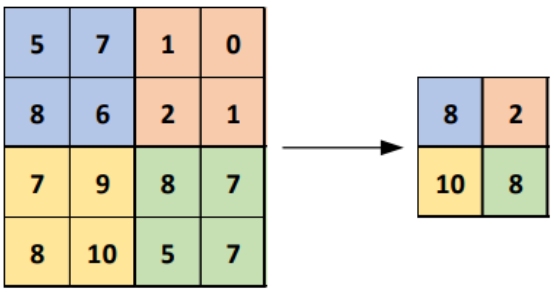
se Dropout vrstvy aplikují na Dense vrstvy, kdy se na základě pravděpodobnosti dropoutu p_d v aktuálním průchodu dat při trénování neuvažují určité neurony. Přístup by při správném návrhu měl dělat neurony na sobě nezávislými.



Obrázek 2.26: Nastínění overfittingu. Od určité fáze se chyba na trénovacích datech stále zmenšuje, avšak na testovacích datech již roste.

Max-pooling

Pooling vrstvy obecně mají za úkol slučovat několik hodnot do jedné, přičemž dochází k redukci počtu vnitřních stavů. V práci byl využit typ max-pooling, který v okně vybrané velikosti zachovává pouze nejvyšší hodnotu (obr. 2.27).

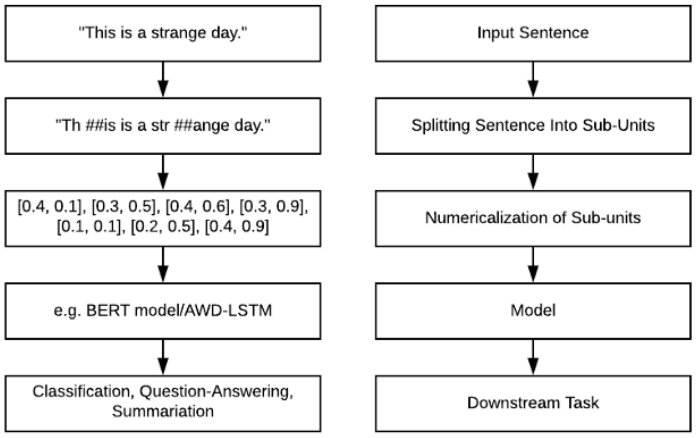


Obrázek 2.27: Názorná ukázka fungování max-poolingu (z [62]).

2.4.2 SentencePiece

SentencePiece je tokenizér a detokenizér založený na učení bez učitele [63]. Hlavním využitím daného algoritmu je vytváření reprezentace textu pro vstup do neuronových sítí. Modul se snaží efektivním způsobem zmírnit problémy s otevřenou slovní zásobou, kdy se na vstupu vkládají věty a algoritmus je rozděluje na vlastní slova, jejichž počet je předem určen při spuštění. Věty není třeba nijak předzpracovávat, implementace SentencePiece je dostatečně rychlá i pro natrénování modelu ze surových vět. Naučený model vstupní větu nejprve rozdělí do naučených částí, které jsou poté na základě ID zakódovány. Věta je poté složena z čísel reprezentujících

jednotlivá slova, které lze dále využít například jako vstup pro LSTM neuronovou síť (viz obr. 2.28). Při předložení DNA algoritmus rozdělí sekvenci na 'slova' s průměrnou délkou přibližně čtyř bází, čímž zmenšuje vstupní sekvenci na čtvrtinu.



Obrázek 2.28: Nastínění funkčnosti a možného využití algoritmu SentencePiece (z [64]).

2.5 Experimentální metody pro ověření výsledků

Před aplikací genetických úprav je třeba vše předem řádně vyzkoušet a ověřit, zda samotný návrh neobsahuje žádné chyby a před objednáváním potřebných věcí vše vypadá dle představ. Jedná se o první předpoklad pro úspěšné provedení zásahu do genetického kódu organismu. I tak ale platí, že v biologii nic není jisté a i návrh, který na první pohled vypadá dobře, nemusí nakonec vůbec fungovat.

Existuje celá řada možností, jak tento proces realizovat. Zde jsou popsány dva nástroje použité pro návrh a vytvoření experimentů: Benchling [65] a Modular Cloning.

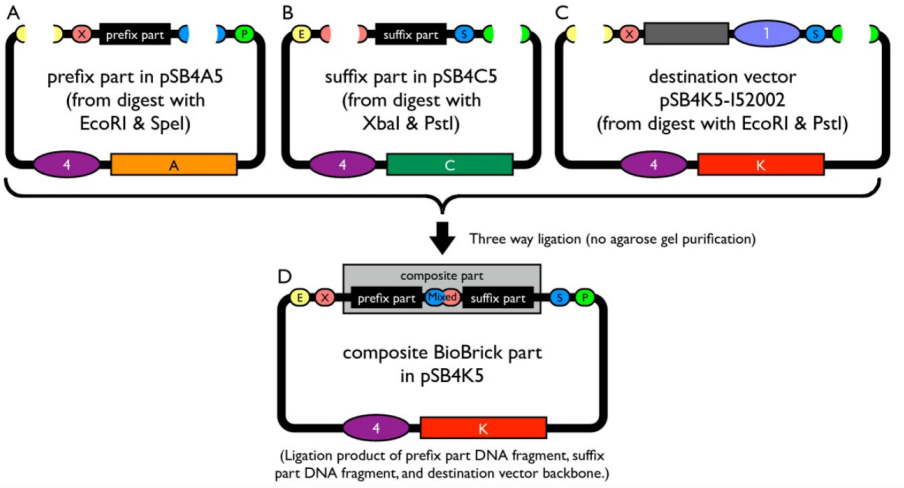
2.5.1 Modular Cloning

V této podsekci bude uveden popis skládání genů na základě metody Modular Cloning (zkráceně MoClo). Popsán bude převážně postup uvedený v článku (Lee et al. [14]), který je zaměřen na genetické úpravy v kvasince druhu *Saccharomyces cerevisiae*, jež byla cílem pro modifikaci i v této práci. Kvasinka *Saccharomyces c.* je populárním organismem v syntetické biologii, je dobře prozkoumaná a i hojně využívaná v průmyslových aplikacích. Základní myšlenkou rozebraného článku je vytvořit modulární standardizovaný systém sestavování genů a multigenů, které se následně mají vkládat do cílového organismu. Zprvu články tohoto typu cílily spíše na bakterii *Escherichia coli*, zde tedy dochází ke konvertování metody na použití

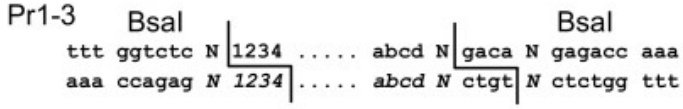
v organismu *Saccharomyces c.*

Pro možnost sestavovat geny v relativně krátké době a nemuset vždy vytvářet vše od začátku vznikly metody, při kterých je gen sestavován po částech s využitím restrikčních enzymů. Velice známou a průkopnickou metodou v této oblasti, je takzvaný BioBrick (2003) [66]. Pomocí restrikčních enzymů se vytvoří z každé potřebné části lineární DNA vektory, které se poté spojí (viz obr. 2.29). Touto metodou však v místech spojů vznikají "jizvy" a výsledný plazmid rekonstituuje tato místa. Není tedy možné je znovu při restrikci využít. Tato metoda má i další omezení. Najednou lze v jedné reakci udělat pomocí jednoho restrikčního enzymu pouze jedno spojení.

V roce 2008 bylo představeno vylepšení této metody využívající Modular Cloning, které se nazývá Golden Gate [67]. Ke zlepšení dochází v oblasti funkčnosti restrikčních enzymů. S tím je spojená i konkatenace více částí najednou pomocí jednoho restrikčního enzymu. Používané restrikční enzymy krájí DNA mimo jejich rozpoznávací sekvenci, je tedy možné navrhnout potřebné části tak, aby při spojení nevznikaly žádné jizvy. Místo, kde se DNA rozsekne, není navíc z hlediska složení na ničem závislé. Pro jeden restrikční enzym vytvářející overhang o délce čtyři (obr. 2.30) je tak možné mít při všech kombinacích nukleových bází teoreticky až 256 možných částí, které by šlo najednou následně spojit dohromady ve one-pot reakci.

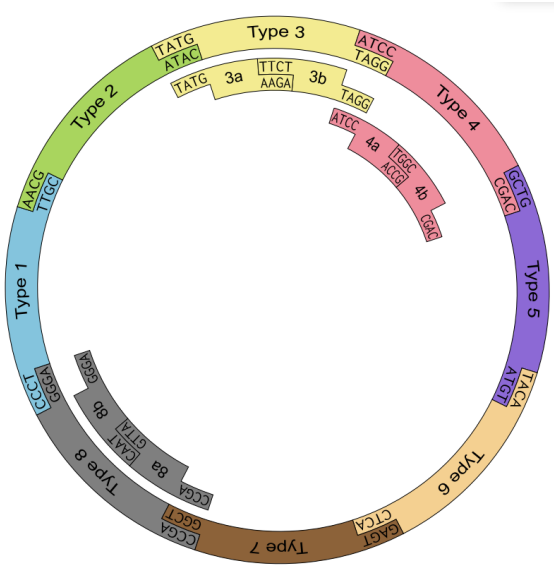


Obrázek 2.29: Ukázka DNA assembly pomocí metody BioBrick s popisy použitých restrikčních enzymů pro jednotlivé části: (A) *EcoRI*, *SpeI*. (B) *XbaI*, *PstI*. (C) *EcoRI*, *PstI*. (z [68])



Obrázek 2.30: Příklad rozkrojení DNA pomocí restrikčního enzymu *BsaI*.

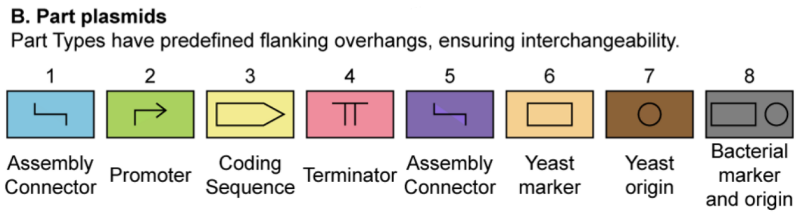
Jsou zde však jistá omezení. Například při *BsaI* restrikci (vznik overhangů o délce čtyři) by v jedné reakci neměly být dva overhangy se shodnými bázemi na třech indexech. Dané omezení lze však jednoduše dodržet se zachováním dostatečného počtu částí pro následnou DNA assembly. Na tomto principu autoři článku (Lee et al. [14]) vytvořili knihovnu cutting-sites vyobrazenou na obrázku 2.31. Každá zde vykreslená oblast (v obrázku *Type*) má přiřazenou vlastní funkcionalitu (obr. 2.32). To zajišťuje potřebnou modularitu pro následné jednodušší kombinování jednotlivých částí a výrazně tak zkracuje dobu, kterou by návrhu jinak člověk musel věnovat.



Obrázek 2.31: Grafické znázornění knihovny cutting-sites spolu s částmi, které by měly na základě probíraného článku ohraničovat (z [14]).

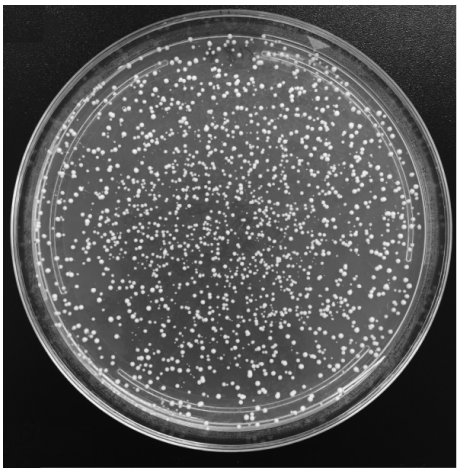
Navrženým rozložením částí je určitě dobré se řídit. Například *Type 2* (z obr. 2.31), tedy promotor (z obr. 2.32) končí ve vláknu 5‘→ 3‘ restrikčním místem se sekvencí bází `TATG`. Při návratu do sekce 2.1, po rozdělení vláken a započetí transkripce ke konci promotoru, poslední tři báze sekvence tohoto overhangu jsou v jazyce mRNA: `AUG`, což odpovídá start kodonu a začíná tím tedy ORF. Jistě by si šlo přidat start kodon na začátek části za jiný overhang, v tu chvíli by se však na konec promotoru přidala minimálně čtyř-bázová jizva, která na konci promotoru není žádoucí. V tomto případě tam je přidána navíc pouze jedna báze `T`, která se statisticky ve směru 5‘→ 3‘ vyskytuje v promotorech nejčastěji. Tato statistika byla provedena na datasetu obsahujícím všechny promotory ze *Saccharomyces c.*

Nachází se zde i popis jak správně využít na první pohled periferní oblasti assembly tak, aby rovnou vznikala kontrola do jakých buněk se podařilo požadovanou genetickou informaci vložit. Na to je třeba mít připravené kmeny buněk, které mají vyřazen některý z genů podílejících se na tvorbě příslušné aminokyseliny [14]. Vyřazený gen se společně s novou genetickou informací vkládá do buňky. Tím buňka, do které se navržená informace vložila, získává zpět schopnost danou aminokyselinu



Obrázek 2.32: Popis, co by jednotlivé části z obr. 2.31 měly mít za funkční vlastnost (z [14]).

vytvářet. Po provedení integrace nového kusu DNA do kvasinek se výsledný produkt dává na Petriho misky se živným médiem, které neobsahuje danou aminokyselinu. V případě, že se do genomu požadovaná informace nedostala, buňka si aminokyselinu nedokáže sama vytvořit, z média ji také nezíská a umírá. Pouze buňky, kterým se vkládaná genetická informace správně zaintegrovala, by měly mít možnost potřebnou aminokyselinu vytvářet a tím jako jediné přežít a vytvořit na misce kolonii (obr. 2.33). V případě volby špatné misky by na základě dalších vyřazených genů v kmenu a obsahu živného média byla pokryta buď celá miska, nebo by tam naopak nevyrostlo vůbec nic.



Obrázek 2.33: Růst kvasinek se zpětně integrovaným genem pro tvorbu potřebné aminokyseliny k přežití na odpovídající misce (z [14]).

2.5.2 Benchling

Benchling je webová aplikace sloužící k návrhu a testování procesů, které mají následně proběhnout v laboratoři. Přestože může existovat znalost chování jednotlivých částí, ze kterých se skládá požadovaný výsledný produkt, jeho chování může být naprosto neočekávané. Jak je napsáno na začátku této kapitoly, je dobré mít všechny kroky nejprve ověřené v nějakém nástroji tohoto typu, aby se poté do

experimentů nezanášely ještě problémy způsobené chybou v návrhu. Celý proces integrace genetické informace do kvasinky je dlouhý a chyby tak nestojí jen peníze za spotřebované vybavení, chemické látky a DNA, ale i za mnoho času.

V Benchlingu je možné si projít postupně všechny potřebné operace pro kopírování postupů v laboratoři. Lze zde s omezením nasimulovat úvodní amplifikaci nějaké oblasti DNA z vybraného organismu pomocí navržených primerů. Přes virtuální PCR reakci se vybraná část vyamplifikuje a při správném navržení primerů se na okrajích vytvoří potřebná restrikční místa pro následnou domestikaci (vložení) vyamplifikovaného fragmentu do part-plazmidu. Z vybraných part-plazmidů se následně pomocí Golden Gate assembly vytvoří navržený produkt. Výsledné geny je dále možno spojovat do multigenů. Celý proces je důkladně popsán s různými typovými příklady jak reprezentovat jednotlivé části assembly (viz obr. 2.31) ve výše rozebíraném článku (Lee et al. [14]).

Kapitola 3

Inovace

Při vypracování diplomové práce byly využity různé přístupy k řešení úlohy vkládání regulačních míst do promotorů. Struktura této kapitoly je rozdělena do tří částí. V první se nachází popis všech zformulovaných přístupů řešení, spolu se schématy naznačujícími tok informací skrz modely neuronových sítí. Další sekce je zaměřena na použitá data. Jsou zde popsány datasety pro trénování sítí a databáze obsahující užitečné informace o organismu *Saccharomyces c.*, která sjednocuje více databázových zdrojů a přidává vlastní analytické informace. Poslední částí je podrobný popis sestavovaných modelů neuronových sítí spolu s přípravou dat.

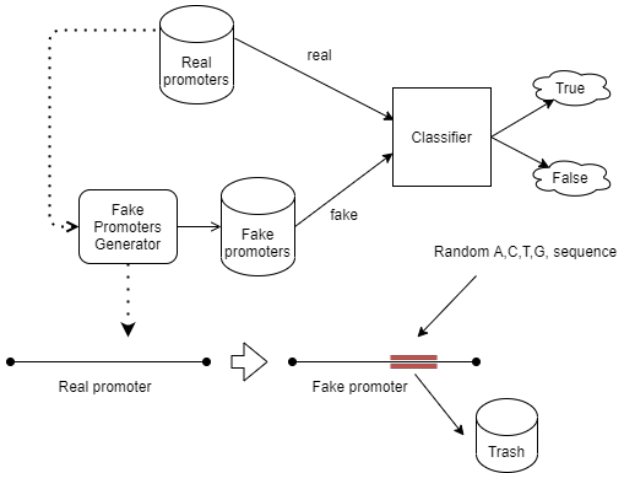
3.1 Formulace úlohy pro machine learning

Pro úlohu vkládání regulačních míst do promotorů byly v diplomové práci vypracovány postupně tři hlavní přístupy. Jednotlivé typy jsou pojmenovány jako *Classifier*, *Place-back* a *Insert-fragment*.

3.1.1 Classifier

Prvním vytvořeným návrhem byl přístup s názvem *Classifier*. Snahou bylo naučit tento systém rozpoznávat reálné promotory od upravených. K tomu byl využit jednoduchý klasifikátor, který přes binary cross-entropy[1] rozhoduje o reálné, či umělé podobě promotoru na vstupu. Umělé promotory byly vytvořeny pomocí generátoru s předem definovanou funkčností. Z reálného promotoru se vyřízne sekvence bází, která se zahodí a následně nahradí náhodně vygenerovanou posloupností bází `A,C,T,G`. Schéma sestavení modelu je na obrázku 3.1. Po natrénování se nakonec síti předkládají upravené promotory, které na výstupu dostanou ohodnocení jejich věrohodnosti. Neuronová síť tak hodnotí, jestli promotor vypadá realisticky a funkčně.

[1] Binary cross-entropy je ztrátová funkce používaná pro binární klasifikaci.

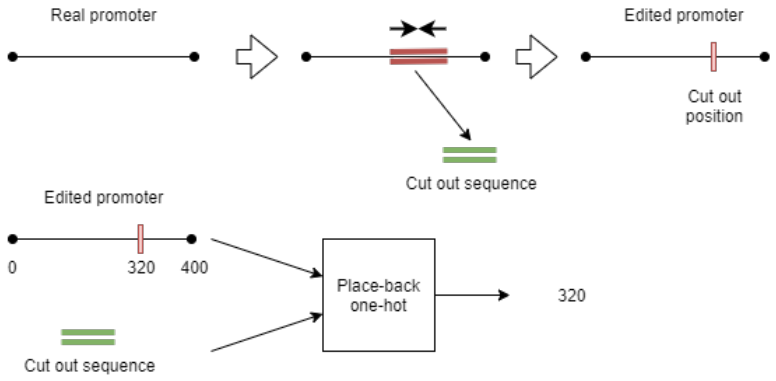


Obrázek 3.1: Schéma přístupu nazývaného *Classifier*. Reálné a vygenerované promotory vstupují do klasifikátoru s odpovídajícím labelem real/fake a klasifikátor hodnotí, jestli vypadají reálně (*True*) nebo uměle (*False*).

3.1.2 Place-back

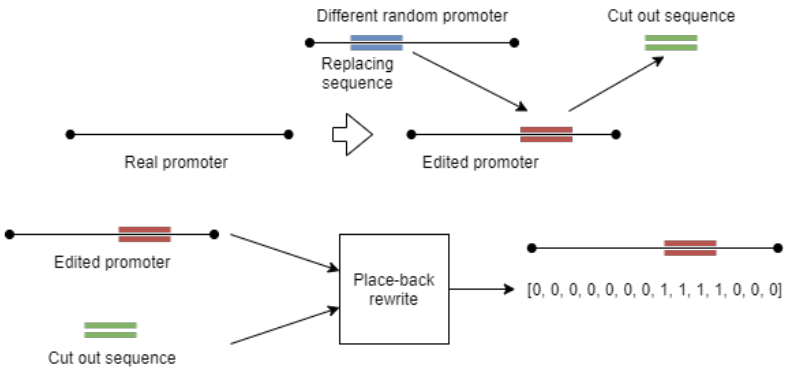
Dalším přístupem pro řešení úlohy je metoda nazvaná *Place-back*. Jedná se o nově navržený typ sítě, při kterém se z promotoru nejprve odebere část sekvence, která má být do promotoru sítí vrácena. Pro daný přístup vznikly dvě různé realizace.

První realizace *Place-back* (s přidruženým názvem *one-spot*) vyřezává z promotoru sekvenci bází, které se poté snaží vrátit na původní místo. Na vstupu sítě je promotor s vyříznutou sekvencí, která se sama stává druhým vstupem. Na výstupu by pak měla být hodnota v rozsahu velikosti vstupního vektoru odpovídající indexu, odkud byla část sekvence vyříznuta. Pro testování se pak síti předloží promotor a sekvence bází (operátor). Na výstupu při testování je pak odhad, kam daný operátor do promotoru umístit. Schéma přístupu se nachází na obrázku 3.2.



Obrázek 3.2: Schéma přístupu nazývaného *Place-back-one-spot*. Pro vyříznutou sekvenci neuronová síť zpětně hledá její původní umístění.

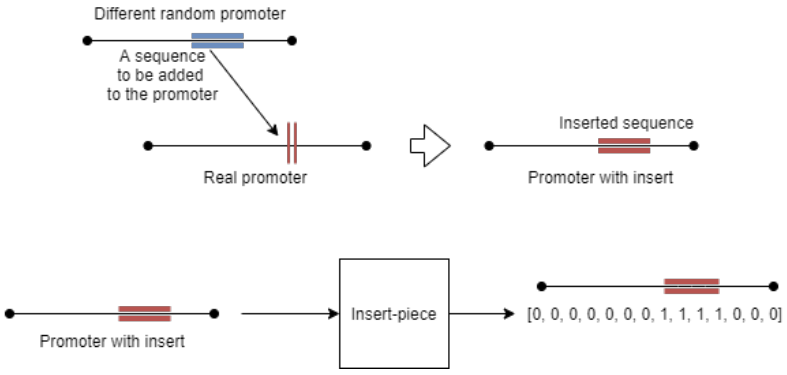
Druhou verzí je následně metoda *Place-back-rewrite*, která vyříznutý kus promotoru nahradí sekvencí stejné délky z jiného náhodně vybraného promotoru. Neuronová síť má v tomto případě za úkol odhadnout místo, kam se má vyříznutý fragment zpátky přepsat. Na vstupu sítě je pozměněná sekvence promotoru a vyříznutý fragment z původního promotoru. Na výstupu je poté dle délky vstupní sekvence dlouhý vektor nul s jedničkami v místě úvodního nahrazení fragmentu. Testování natrénované sítě pak probíhá (jako v *Place-back-one-spot*) vložením promotoru a sekvence na vstup, kdy na výstupu je vektor s jedničkami v místě, kde by se měla daná sekvence přepsat. Schéma daného přístupu se nachází na obrázku 3.3.



Obrázek 3.3: Schéma přístupu nazývaného *Place-back-rewrite*. Vyříznutá sekvence je nahrazena sekvencí z jiného operátoru, kdy neuronová síť poté vyříznutou sekvenci umisťuje zpátky do promotoru na její původní místo.

3.1.3 Insert-fragment

Posledním modelem je architektura nazvaná *Insert-fragment*. Metoda je svou myšlenkou podobná přístupu *Place-back-one-spot*. Do přírodního promotoru je do vybraného místa vložena cizí sekvence bází, kterou má neuronová síť odhalit. Vstupem této sítě je promotor, který má v sobě umístěný cizí fragment. Oproti tomu na výstupu má být nulový vektor délky vstupu s jedničkami v oblasti uměle vložené sekvence. Po natrénování se síti předkládají promotory s uměle vloženými regulačními oblastmi. V případě, že by síť takto vložené regulační oblasti odhalila, znamenalo by to, že je pro síť vkládaná sekvence nápadná a mohla by narušovat funkci promotoru. Schéma přístupu se nachází na obrázku 3.4.



Obrázek 3.4: Schéma přístupu nazývaného *Insert-fragment*. Do přírodního promotoru je vložena na náhodné místo vybraná cizí sekvence, kterou se neuronová síť snaží odhalit.

3.2 Data

K natrénování modelů neuronových sítí byla postupně využita celá řada datasetů, ve výsledku však byly pro trénování a validaci využity pouze tři z nich. V následujících podsekcích budou probrány vytvořené trénovací a validační datasety společně s popisem databáze, která vznikla kombinací několika zdrojů a která sloužila k získání některých specifických informací.

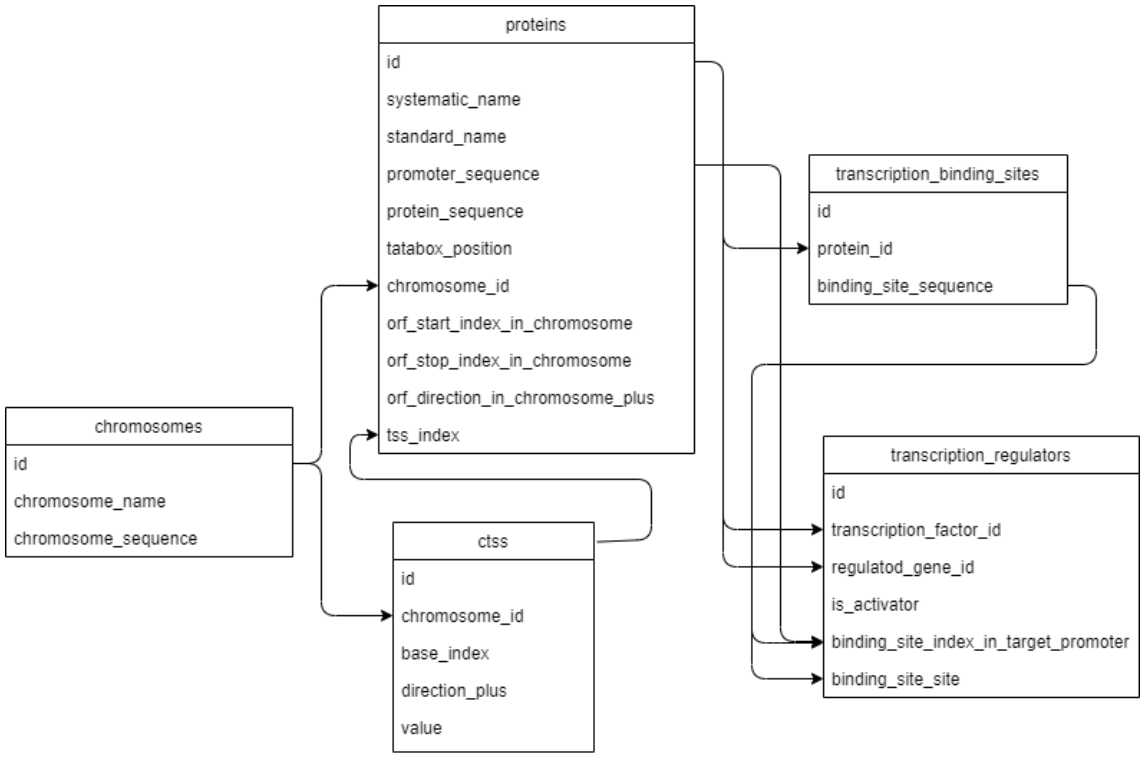
3.2.1 Databáze *Saccharomyces cerevisiae*

V průběhu vypracování diplomové práce byl založen databázový systém shrnující potřebné informace o organismu *Saccharomyces c.* Jako zdroje posloužily databáze: *Saccharomyces Genome Database*[69], *Yeastract* [70] a *YeasTSS* [71]. Propojením dílčích informací, které jednotlivé databáze poskytují, byl vytvořen celek zaměřený na práci s promotory. Schéma databáze se nachází na obrázku 3.5.

Ze schématu na obrázku 3.5 vyplývá, že první tabulkou celého stromu je tabulka *chromosomes*, která obsahuje jméno (*chromosome_name*) a celou sekvenci (*chromosome_sequence*) pro všechny chromozomy organismu *Saccharomyces c.*

Další tabulkou v hierarchii je *ctss* získaná z YeasTSS. Ta udává hodnoty odpovídající odhadovanému počátku transkripce v promotoru, tedy TSS (v tabulce *ctss* sloupec *value*). Pro danou *value* jsou přidruženy informace jako její souřadnice. Postupně udávají, v jakém chromozomu se hodnota nachází (*chromosome_id*), index báze v chromozomu (*base_index*) a směr, ze kterého byla hodnota naměřena (*direction_plus*).

Další v pořadí je tabulka s pojmenováním *proteins*, která obsahuje vyjmenované informace: systematické jméno proteinu (*systematic_name*), standardní jméno proteinu (*standard_name*), sekvenci promotoru (*promoter_sequence*), aminokyselinovou sekvenci proteinu (*protein_sequence*), chromozom do kterého protein patří (*chromosome_id*), start a konec kódující sekvence v promotoru



Obrázek 3.5: Schéma pěti-tabulkové databáze obsahující informace pro práci s promotory organismu *Saccharomyces cerevisiae*.

(*orf_start_index_in_chromosome*, *orf_stop_index_in_chromosome*) a směr kódující sekvence v DNA (*orf_direction_in_in_chromosome_plus*). Dále byly pomocí vytvořeného algoritmu nalezeny TATA-boxy, jejichž pozice od začátku promotoru je zanesena ve sloupci *tatabox_position*. Posledním sloupcem je *tss_index* udávající TSS získanou průchodem dat z *ctss*.

Předposlední tabulkou je *transcription_binding_sites*, která obsahuje zpracované informace z Yeastract. V tabulce jsou vazebné sekvence v promotoru (*binding_site_sequence*) a odkaz na protein, který se na danou vazebnou pozici váže (*protein_id*).

Poslední v pořadí je tabulka *transcription_regulators*, pro kterou byla získána data z *Yeastract*. Tabulka obsahuje nativně informace o transkripčním faktoru, což je zde ID regulačního proteinu (*transcription_factor_id*). Dále zde najdeme ID proteinu, který je transkripčním faktorem regulován (*regulated_gene_id*) a nakonec informaci o typu regulace, zda-li indukuje, nebo represuje expresi (*is_activator*). Pro dvojice transkripční faktor-regulovaný protein s nalezeným odpovídajícím regulačním místem v sekvenci cílového promotoru má tabulka vyplněny další dva sloupce. Sloupec *binding_site_index_in_target_promoter* odpovídá indexu, kde bylo v promotoru nalezeno regulační místo. Poslední sloupec (*binding_site_site*) poté obsahuje sekvenci regulačního místa, které odpovídá páru transkripční faktor-regulovaný protein, a které bylo nalezeno v cílovém promotoru.

3.2.2 Vytvořené datasety

Vytvořené datasety pro práci s neuronovými sítěmi se dají rozdělit do tří skupin. První sadu tvoří datasety určené pro předtrénování, další skupinou jsou datasety na dotrénování a poslední jsou datasety pro validaci a testování. Od začátku zpracování této práce bylo cíleno na modifikace v organismu *Saccharomyces c.* Použitá data jsou silně ovlivněna tímto cílem.

Pro předtrénování vznikaly datasety obsahující obecnější sekvence pro naučení sítí širšího kontextu. Prvním pokusem byla data s celými sekvencemi chromozomů *Saccharomyces c.* Sekvence promotorů jsou však velice specifické a informace o zbytku genomu tak nebyly pro trénování relevantní. Další datasety se tak zaměřovaly vyloženě na oblasti promotorů. První takto vytvořený dataset obsahoval promotory z čtrnácti různých eukaryot získaných z *Eucaryotic Promoter Database* [72]. Dataset obsahuje promotory savčí, ptačí, rostlinné, promotory kvasinek a řas. Promotory nemají jenom specifickou charakteristiku oproti zbytku genomu, mají také specifickou strukturu i napříč různými organismy. Tento dataset byl po několika pokusech trénování také zavrhnut. Další dataset byl vytvořen ze zpracovaných informací ve výše popsané databázi (podsekce 3.2.1). S tímto datasetem byl pozměněn styl učení neuronové sítě v metodě *Place-back-rewrite* (na jiné metody se dataset nezkoušel). Standardně se síť učila obecnou strukturu promotorů tím, že se vyřezávala náhodně vybraná místa, která se zaplnila cizí sekvencí a vyříznutá sekvence se zpětně mapovala pomocí modelu sítě do promotorů. V tomto případě byla trénovací množina promotorů omezena na ty, ve kterých byly nalezeny nějaké regulační oblasti. Při přípravě dat pak nedocházelo k náhodnému vyřezávání sekvence bází, ale byla vyjmuta vždy nalezená regulační oblast. Síť se tak měla naučit pracovat přímo s regulačními místy v promotorech. Problémem tohoto přístupu byl nedostatek dat pro natrénování a také malá rozmanitost známých regulačních oblastí, která by následně omezovala možnosti libovolné volby regulačního místa pro vložení do promotoru. Posledním a nakonec použitým datasetem pro předtrénování, byla sada promotorů z říše *Funghi* získaná zpracováním dat z NCBI [73]. Celkový počet organismů, ze kterého se složil dataset na předtrénování, obsahoval nakonec promotory z 32 organismů.

Samotná data na dotrénování byla již jen jednoho typu. Jednalo se o devět desetin promotorů ze *Saccharomyces c.* V předtrénovacím datasetu se promotory *Saccharomyces c.* vůbec nevyskytují.

Ve validačním datasetu se nacházela zbylá desetina promotorů ze *Saccharomyces c.* obsahující všechny promotory, které byly následně využity pro testování natrénovaných modelů.

Posledním datasetem je sada promotorů určených pro testování. Jedná se o devatenáct konstitutivních promotorů popsaných v článku (Lee et al. [14]) a jeden promotor, který byl vybrán na základě znalosti jeho expertní regulace v laboratoři společnosti Xeno Cells Inovations s.r.o., jíž je duševním vlastnictvím. Promotor tedy nebude v práci blíže specifikován a bude dále nazýván jako *pX*.

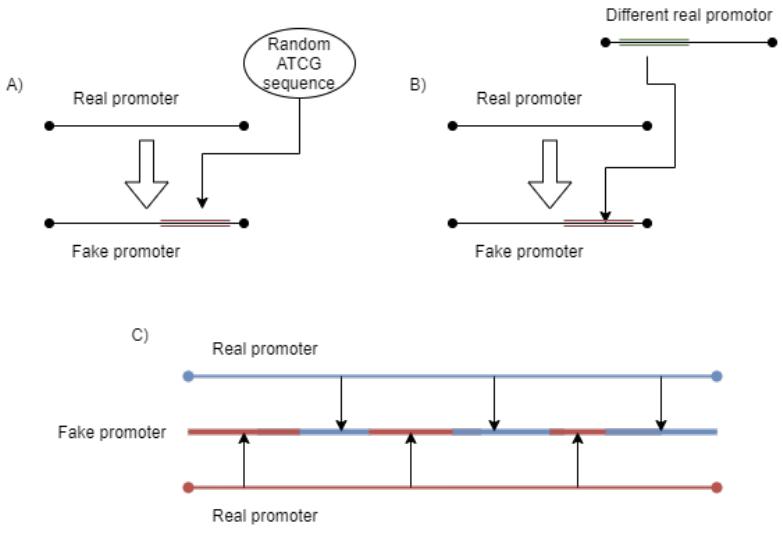
3.3 Neuronové sítě

Pro každou metodu popsanou v sekci 3.1 byla vyzkoušena řada modelů neuronových sítí s různě pozměněnými strukturami, parametry a vstupními datasety. V následujících podsekcích bude pro každý přístup shrnuto, jak probíhala příprava dat, modelu a jakým způsobem se dokonvergovalo k finálnímu nastavení.

3.3.1 Classifier

Classifier byl celkově prvním vyzkoušeným přístupem pro řešení problému regulace genetické exprese pomocí neuronových sítí. Vytvořené modely lze rozdělit do dvou hlavních bloků: využívající vs. nevyužívající modul SentencePiece. Pro obě možnosti byly vyzkoušeny tři shodné typy generování umělých promotorů, které teď budou popsány.

První typ generování využíval čistě náhodných sekvencí, které se umístily do promotoru (obr. 3.6-A). V druhém případě se vypůjčila sekvence z cizího promotoru, která nahradila stejně velkou oblast v promotoru původním (obr. 3.6-B). Posledním typem byla fúze dvou promotorů. K původnímu reálnému promotoru se náhodně vybral druhý reálný promotor a s nastavenou pravděpodobností se mezi oběma promotory přepínalo, až se vygeneroval celý kombinovaný promotor (obr. 3.6-C).



Obrázek 3.6: Schéma generování umělých promotorů. A) Přepsání části sekvence náhodnou sekvencí. B) Přepsání části sekvence sekvencí z jiného promotoru. C) Fúze dvou reálných promotorů do jednoho umělého.

Classifier bez SentencePiece

První pokusy o trénování byly prováděny bez modulu SentencePiece, který kóduje vstupní sekvence pro neuronové sítě. V této fázi byla práce z větší části experi-

mentální a zkoušely se všechny různé přístupy pro opatření úvodního směrodatného výsledku. Trénování zde probíhalo pouze na promotorech ze *Saccharomyces c.* Nyní budou popsány aplikované metody trénování s analýzou získaných výstupů.

Příprava dat probíhala dle vyobrazení na schématu 3.6, následná forma modelu se však v jednotlivých případech značně lišila. První pokusy o natrénování sítě byly prováděny pomocí konvolučních neuronových sítí[2]. Na vstupu byla matice složená z one-hot[3] vektorů délky čtyři s jedničkou na indexu odpovídajícímu aktuální bázi (`A/T/C/G`). Matice tak měla čtyři řádky a počet sloupců odpovídal délce promotoru (1000 bází). Přes takto reprezentovaný promotor poté přejížděla konvoluční jádra s výškou čtyři řádky a délkou od 32 do 128 sloupců. Mezi vrstvy byl zařazen Max-pooling pro zmenšení průběžných matic v síti a na konec byla umístěna Dense vrstva. Vzhledem ke klasifikaci mezi *True* a *False* byla jako ztrátová funkce použita *binary crossentropy* a jako kontrolní metrika *accuracy*. Díky architektuře, kdy dochází ke konvoluci jádra se vzorem, umí tyto neuronové sítě zachytit kontext v obrázku i přes různá otočení a jiné úpravy. Předpokladem tak bylo, že se síť naučí poznat přírodní bloky bází. V případě narušení přirozené sekvence by to síť rozpoznala a klasifikovala promotor jako umělý. Přes všechny vyjmenované typy generování umělých promotorů a různé počty konvolučních vrstev a jejich nastavení se nepodařilo získat pro daný model žádné výsledky, protože neuronová síť predikovala na výstupu vždy jen *True*. Validační *accuracy* tak byla 50 %, což při pravděpodobnosti 1:1 neříká vůbec nic.

Stejným neúspěchem skončil i pokus, kdy byl celý model sestaven pouze z Dense vrstev. Dense vrstvy obecně nemají vlastnost zachycovat vzdálený kontext. Nefunkčnost této architektury se tím pádem dala předpokládat.

Poslední přístup řešení byly LSTM sítě, jež jsou určeny pro práci s textem, kdy je možno díky vlastnímu vnitřnímu stavu a rekurenci zachytit vzdálený kontext. LSTM vrstvy byly využity oboustranné (Bidirectional). V tomto případě síť zachycuje kontext jak před tak i po oblasti, ve které se zrovna nachází. Za LSTM vrstvu pak byly umisťovány Dense vrstvy. Na vstupu sítě tentokrát nebyla matice představující posloupnost bází, ale byla zde posloupnost bází samotných, kterou si embedding vrstva[4] na startu sítě sama zakódovala. Takto postavená síť se již dokázala něco naučit a vytvořila slabý klasifikátor s úspěšností odhadu nad 52 %. Bylo to stále nesmírně nízké číslo, ukázalo to však, že pro danou úlohu je síť schopná alespoň omezeného učení.

Popsané modely se svými výstupy ani nepřibližovaly požadovanému výsledku. Důvodem nezdaru mohla být přílišná komplexita a délka promotorů (1000 bází), kterou neuronová síť nebyla schopna na předkládaném vzorku promotorů pochopit. Z důvodu redukce velikosti promotoru se poté přešlo na práci s enkodérem SentencePiece.

[2]Konvoluční neuronové sítě jsou specializovány na obraz. V této práci byly využity jen zde a bez hodnotných výsledků. Do sekce 2.4 tedy nebyl jejich popis zařazen

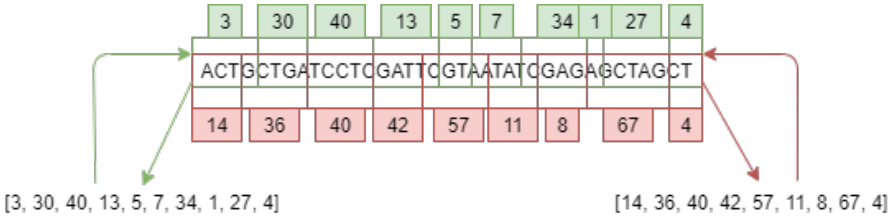
[3]Nulový vektor obsahující jednu hodnotu rovnou jedné.

[4]Embedding v neuronových sítích převádí diskrétní hodnotu do vektoru se spojitými hodnotami.

Classifier se SentencePiece

S aplikací modulu SentencePiece se už architektura zaměřila pouze na přístup využívající sítě LSTM, které se z předcházející úlohy jevily jako nejlepší. Pro zakódování sekvencí bylo nejprve třeba natrénovat model SentencePiece. Bylo tak provedeno na promotorech z *Funghi* datasetu obsahujícího data z 32 organismů s velikostí slovníku na 100 slov[5] (vzhledem k tomu, že v této aplikaci se nejedná o rozdělení na slova, ale na kousky sekvencí DNA, bude 'slovo' nahrazeno pojmenováním **piece**). Takto předtrénovaný model byl dále využíván i ve všech následujících typech sítí. V předchozím případě byla na začátku vygenerována pevná sada falešných promotorů, avšak se SentencePiece do kódu přibyla i funkce generátoru pro vytváření umělých promotorů. S využitím generátoru tak bylo možné při trénování předložit síti větší varianci umělých promotorů, neboť se při trénování pro každý batch[6] vytváří nová sada vstupních dat.

Kódování do SentencePiece není deterministické, pro dvě shodné vstupní sekvence se zakódováním vygenerují dva různé vektory, které po dekódování nesou opět stejnou informaci (viz obr. 3.7). Dle průběžných výsledků bylo patrné, že je třeba zmenšit velikost promotoru z původních tisíce bází, přičemž důležité oblasti promotoru se nacházejí až k jeho konci. Ořezávání promotoru tak probíhalo směrem od konce. Pro získání větší variance sekvencí jednotlivých pieců byly tisíci-bázové promotory (reálné i umělé) nejdříve náhodně oseknuty na posledních 450-500 bází, poté došlo k jejich zakódování do SentencePiece a následně k finálnímu ořízntutí na posledních 100 pieců. Jeden piece obvykle zakóduje přibližně tři báze, oseknutý promotor tak reálně obsahuje odhadem 300 posledních bází.



Obrázek 3.7: Schéma dvou zakódování stejné sekvence promotoru pomocí SentencePiece. Přestože jsou většinou IDs jednotlivých pieců v obou získaných vektorech rozdílná, při dekódování skládají oba vektory opět stejnou sekvenci.

Po různých pokusech generování umělých promotorů (viz obr. 3.6) byl nakonec vybrán přístup s nahrazováním části promotoru vygenerovanou sekvencí (obr. 3.6-A)). Výběr byl proveden na základě experimentů, kdy ani pro jeden ze zbylých přístupů (obr. 3.6-B,C)) nebyla vytvořená neuronová síť schopna detekovat narušení

[5]Od velikosti slovníku se odvíjela i velikost vstupní dimenze embeddingu v jednotlivých modelech. Vstupní dimenze embeddingu by měla být vyšší než velikost slovníku a byla nastavena na 102 nebo 105.

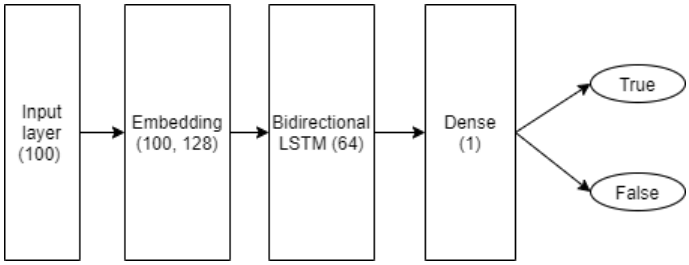
[6]Tento hyperparametr určuje pro kolik dat najednou se počítá gradient, podle něhož dochází k úpravě vah v metodě back-propagation.

promotoru a správně tak rozhodovat mezi promotory reálnými a umělými.

Podoba sítě nakonec dokonvergovala k poměrně jednoduchému modelu. Po vstupním embeddingu následovala bidirectional LSTM vrstva, za kterou už byl pouze jeden Dense neuron se sigmoidální aktivační funkcí (obr. 3.8). Ztrátová funkce byla opět zvolena *binary-crossentropy* se sledovanou metrikou *accuracy*. Výsledky takto sestaveného modelu byly srovnatelné se složitějšími architekturami s více LSTM vrstvami a navíc přidanými Dense vrstvami.

Pro reálné použití bylo třeba dosáhnout citlivosti na nepříliš veliké zásahy do promotoru. Na základě toho byly testovány různé délky vkládaných sekvencí. Pokud se vkládala do promotoru náhodná sekvence o délce 20-30 bází, byla přesnost klasifikátoru těsně nad 55 %. V případě záměny 30-40 bází docházelo ke klasifikaci s úspěšností kolem 68 % a pro záměnu 80-100 bází se dostala správnost klasifikace nad 85 %.

Pro potřeby diplomové práce nebylo třeba získat klasifikátor rozdělující vstupní sekvence přesně podle jejich labelu na reálné a umělé. Důležitější bylo naučit síť poznat strukturu promotoru a dokázat v ní vyhledat určité zásahy. Výsledky tohoto přístupu se spolu s jejich analýzou nacházejí v sekci 4.1.



Obrázek 3.8: Finální schéma modelu typu Classifier. V závorkách je uvedena dimenzionalita v jednotlivých vrstvách.

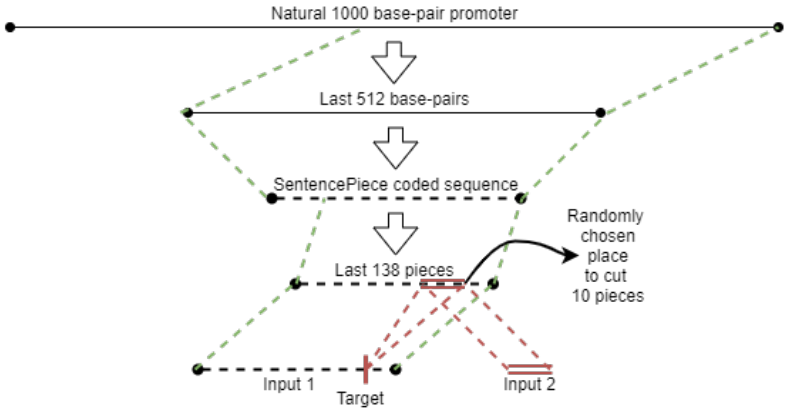
3.3.2 Place-back

Zatímco u typu *Classifier* se pomocí sítě získávalo ohodnocení věrohodnosti celého promotoru, u metod *Place-back* bylo úkolem predikovat správné umístění operátorů do přirozené sekvence promotoru. Jedná se o dvou-vstupovou neuronovou síť, kdy se dle funkčnosti dají modely rozdělit na dvě větve vycházející z dvou hlavních přístupů pro vkládání regulačních oblastí do promotorů: *Place-back-one-spot* (vložení operátoru) a *rewrite* (přepsání sekvence v promotoru). Díky větším rozdílům v implementaci je následně celý proces trénování, včetně přípravy dat popsán v jednotlivých podsekcích. Pro obě možnosti byla stejně jako u typu *Classifier* vyzkoušena práce s přirozenými sekvencemi promotorů a se zakódovanými promotory pomocí SentencePiece. Vzhledem ke kvalitě trénování se brzy přestalo úplně pracovat s modely s nezakódovanými sekvencemi do SentencePiece a dále už tak nebudou zmiňovány.

Place-back-one-spot

Prvním uvedeným modelem *Place-back* bude typ *one-spot*. Pojmenování vychází z funkce sítě, kdy se z promotoru vyjme sekvence, pro kterou se hledá index odpovídající místu vyříznutí (obr. 3.2). Síť se trénovala na trénovacím datasetu promotorů ze *Saccharomyces c.*

Vstupní data byla připravována pomocí generátoru, v průběhu trénování tak byla proměnná. Přirozená sekvence promotoru se nejprve ořízla na posledních 512 bází, následně se sekvence zakódovala pomocí SentencePiece a ořízla na 138 posledních pieců. Z tohoto vektoru pak bylo vyjmuto deset za sebou jdoucích pieců. Zbytek vektoru (128 pieců) sloužil jako jeden vstup sítě, vyjmutých deset pieců pak jako druhý vstup, který měl být namapován zpět do původního vektoru. Referenční informací byla na výstupu modelu hodnota odpovídající indexu v místě vyříznutí (obr. 3.9).

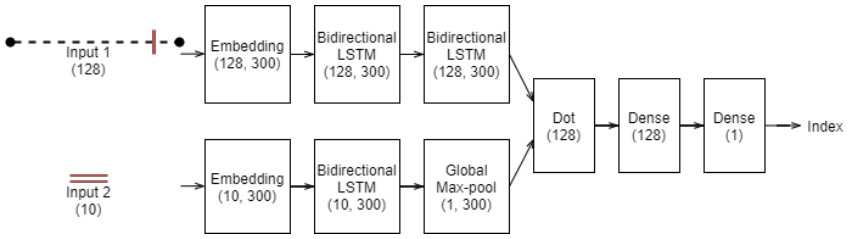


Obrázek 3.9: Vizualizace přípravy trénovacích dat pro metodu *Place-back-one-spot*.

Podoba modelu sítě je znázorněna na obrázku 3.10. Oba vstupy prošly nejprve embeddingem, sekvence promotoru poté přes dvě bidirectional LSTM vrstvy a vyříznutý fragment jednou bidirectional vrstvou, za kterou byl proveden globální max-pooling. Tím bylo docíleno, že vyříznuté piecy byly reprezentovány v každé dimenzi pouze jednou hodnotou. Reprezentace promotoru se následně přenásobuje s reprezentací vyříznutého fragmentu pomocí vrstvy počítající skalární součin (vrstva Dot). Výstupem vrstvy je tak 128 hodnot, které se posílají do Dense vrstvy se 128 neurony. Neurony jsou poté svedeny do jednoho výstupního neuronu, z nějž by měla vzejít hodnota ukazující na místo vyříznutí fragmentu. Cílem učení tak bylo minimalizovat vzdálenost hodnoty odhadu od reality. Jako ztrátová funkce byla využita střední kvadratická chyba a jako sledovací metrika střední průměrná chyba.

Výsledky pro tento typ modelu nebudou dále v sekci 4.1 prezentovány. Takto sestavená síť se nebyla schopna naučit vkládat vyříznuté místo zpátky do místa určení. Při daném způsobu učení se síť vždy dostala do lokálního optima, ze kterého se už nedokázala dostat. Optimum nalezené sítí byl střed promotoru, což ze sta-

tistického hlediska není překvapením. Výpovědní hodnotu pro danou úlohu to však nemá žádnou.



Obrázek 3.10: Schéma modelu Place-back - one-spot. V závorkách je uvedena dimenzionalita v jednotlivých vrstvách.

Place-back-rewrite

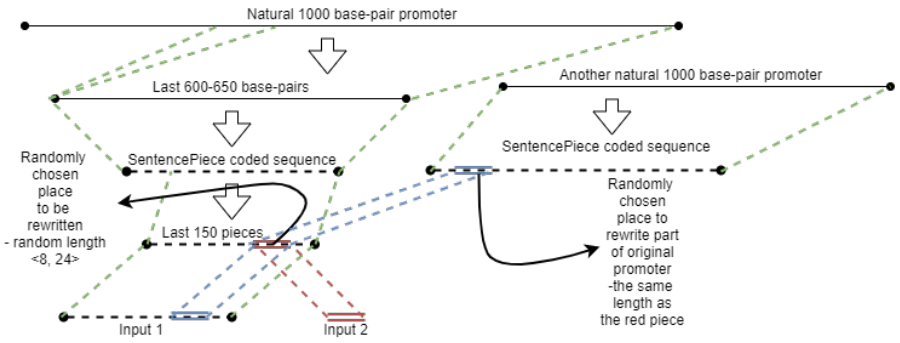
Druhým typ metody *Place-back* je nazvaný *rewrite*. Typy se vzájemně liší ve stylu úpravy promotoru. Metoda *one-spot* vyřezává kus sekvence, zatímco *rewrite* ji přepisuje (viz obr. 3.3). Trénování, potažmo zde použité předtrénování, probíhalo již na větším počtu datasetů. Kromě základního trénovacího datasetu z promotorů *Saccharomyces c.* byl dále využit dataset s promotory z 32 zástupců *Funghi*, dataset obsahující známé regulace z vytvořené databáze (podsekce 3.2.1) a EPD dataset.

Příprava vstupů byla podobná jako u typu *one-spot*. Data byla postupně vytvářena pomocí generátoru dle schématu 3.11. Přirozený promotor se nejprve ořízl na posledních 600-650 bází, které byly následně zakódovány pomocí SentencePiece a zkráceny na posledních 150 pieců. Z tohoto vektoru se vybrala sekvence 8-24 pieců[7], která se uložila stranou. Z cizího promotoru se náhodně vybrala stejně dlouhá sekvence, kterou se přepsalo vybrané místo v původním promotoru. Promotor s přepsaným místem se stal prvním vstupem sítě. Druhým vstupem byla poté vyjmutá sekvence, která se pomocí sítě mapovala zpátky na patřičné místo. Metodě *Place-back-rewrite* byla věnována při trénování největší pozornost a vznikly z ní tři typy částečně odlišných modelů. Ve všech případech byla použita ztrátová funkce *binary-crossentropy* a metriky *precision* (P), *recall* (R) a F1[8]. Sledovala se především metrika R, která byla využita během trénování pro ukládání nejlepšího dosaženého skóre ve validačních datech.

U prvního typu modelu je vše v podstatě stejné jako u *Place-back-one-spot*. Rozdíl je však v použití více dimenzí a jiné výstupní vrstvy. Architektura modelu se nachází na obrázku 3.12. Oba vstupy nejprve prošly embeddingem, poté sekvence promotoru prošla skrz dvě LSTM vrstvy a vyříznutý fragment přes jednu LSTM vrstvu a globální max-pooling. Dále došlo ke skalárnímu součinu obou větví. Výstupy se nakonec přefiltrovaly přes konvoluční vrstvu s velikostí okna 5 a sig-

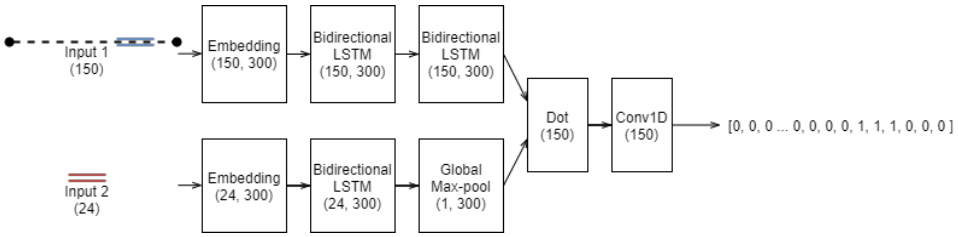
[7]Tato vybraná sekvence je druhým vstupem sítě. Aby byly všechny vektory vždy stejně dlouhé, vybraná sekvence se doplňuje nulami na délku vektoru 24.

[8]P=Precision=True Positives/(True Positives + False Positives),
R=Recall=True Positives/(False Negatives + True Positives),
F1=2*(P*R)/(P+R)



Obrázek 3.11: Vizualizace přípravy trénovacích dat pro metodu *Place-back-rewrite*.

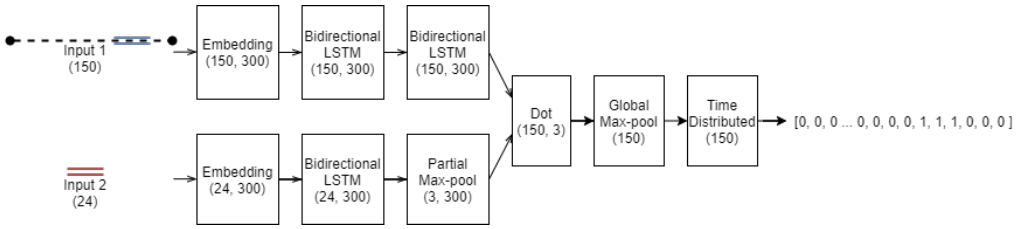
moidální aktivační funkcí. Na výstupu je tak vektor o 150 hodnotách v intervalu od nuly do jedné. Velikost výstupní hodnoty v každém indexu pak znamená, jak moc model věří, že vkládaný fragment patří na dané místo. Reprezentace zpětně vkládaného fragmentu je zde nastavena na jednu hodnotu, tato informace později slouží jako rozcestník pro rozlišování zbylých typů *Place-back-rewrite*.



Obrázek 3.12: Schéma modelu *Place-back-rewrite* s reprezentací vyříznutého fragmentu pomocí jedné hodnoty. V závorkách je uvedena dimenzionalita v jednotlivých vrstvách.

Druhý typ *Place-back-rewrite* se liší reprezentací vyjmuté sekvence (obr. 3.13). Větev sítě pro reprezentaci sekvence promotoru je stále stejná, kdy dojde k embeddingu a dvěma průchody LSTM vrstvami. Liší se tedy práce s vyříznutou sekvencí. Ta nejprve prochází stále stejně embeddingem a LSTM vrstvou, poté je na řadě max-pooling, který tentokrát není globální, ale částečný přes tři okna. Vyjmutá část je tak reprezentována třemi hodnotami oproti jedné, čímž se fragmentu a celkově síti přiřazuje vyšší vyjadřovací schopnost. Přenásobení obou větví pak proběhne stejným způsobem, kdy je Dot-produktem stále matice se šířkou tři. Poté následuje globální max-pooling, kterým už je získán vektor hodnot s délkou požadovaného výstupu. Na konci má každý neuron sigmoidální aktivační funkci použitou pro ztrátovou funkci *binary-crossentropy*.

Třetím vytvořeným typem *Place-back-rewrite* je metoda využívající skip-connections. Schéma modelu je stejné jako v předchozím případě (obr. 3.13) s výjimkou dvou LSTM vrstev pro práci se sekvencemi promotorů. Obě bidirectional LSTM vrstvy jsou nahrazeny rovněž bidirectional LSTM vrstvami, ovšem s rozdílem použi-

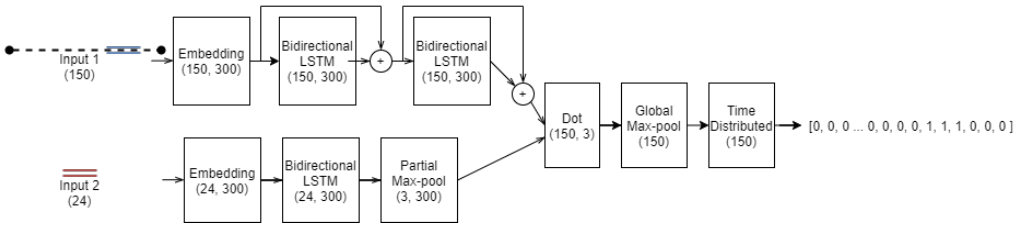


Obrázek 3.13: Schéma modelu *Place-back-rewrite* s reprezentací vyříznutého fragmentu pomocí tří hodnot. V závorkách je uvedena dimenzionalita v jednotlivých vrstvách.

tých přeskočení (reziduální blok) (viz obr. 3.14). Skip-connection se liší tím, že se síť neučí tradičně výstup $H(x)$, ale místo toho reziduum $R(x)$:

$$R(x) = Output - Input \rightarrow H(x) - x. \tag{3.1}$$

Díky metodě skip-connections je možné za sebe skládat větší množství vrstev bez toho, aby se dostavil nechtěný jev degenerace, kdy síť již není schopna zlepšit své predikce s přibývajícími vrstvami [74, 75].



Obrázek 3.14: Schéma modelu *Place-back-rewrite* s reprezentací vyříznutého fragmentu pomocí tří hodnot a s využitím skip-connections. V závorkách je uvedena dimenzionalita v jednotlivých vrstvách.

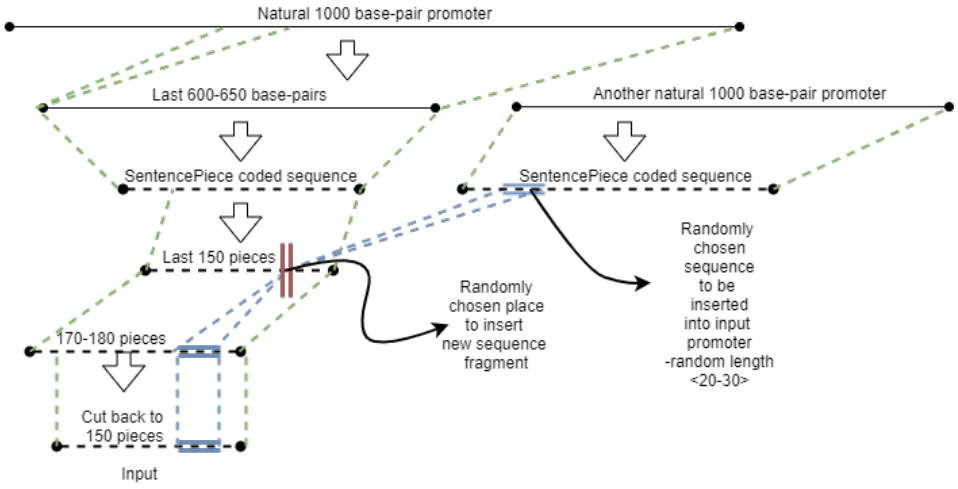
Všechny tři modely dosáhly použitelných výstupů. Bližší popis použitých datasetů a získaných výsledků je v sekci 4.1.

3.3.3 Insert-fragment

Posledním navrženým pokusem pro vkládání regulačních míst do promotorů je metoda nazvaná *Insert-fragment*. Formulace úlohy je vyobrazena na schématu 3.4. V tomto přístupu má neuronová síť za úkol detekovat cizí sekvence v promotoru. Vede to tak ke genetickým změnám, kdy se sekvence nepřepisuje jako v *Place-back-rewrite*, ale naopak se požadovaný fragment do promotoru vkládá.

Pro vytvoření vstupních dat byl opět využit generátor. Vizualizace přípravy vstupních dat je na obrázku 3.15. Z přírodního promotoru se ponechá posledních 600-650 bází, které se zakódují pomocí SentencePiece a získaný vektor se zastřihne

na délku 150 pieců. Zde se náhodně vybere místo, kam se vloží zakódovaná sekvence z cizího promotoru. Tímto krokem se promotor prodlouží. Následně dojde k dalšímu zkrácení na 150 pieců. Na výstupu se pak síť trénuje na nulovém vektoru s jedničkami v oblasti vložení cizí sekvence.



Obrázek 3.15: Vizualizace přípravy trénovacích dat pro metodu *Insert-fragment*.

Byla vytvořena řada modelů různě využívajících bidirectional LSTM vrstvy i se skip-connections, za kterými následovaly Dense vrstvy. V žádném z případů se však nepodařilo přiblížit očekávaným hodnotám. Ve velké většině pokusů se nastavily váhy po průchodu přes prvních několik batchů tak, že na výstupu byl vektor se samými nulami a síť už se z tohoto stavu nesnažila dostat. Z tohoto důvodu nejsou dané modely dále rozebírány a jejich výsledky se nebudou v sekci 4.1 nacházet.

Kapitola 4

Výsledky

Pomocí natrénovaných modelů neuronových sítí byly získány odhady pro vložení regulačních míst do promotorů, které byly integrovány do genomu organismu *Saccharomyes c.* Následně byly provedeny experimenty porovnávající úspěšnost provedených úprav. První část této kapitoly je zaměřena na seznámení s výsledky z neuronových sítí a popis promotorů a regulačních prvků, které byly vybrány na základě výstupů modelů. Druhou částí pak je popis postupu při vypracování experimentů společně s analýzou získaných výsledků. Nakonec je provedena diskuze shrnující spojitosti kolem vypracování experimentů s obdrženými výsledky.

4.1 Výsledky in silico

Natrénované modely neuronových sítí byly podrobeny testu, který ohodnocoval výstupy pro testovací dataset (popsáno na konci 3.2.2). Na základě těchto výsledků bylo pro laboratorní experimenty vybráno pět promotorů a dva represibilní operátory, které se do nich vkládaly. Pro experimenty byly zvoleny *Lac* operátor (nadále ***Olac***) a *Tet* operátor (nadále ***tet0***). Jejich bližší popis se nachází v podsekci 4.1.3. Z vybraných promotorů jsou čtyři konstitutivní (z článku (Lee et al. [14]) - *pHHF2*, *pPAB1*, *pPOP6*, *pREV1*), poslední zbývající je promotor *pX*. Regulační proteiny vázající se na vybrané operátory pocházejí z bakterií. Pro správnou funkčnost regulačních mechanismů je běžnou praxí, že uměle vložené proteiny a regulační místa jsou k cílovému organismu ortogonální. To znamená, že cokoliv nově vloženého do organismu by se v něm nemělo přirozeně vyskytovat, aby tím nebyly narušeny získané výsledky. Všechny zde zobrazené grafy se pohybují na ose y v intervalu $\langle 0, 1 \rangle$, kdy hodnota 1 vždy odpovídá maximálnímu ohodnocení, či věrohodnosti.

4.1.1 Analýza výsledků modelů neuronových sítí

Na základě výsledků trénování neuronových sítí byli pro testování vybráni nejlépe se jevící zástupci jednotlivých typů modelů. Jedná se o jeden model typu *Classifier*

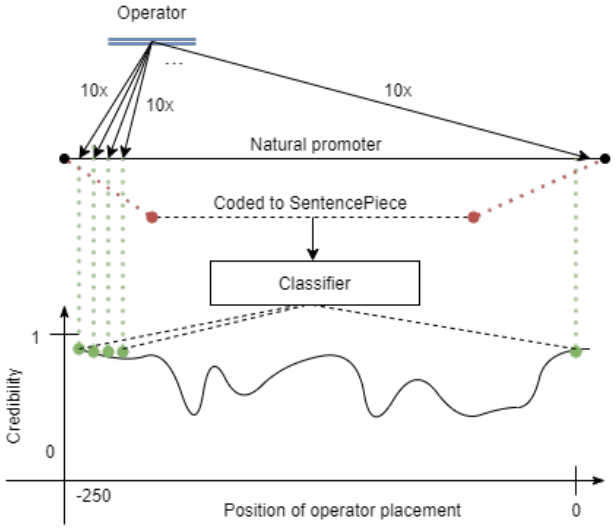
se SentencePiece a z *Place-back-rewrite* po jednom modelu z každého podtypu: reprezentace vyříznutého fragmentu jednou hodnotou, reprezentace vyříznutého fragmentu třemi hodnotami a metoda se skip-connections.

Classifier se SentencePiece

Použitý model pro testování byl natrénován rovnou na datasetu obsahujícím pouze promotory ze *Saccharomyces c.* Pro predikci byl využit dataset s konstitutivními promotory a *pX* a použité sekvence operátorů pro vkládání byly:

Olac-`TGGAATTGTGAGCGGATAACAATT`, *tet0*-`TCCCTATCAGTGATAGAGA`.

Příprava dat pro test probíhala dle schématu 4.1. Do promotoru se od určitého indexu vložil přepsáním operátor a takto upravený promotor byl vstupem natrénovaného modelu *Classifier*, na jehož výstupu bylo ohodnocení upravené sekvence. Tento proces byl proveden pro každý index v promotoru s desetinásobným opakováním. Důvodem opakování je nedeterminičnost SentencePiece, kdy stejné sekvence mohou být zakódovány odlišně a opakováním se pak zprůměruje odhadovaná hodnota. Vkládáním operátoru do promotoru index po indexu je pak možné sestavit graf znázorňující vývoj věrohodnosti dle místa vložení. Pro tyto grafy je na ose y věrohodnost promotoru a na ose x je pozice vložení operátoru. Ta odpovídá počátečnímu indexu v promotoru, kde započal přepis vkládaným operátorem. Osa x se pohybuje v záporných číslech, neboť pozice je vztažena ke konci promotoru, tedy ke start kodonu.

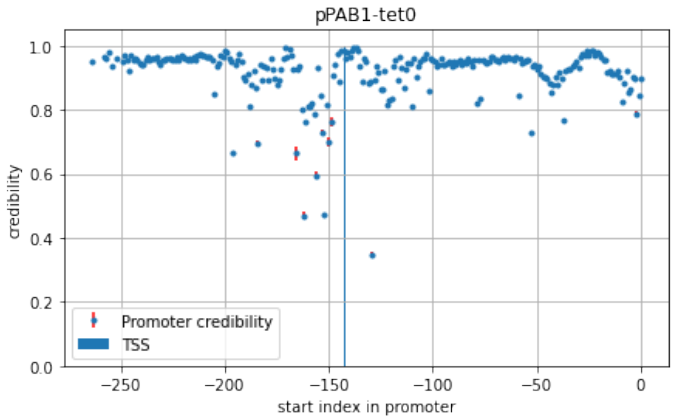


Obrázek 4.1: Schéma testování metody Classifier.

Stejný postup byl proveden na všech testovacích promotorech. Následné ukázky grafů tedy na ose y ukazují věrohodnost promotoru, kdy daným indexem začal přepis promotoru operátorem. Stejně jako při trénování, i tady musel mít vstup sítě délku 100. Z toho důvodu se na testování využilo z promotoru jen posledních

100 zakódovaných pieců, které po dekódování dávaly délku přibližně 250 bází. Nyní budou ukázány vybrané grafy získané při testování. Nadpisy jednotlivých grafů se skládají z názvu promotoru a operátoru, který se do něj vkládal.

První ukázkou je bodový graf pro vkládání *tet0* do promotoru *pPAB1* (obr. 4.2). Jedná se o graf zobrazující zároveň i varianci hodnot přes desetinásobné opakování pro vložení operátoru do stejného místa promotoru..



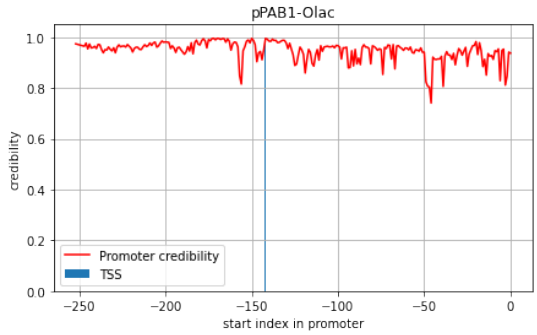
Obrázek 4.2: Bodový graf získaných výstupních hodnot pro postupné vkládání operátoru *tet0* do promotoru *pPAB1*.

Z grafu je patrné, že variance hodnot způsobená rozdílným zakódováním do SentencePiece je zanedbatelná a dále tak budou grafy pro lepší orientaci vykresleny spojitě (obr. 4.3). Zobrazené výsledky jsou pro náhodně vybrané konstrukty, jejich výběr tak nemá spojitost s výběrem promotorů pro reálné experimenty. Grafy pro zbylé konstrukty vypadají obdobně a lze tak zhodnotit, že takto sestavená a natrénovaná síť nezískala požadovanou schopnost určení místa pro vložení operátoru. Posouvání vloženého operátoru nemívá na ohodnocení velký vliv a charakteristikou je to více podobné šumu. Dalším problémem je, že síť predikuje ve všech promotorech možné změny i v oblasti TSS. Přepsáním nebo nadměrným zásahem do oblasti TSS se pak s vysokou pravděpodobností promotor zničí. Výše zmíněné faktory rozhodly o tom, že metoda *Classifier* se SentencePiece nebyla využita pro experimentální testy v laboratoři.

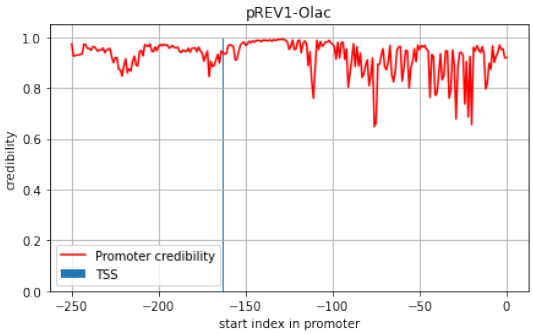
Place-back-rewrite - fragment reprezentovaný jednou hodnotou

Vybraný typ je jedním ze tří vytvořených *Place-back-rewrite* modelů. V tomto případě se jedná o možnost, kdy fragment vkládaný zpět do promotoru je reprezentován pouze jednou hodnotou (schéma modelu na obr. 3.12).

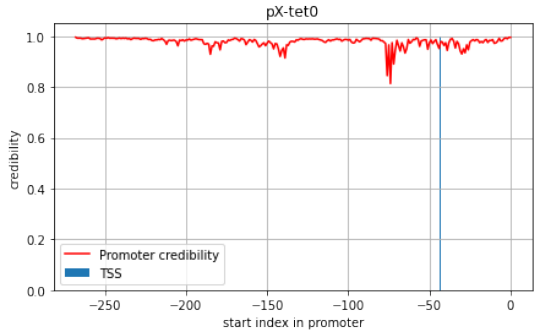
Neuronová síť byla předtrénována na datasetu z *Funghi* promotorů a dotrénována na trénovacím datasestu z promotorů *Saccharomyces c.* Pro predikci byl jako ve všech případech použit testovací dataset s konstitutivními promotory z článku (Lee



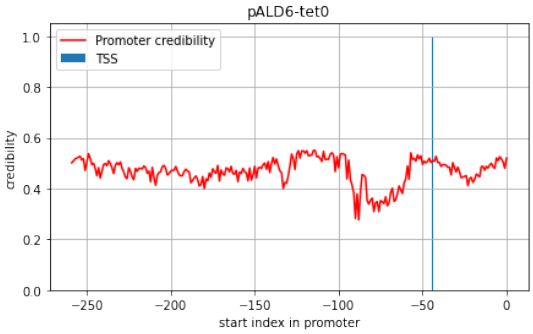
(a) Graf výstupních hodnot pro vkládání operátoru *tet0* do *pPAB1*.



(b) Graf výstupních hodnot pro vkládání operátoru *Olac* do *pREV1*.



(c) Graf výstupních hodnot pro vkládání operátoru *tet0* do *pX*.



(d) Graf výstupních hodnot pro vkládání operátoru *tet0* do *pALD6*.

Obrázek 4.3: Odhady neuronové sítě *Classifier* pro vkládání operátorů do vybraných promotorů.

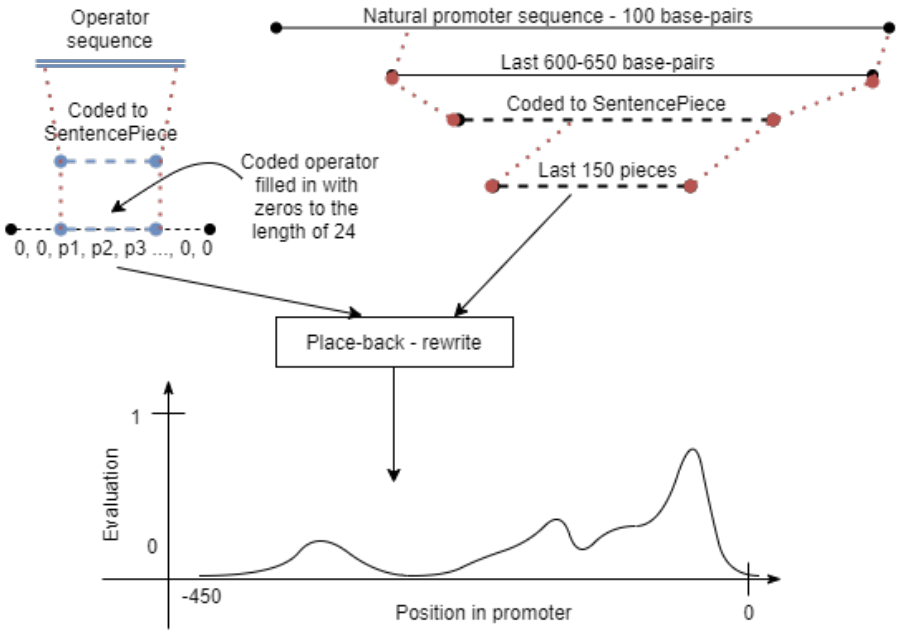
et al. [14]) a indukovaným promotorem *pX*. Příprava dat probíhala dle schématu 4.4. Tato architektura má dva vstupy, jedním z nich je promotor a druhým je sekvence operátoru. Před vložením na vstup sítě je promotor zkrácen na 600-650 bází, poté je zakódován pomocí modulu SentecePiece a následně oříznut na posledních 150 pieců. Sekvence operátoru je rovněž nejprve zakódována do pieců, které musí mít maximální délku 24. V případě zakódovaného vektoru s menší velikostí je sekvence pieců ze stran doplněna nulami na délku 24. Takto připravené vektory jsou vstupem sítě, která predikuje vektor délky 150 s ohodnocením jednotlivých pieců promotoru, jak na dané místo sedí vkládaná zakódovaná sekvence operátoru. Po zjištění délky jednotlivých zakódovaných pieců v promotoru na vstupu je výstup zpětně namapován na přirozenou nezakódovanou sekvenci. Zakódovaná sekvence na vstupu s délkou 150 pieců reprezentuje přibližně 450 nezakódovaných nukleových bází. Kvůli dříve zmíněnému nedeterministickému kódování pomocí SentencePiece bylo stejně jako v případě metody *Classifier* pro testování jednoho konstruktu využito replikátů. Promotor byl zakódován 40krát a operátor 5krát. Pro každý konstrukt tak bylo získáno 200 predikcí, které se nakonec zprůměrovaly. Popsaný postup byl takto proveden vzájemně pro všechny testovací promotory a oba vybrané operátory. Ve vytvořených grafech jsou poté na ose y hodnoty odpovídající přirozenosti vkládaného

operátoru na jednotlivé indexy promotoru, které jsou vyneseny na ose x. Indexování promotoru je opět v záporných hodnotách, neboť je vztaženo ke konci promotoru, tedy start kodonu. Použité sekvence pro operátory jsou v tomto případě:

Olac-`TGTTGTGTGGAATTGTGAGCGGATAACAATTTCACACA`,

tet0-`TCCCTATCAGTGATAGAGATCTCCCTATCAGTGATAGAGA`.

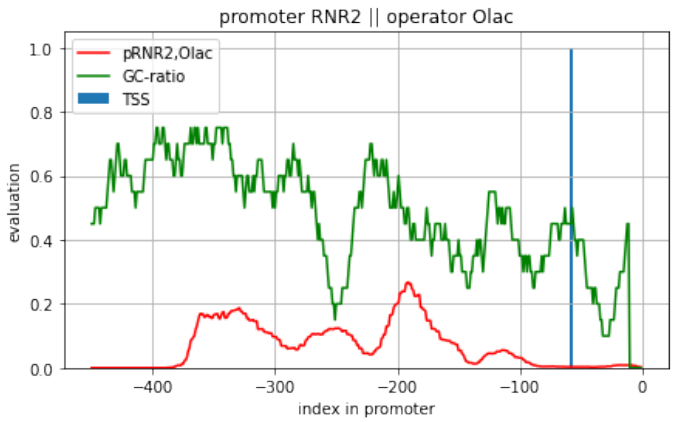
Popsaný postup testování v předchozím odstavci je shodný pro všechny typy *Place-back-rewrite*. V dalších případech tak bude na tento odstavec pouze odkazováno.



Obrázek 4.4: Schéma testování metody *Place-back-rewrite*.

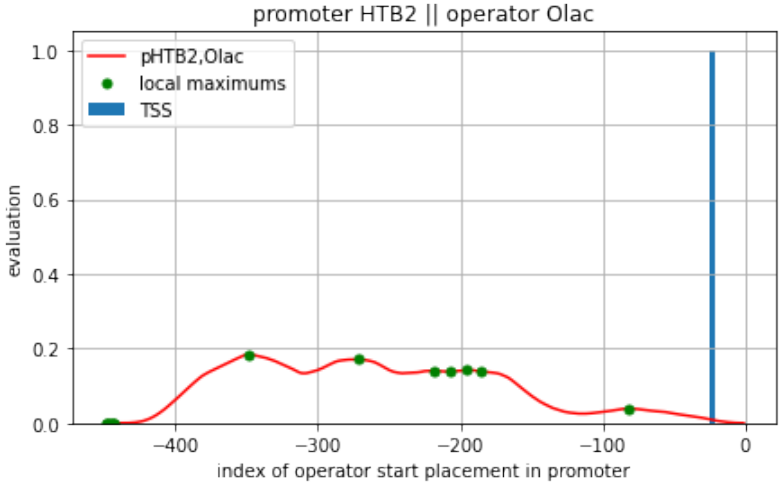
Následující ukázka grafu obsahuje neupravený výstup sítě pro vkládání operátoru *Olac* do promotoru *pRNR2* (obr. 4.5). Výstup modelu má červenou barvu. Vytvořené peaky udávají kandidátní místa, kam by síť operátor umístila. Zelená křivka ukazuje procentuální zastoupení bází `G` a `C` v okně velikosti 20 (GC-ratio). Křivka byla vytvořena jako kontrola, zda síť nevyužívá pro svoje rozhodování jen nějakou jednodušší informaci. Navíc, jak bylo zmíněno v sekci 2.1, nejsou pro přepisování vhodné oblasti s vyšší koncentrací bází `GC`. Z grafu je patrné, že k podobnému jevu nedochází.

Vynesená informace do dalších grafů je pozměněna. GC-ratio je vynecháno a křivka výstupu z neuronové sítě prochází konvolucí s jednotkovým oknem velikosti vkládaného operátoru. Výsledná podoba tak ukazuje místa, kde by mělo začít vkládání operátoru do promotoru, na rozdíl od předchozího zobrazení, které jen ohodnocuje jednotlivé indexy. Nejprve se zde nachází ukázka konstruktu, který z důvodu nepříliš vypovídajícího výsledku přes všechny modely nebyl vybrán pro pozdější experimenty (obr. 4.6). Následující sada grafů zobrazuje promotory, které



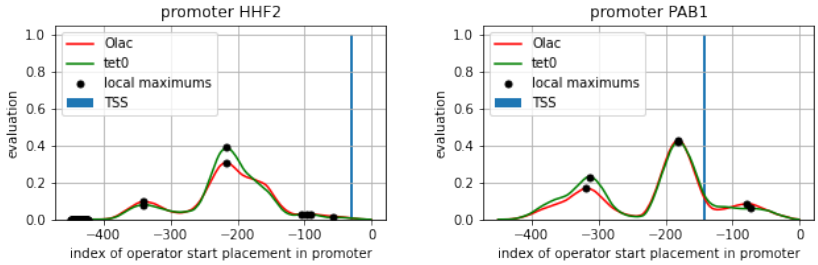
Obrázek 4.5: Graf obsahující výstup neuronové sítě typu *Place-back-rewrite* s reprezentací zpětně vkládaného fragmentu pomocí jedné hodnoty. V grafu je vynesen samotný výstup sítě (červená), GC-ratio (zelená) a TSS (svislá modrá linka).

již pro laboratorní experimenty byly vybrány (obr. 4.7). Výběr však neprobíhal na základě tohoto daného modelu, ale až toho následujícího. Tento model byl natrénován s výslednými metrikami: $P = 0.7231, R = 0.2475, F1 = 0.3646, P_{val} = 0.6331, R_{val} = 0.2416, F1_{val} = 0.3461$[1].



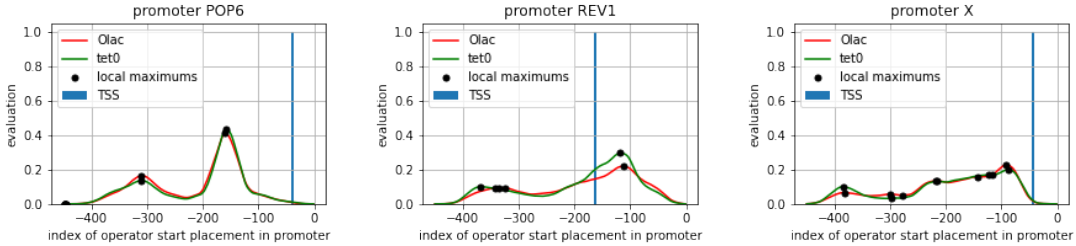
Obrázek 4.6: Ohodnocení míst v promotoru *pHTB2* pro počátek přepisu vkládaným operátorem *Olac*.

[1]Metriky X_{val} označují hodnoty pro validační data



(a) Ohodnocení míst v promotoru *pHHF2* pro počátek přepisu vybranými operátory.

(b) Ohodnocení míst v promotoru *pPAB1* pro počátek přepisu vybranými operátory.



(c) Ohodnocení míst v promotoru *pPOP6* pro počátek přepisu vybranými operátory.

(d) Ohodnocení míst v promotoru *pREV1* pro počátek přepisu vybranými operátory.

(e) Ohodnocení míst v promotoru *pX* pro počátek přepisu vybranými operátory.

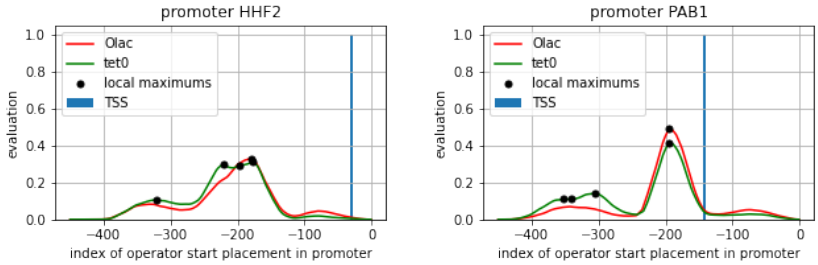
Obrázek 4.7: Sada vybraných promotorů pro laboratorní experimenty s predikovanými počátečními místy vložení operátorů *Olac* a *tet0*. Typ *Place-back-rewrite* s reprezentací vkládaného fragmentu jednou hodnotou.

Place-back-rewrite - fragment reprezentovaný třemi hodnotami

Hlavním rozdílem oproti předchozímu typu je reprezentace vkládaného fragmentu pomocí tří hodnot. Danou úpravou došlo i ke změně struktury sítě (viz obr. 3.13). Trénování modelu už však probíhalo shodně pomocí předtrénování na *Fungi* promotorech a dotrénování na promotorech *Saccharomyces c.*, kdy sledované metriky dosahovaly hodnot:
$P = 0.7281, R = 0.2723, F1 = 0.3925, P_{val} = 0.6085, R_{val} = 0.2675, F1_{val} = 0.3689.$
Příprava dat pro testování a úprava výstupů ze sítě probíhala totožně s předchozím typem (obr. 4.4). Rovnou se tedy přejde k výsledkům pro vybrané promotory (obr. 4.8).

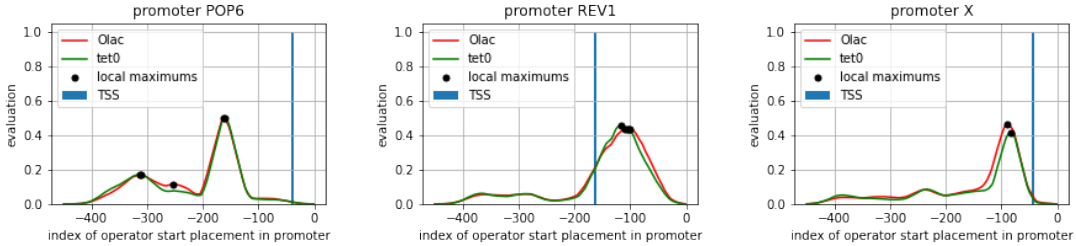
Při pohledu na grafy se potvrdil předpoklad ze sekce 3.3, kde bylo zmíněno, že reprezentací fragmentu pomocí tří hodnot se zvyšuje vyjadřovací schopnost sítě. Získané peaky mají jasnější charakter a výsledky tohoto přístupu byly dále využity pro laboratorní experimenty. Vzhledem k faktu, že jsou vždy odhady vložení obou operátorů do promotoru dost podobné, pro sestavení v laboratoři bylo vybráno v každém promotoru pouze jedno místo pro vložení obou operátorů. Na základě výsledků začal přepis promotoru operátorem pro *pHHF2* v bázi -177[2], *pPAB1* v bázi

[2]Promotory se přepisovaly ve směru k jeho konci, v grafech tedy k nule



(a) Ohodnocení míst v promotoru *pHHF2* pro počátek přepisu vybranými operátory.

(b) Ohodnocení míst v promotoru *pPAB1* pro počátek přepisu vybranými operátory.



(c) Ohodnocení míst v promotoru *pPOP6* pro počátek přepisu vybranými operátory.

(d) Ohodnocení míst v promotoru *pREV1* pro počátek přepisu vybranými operátory.

(e) Ohodnocení míst v promotoru *pX* pro počátek přepisu vybranými operátory.

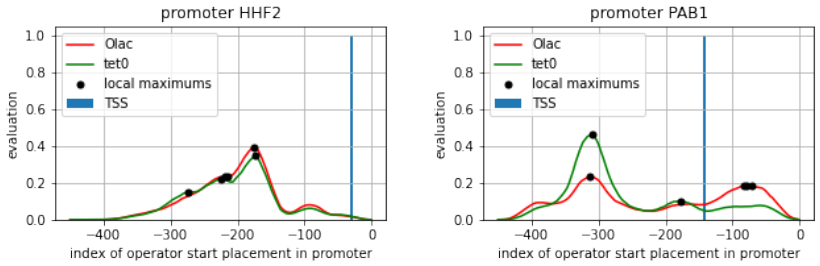
Obrázek 4.8: Sada vybraných promotorů pro laboratorní experimenty s predikovanými počátečními místy vložení operátorů *Olac* a *tet0*. Typ *Place-back-rewrite* s reprezentací vkládaného fragmentu třemi hodnotami.

-195, *pPOP6* v bázi -161 a *pREV1* v bázi -116. Jedinou výjimkou byl promotor *pX*, kdy se operátory vkládaly na různá místa. Operátor *Olac* se začal přepisovat na bázi -90 a *tet0* na -83. Průběh přípravy experimentu je popsán v sekci 4.2.1.

Place-back-rewrite - skip-connections

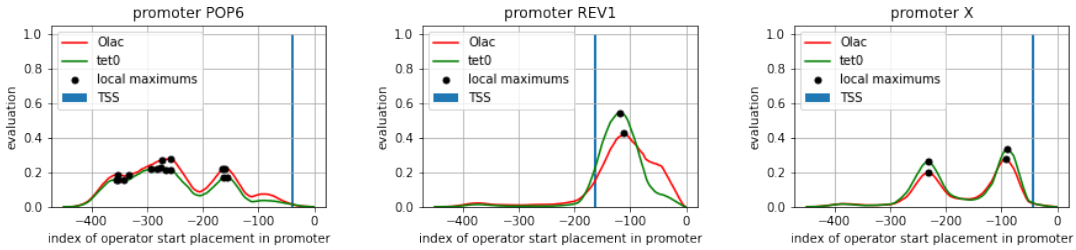
Poslední *in silico* testovanou metodou byl typ *Place-back-rewrite* s využitím skip-connections pro LSTM vrstvy. Využití této metody je jediným rozdílem oproti předchozímu typu. Model sítě je strukturovaný stejně, včetně reprezentace vkládaného fragmentu pomocí tří hodnot (viz obr. 3.14). Struktrura skip-connections umožňuje lepší vrstvení sítě a tím dosahuje lepší funkcionality. V době tvorby tohoto modelu již bylo jisté, že nebude možné získané výsledky otestovat v laboratoři. Nebylo tedy využito potenciálu skip-connections s mnohočetným vrstvením LSTM vrstev a pro práci s promotory byly ponechány pouze dvě LSTM vrstvy jako v předchozích typech. Důvodem byla snaha získat výstupy, které by se daly porovnat s již získanými výsledky. Trénování modelu včetně následné přípravy vstupních dat probíhalo opět totožně jako u typu *Place-back-rewrite* s reprezentací vkládaného fragmentu jednou hodnotou. Hodnoty pro měřené metriky zde byly:
$P = 0.7477, R = 0.3781, F1 = 0.4991, P_{val} = 0.5252, R_{val} = 0.3126, F1_{val} = 0.3903.$

Zpracované výstupy modelu se nacházejí v setu obrázků 4.9.



(a) Ohodnocení míst v promotoru *pHHF2* pro počátek přepisu vybranými operátory.

(b) Ohodnocení míst v promotoru *pPAB1* pro počátek přepisu vybranými operátory.



(c) Ohodnocení míst v promotoru *pPOP6* pro počátek přepisu vybranými operátory.

(d) Ohodnocení míst v promotoru *pREV1* pro počátek přepisu vybranými operátory.

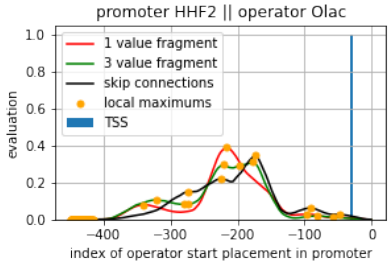
(e) Ohodnocení míst v promotoru *pX* pro počátek přepisu vybranými operátory.

Obrázek 4.9: Sada vybraných promotorů pro laboratorní experimenty s predikovanými počátečními místy vložení operátorů *Olac* a *tet0*. Typ *Place-back-rewrite* s reprezentací vkládaného fragmentu třemi hodnotami. Typ se skip-connections.

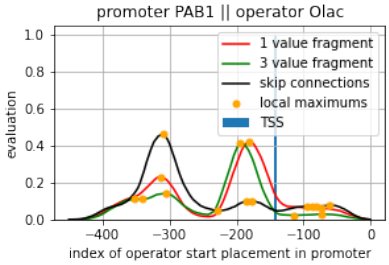
Place-back-rewrite - srovnání

Finální srovnání všech tří přístupů se nachází na sadách obrázků 4.10 a 4.11.

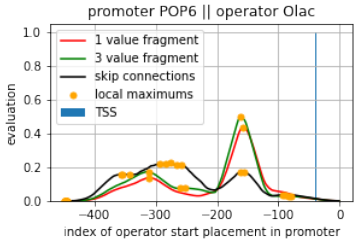
Ze získaných grafů je patrné, že se v predikcích jednotlivých typů vyskytují určité rozdíly. Většinou je tendence grafů velmi podobná, jen jsou odlišné hodnoty, kterých peaky dosahují. Přechod na reprezentaci vkládaných fragmentů třemi hodnotami posílil vyjadřovací schopnost sítě. V místech, kde předchůdce nevytvářel jednoznačná rozhodnutí, bylo dosaženo posíleného vrcholu. V případě metody se skip-connections docházelo k největším rozdílům ve srovnání se zbylými dvěma typy. Při zpětném pohledu na sadu obrázků 4.9 je však vidět, že byla získána i největší vnitřní variance mezi hodnotami pro vkládání dvou operátorů v rámci jednoho typu. Pomocí skip-connections bylo také dosaženo nejvyšší hodnoty metriky R. Zdá se tedy, že následné pokračování s typem skip-connections by mohlo vést k vylepšení dané metody pro vkládání regulačních míst do promotorů.



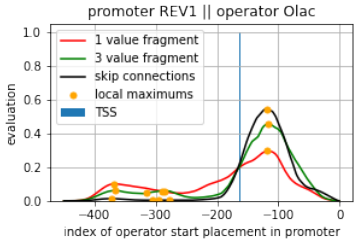
(a) Srovnání přístupů - *pHHF2*.



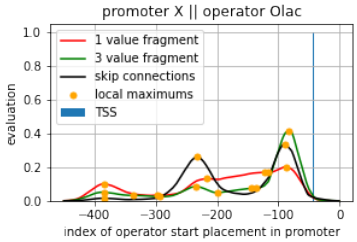
(b) Srovnání přístupů - *pPAB1*.



(c) Srovnání přístupů - *pPOP6*.

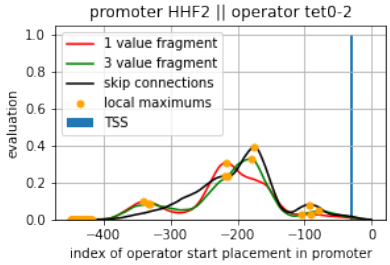


(d) Srovnání přístupů - *pREV1*.

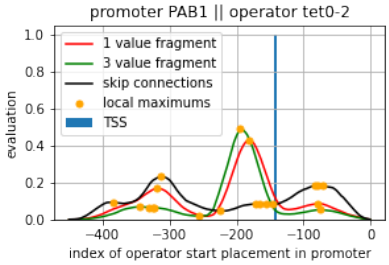


(e) Srovnání přístupů - *pX*.

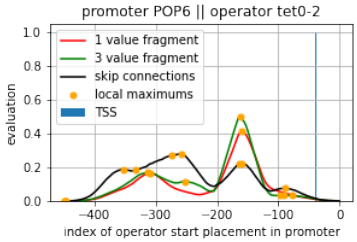
Obrázek 4.10: Porovnání přístupů *Place-back-rewrite* pro vybranou sadu promotorů s vkládáním operátoru *Olac*.



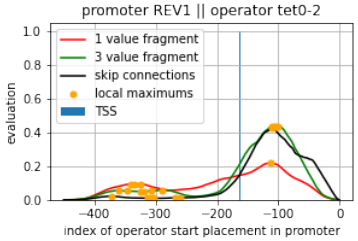
(a) Srovnání přístupů - *pHHF2*.



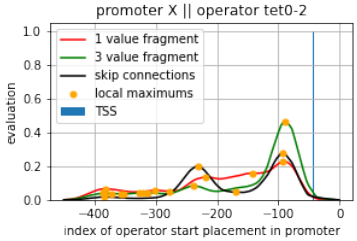
(b) Srovnání přístupů - *pPAB1*.



(c) Srovnání přístupů - *pPOP6*.



(d) Srovnání přístupů - *pREV1*.



(e) Srovnání přístupů - *pX*.

Obrázek 4.11: Porovnání přístupů *Place-back-rewrite* pro vybranou sadu promotorů s vkládáním operátoru *tet0*.

Použité skripty a datasety

Všechny použité datasety a skripty pro trénování neuronových sítí, zpracování a zobrazení výsledků jsou k dispozici na sdíleném úložišti Google Disk. Je zde možnost si prohlédnout i ostatní predikce pro konstrukty, které kvůli místu nebyly zobrazeny.

4.1.2 Analýza vybraných promotorů

Pro laboratorní experimenty byly na základě výsledků z neuronových sítí vybrány promotory *pHHF2*, *pPAB1*, *pPOP6*, *pREV1* a *pX*, přičemž první čtyři promotory jsou konstitutivní a jsou popsány v článku (Lee et al. [14]) a poslední, *pX*, je přirozeně indukovaný. Nejprve budou popsány konstitutivní promotory.

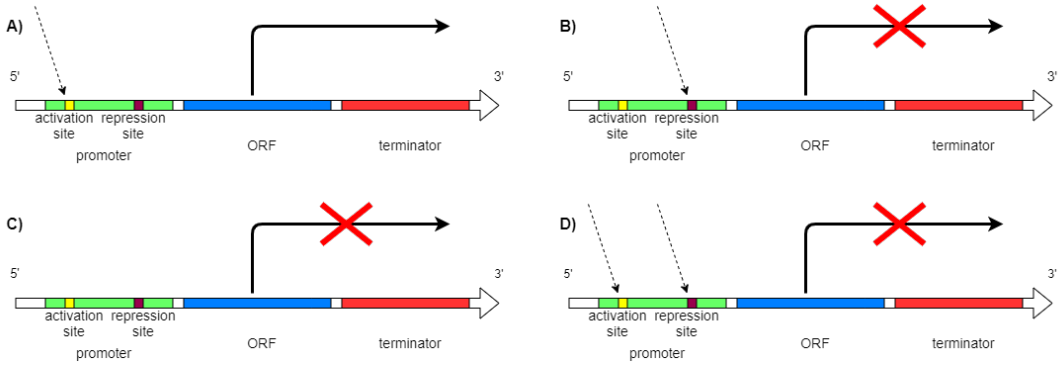
Na obrázku 4.13 se nachází znázornění měření relativní síly vybraných promotorů. Z pohledu na graf je patrné, že přes oba signalizační proteiny (mRuby2 a Venus) měly všechny promotory konzistentní relativní sílu. Promotory lze na základě experimentu (Lee et al. [14]) rozdělit z hlediska relativní síly na tři skupiny - silné (*pTDH3*, *pCCW12*, *pPGK1*, *pHHF2*, *pTEF1*, *pTEF2*), střední (*pHHF1*, *pHTB2*, *pRLP18B*, *pALD6*, *pPAB1*) a slabé (*pRET2*, *pRNR1*, *pSAC6*, *pRNR2*, *pPOP6*, *pRAD27*, *pPSP2*, *pREV1*). Zde testované promotory jsou:

pHHF2 - silný, *pPAB1* - střední, *pPOP6* - slabý, *pREV1* - slabý.

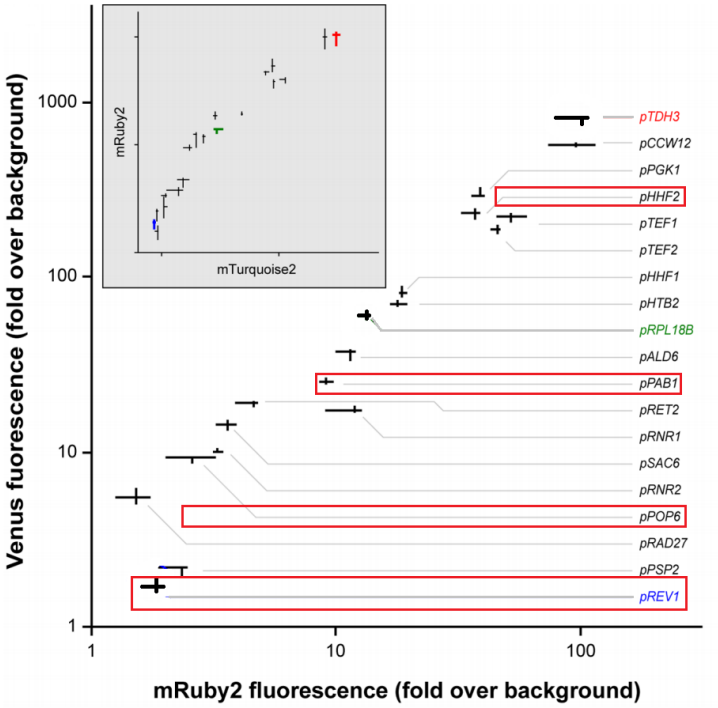
Vybrané promotory tedy pokrývají všechny tři skupiny. Při vložení represibilního operátoru do konstitutivního promotoru by v přítomnosti odpovídajícího proteinu mělo dojít k "vypnutí"promotoru a neměla by tak probíhat exprese genu. Bližší popis regulačního procesu pro vybrané operátory se nachází v sekci 4.1.3.

Pátý použitý promotor spadá do duševního vlastnictví společnosti XENO Cell Innovations s.r.o. a není tak možná bližší specifikace, než která zde bude uvedena. Promotor *pX* je relativní silou zařazen mezi silné a je přírodně indukovaný. Má v sobě tedy aktivační vazebné místo. Přidáním represibilního operátoru je tak možné dosahovat komplexnějšího řízení (viz obr. 4.12), než jak je tomu u konstitutivních promotorů, kde je při negativní regulaci možnost pouze represován/nerepresován. Jak je na konci podsekce 3.2.2 uvedeno, jedná se o promotor, ke kterému již existovaly výsledky na základě expertního návrhu regulace, kdy experimenty byly provedeny pro vkládání obou zde použitých operátorů.

Z hlediska regulace, v biologii neexistuje úplné "zapnuto"nebo "vypnuto", vždy bude docházet k nějaké transkripci kódující sekvence, i když je promotor "vypnutý". V případě silnějších promotorů by tak měl být rozdíl při správné regulaci výraznější než u promotorů slabších. Dalším parametrem je síla exprese kódující sekvence regulačního proteinu (transkripčního faktoru), který se stará o represi upraveného promotoru. Pro tyto účely byl pro oba transkripční faktory (*LacI*, *TetR*) vytvořen konstrukt s dvěma typy promotorů. Prvním je *pPOP6* (slabý) a druhým vybraným potom *pCCW12* (silný). V případě silného promotoru nacházejícího se před kódující sekvencí pro transkripční faktor by měla být míra represe cílového



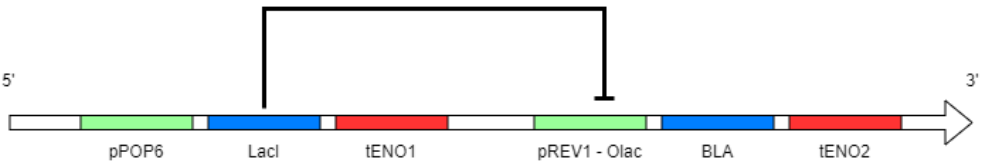
Obrázek 4.12: Schéma aktivace a represe promotoru. Čárkovaná čára označuje přítomný transkripční faktor. A) Přítomný pouze induktor → promotor je aktivní a dochází k transkripci. B) Přítomný pouze represor → promotor je vypnutý a nedochází k transkripci. C) Žádný transkripční faktor není přítomný → promotor je vypnutý a nedochází k transkripci. D) Přítomny oba transkripční faktory → promotor je vypnutý a nedochází k transkripci.



Obrázek 4.13: Relativní síla promotorů vyzkoušená přes dvě kódující sekvence signalizačních proteinů (z [14]). Vybrané promotory pro laboratorní experimenty jsou zvýrazněny v červených obdélnících.

promotoru účinnější. Pro sestavení experimentu je třeba do vybraného kmenu kvasinek vložit dva geny. První se stará o expresi transkripčního faktoru, ve druhém se nachází promotor s odpovídajícím operátorem, za kterým je kódující sekvence pro signalizační protein. V této práci byl použit protein *beta-laktamáza* (v originálu

beta-lactamase, dále zkráceně *BLA*). ***BLA*** se v reakci s nitrocefinem (chromogenní cefalosporinový substrát) zbarvuje dočervena, což lze pomocí speciálního čtecího zařízení měřit [76]. Pro měření *BLA* s nitrocefinem se využívá vlnová délka 486nm (viz (Chow et al. [77])). Naměřená hodnota poté slouží k identifikaci vlastností promotoru pomocí míry exprese *BLA*. Ilustrační schéma potřebného poskládání genů je vyobrazeno na schématu 4.14.



Obrázek 4.14: Schéma použité genetické regulace. První gen se skládá z promotoru *pPOP6*, ORFu pro *LacI* a terminátoru *tENO1*. Druhý gen se skládá z promotoru *pREV1* s vloženým represibilním operátorem pro *LacI*, z ORFu *BLA* a terminátoru *tENO2*.

4.1.3 Analýza vybraných regulačních prvků

Pro experimentální otestování natrénovaných modelů bylo využito dvou regulačních prvků - do vybraných promotorů vkládaných operátorů pro protein *TetR* (*tet0*) a pro protein *LacI* (*Olac*). Důvod této volby spočíval v dobré znalosti obou transkripčních faktorů a v laboratoři společnosti XENO Cell Inovations s.r.o. byla možnost připravené kódující sekvence již využít.

Protein LacI, operátor Olac

Protein *LacI* je bakteriální protein izolovaný roku 1966 [78]. Díky rozsáhlé kvantitativní charakterizaci regulace [79, 80] a charakterizaci vazby mezi zdatností a expresí operonu[3] [82, 83] slouží *Lac* operon jako paradigma pro genetické regulační systémy [84, 85]. Bakterie přednostně jako zdroj energie upřednostňují glukózu, při vyčerpání zásob se bakterie přepínají na alternativní zdroje uhlíku jako je například laktóza [47]. Když se v růstovém médiu stane laktóza primárním zdrojem uhlíku, dojde k indukci represoru *LacI*, což umožňuje transkripci tří genů koordinujících využití laktózy (*LacZ*, *LacY*, *LacA*) [86].

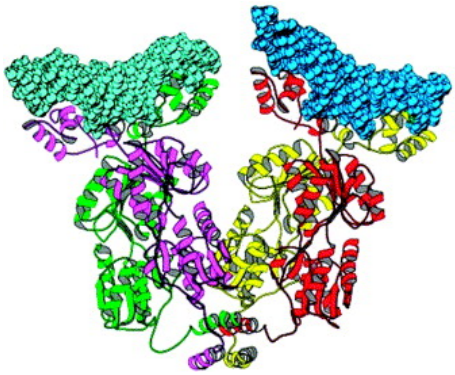
LacI je represující protein o délce 360 aminokyselin, který asociuje do homotetrameru[4], který je složen z dimeru dimerů (obr. 4.15) [87]. Každá jednotka tetrameru se skládá ze tří funkčních oblastí. První oblast je tetramerizační doména[5], která spojuje všechny podjednotky do funkčního komplexu, druhá je základní doména,

[3]Operon je řada po sobě jdoucích genů v prokaryotách, které mají společný promotor a jsou regulovány společným operátorem. Jsou také exprimovány najednou [81]. Transkripční faktor *LacI* reguluje promotor předcházející kódujícím sekvencím pro proteiny *LacZ*, *LacY* a *LacA*.

[4]Homotetramer je celek složený ze čtyř identických podjednotek.

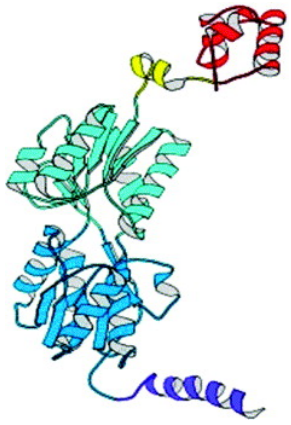
[5]Doména je funkční oblast proteinu

která se váže na laktózu a další podobné molekuly a třetí podjednotka je **head-piece**, což je doména vázající se na DNA (obr. 4.16) [88].



Obrázek 4.15: Vizualizace podoby homotetrameru proteinu LacI.

Sekvence operátoru byla původně považována za 24 bází dlouhou (5'-`TGGAATTGTGAGCGGATAACAATT` 3') [89]. Následně ukázalo, že minimální operátor pro navázání je dlouhý 17 bází uprostřed původní sekvence [90]. Operátor je přírodně pseudosymetrický. S vytvořením zcela symetrického fragmentu se afinita transkripčního faktoru *LacI* k operátoru zvýší desetkrát (5'–`TGTGGAATTGTGAGCGCTCACAATTCCACA` 3') [91].



Obrázek 4.16: Složení jedné podjednotky tetrameru *LacI*. Červená část je head-piece interagující s operátorem, žlutý je hinge (linker spojující head-piece s jádrem proteinu), u kterého se předpokládá specifická interakce s *Olac* pomocí natáčení head-piece. Dvě modré domény vážou cukr a fialová je terminální šroubovice (z [58]).

LacI klouže po DNA do nalezení svého vazebného místa, kam se naváže a ohnutím promotoru přerušuje transkripci. Krystalová struktura *LacI* ukazuje, že všechna čtyři vazebná místa tetrameru směřují jedním směrem (obr. 4.15), pokud jsou tedy oba dimery navázány na operátory, vytvářejí tak na DNA smyčku [85].

Přirozeným induktorem molekuly *LacI* je alolaktóza, která je vytvořena vedlejší reakcí laktózy s β-galaktosidázou [92]. Analogem alolaktózy je IPTG (*1-isopropyl-β-d-thiogalaktosid*) fungující stejným způsobem [86]. Po navázání induktoru na protein *LacI* dochází ke změně struktury, která má za následek snížení afinity transkripčního faktoru s cílovou DNA. Dochází tak k vypnutí represe. Když je tedy laktóza v buňce vzácná, *LacI* represuje geny za promotorem s *Olac*, protože nejsou potřeba. Když poté bakterie narazí na velký zdroj laktózy, *LacI* po navázání alolaktózy pozmění strukturu, což umožní produkci enzymů *LacZ*, *LacY* a *LacA*, které začnou využívat laktózu jako zdroj energie. Po vyčerpání zásob *LacI* ztrácí navázané cukry a začíná opět represovat expresi genů *Lac* operonu.

Opakem induktorů jsou anti-induktory, které zvyšují afinitu proteinu k vazebnému místu. Nejúčinnějším anti-induktorem je ONPF (ortonitrofenyl-β-d-fukosidem), anti-induktory však nemají zatím žádnou známou regulační funkci v *E.coli* [93].

Protein TetR, operátor tet0

Stejně jako *LacI*, *TetR* je bakteriální protein, který má dobře zmapovanou svoji funkci. Náleží do *TetR* rodiny proteinů čítající 2353 neredundantních zástupců, která je po něm díky nejpodrobnějšímu popisu pojmenována [21]. Tyto proteiny jsou součástí mechanismů efluxních pump[6], reakcí na osmotický stres, kontroly katabolických cest, diferenciace a dalších procesů.

TetR je protein tvořící homodimer (viz obr. 4.17) a řídící rezistenci vůči širokospektrým antibiotikům. Při navázání tetracyklinu[7] (dále **Tc**) dochází k deaktivaci ribozomu s následnou smrtí buňky [94]. Bakterie si vůči tomuto mechanismu vytvořily rezistenci, která funguje jako efluxní pumpa. Po vniknutí Tc do bakterie dochází k jeho chelataci s Mg^2 a vzniká $[MgTc]^+$ komplex, který se naváže na *TetR*, změní jeho strukturu a zabrání mu se jednoduše navázat na odpovídající operátor (viz obr.4.18) [94, 95]. Princip spočívá v tom, že na regulační doménu navázaný induktor nastavuje pohyb rozpoznávacích šroubovic víc od sebe a tím narušuje kontakty s DNA [96]. *TetR* se jinak váže na palindromický operátor (obr. 4.19), kde při navázání dochází k zalomení DNA [97].

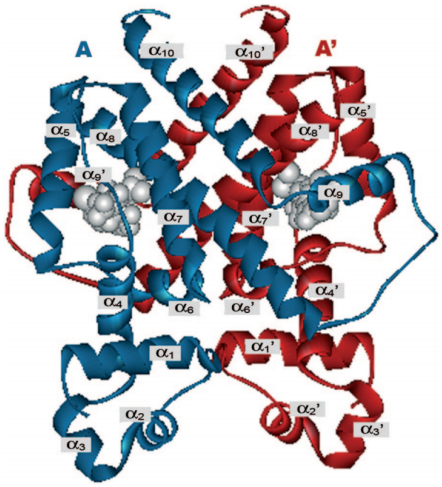
Struktura proteinu *TetR* může být rozdělena do dvou vazebných částí. První oblastí je N-terminální konec vázající se na operátory (z obr. 4.17 šroubovice α_1[8], α_2 a α_3) a core-doména, která se podílí na dimerizaci a obsahuje kapsu pro navázání Tc [95].

V případě, že je represor takto vypnutý, dochází k expresi proteinu *TetA*, který umí navázat Tc a odpravit ho ven z buňky [94]. Vzhledem k tomu, že i nízká úroveň exprese membránového proteinu *TetA* je pro bakteriální buňky nevýhodná, jak bylo prokázáno u *E.coli*, je represe tohoto genu velmi účinná, zatímco indukce dosti citlivá [18]. Experimenty na tomto operátoru ukázaly, že nejlepší výsledky mají přírodní

[6]Efluxní pumpa je substrátově specifický transportní mechanismus

[7]Tetracyklin je širokospektré antibiotikum.

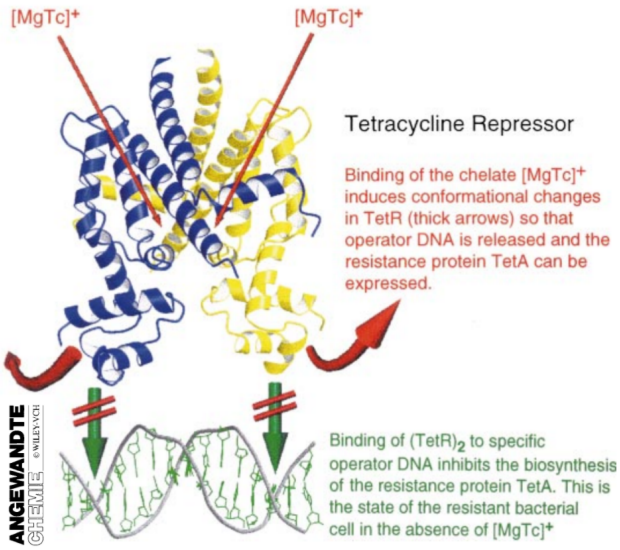
[8]Existují dva typy šroubovic lišící se ve směru zatočení. α-šroubovice má pravotočivý tvar, zatímco β-šroubovice má tvar levotočivý.



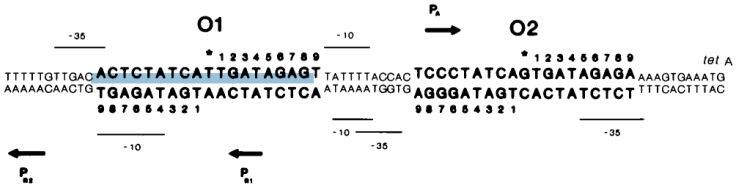
Obrázek 4.17: Podoba *TetR* homodimeru, kdy obě jednotky se skládají z deseti α-šroubovic (z [21]).

verze, kdy každá mutace způsobovala pokles úspěšnosti represe.

Zajímavé na tomto transkripčním faktoru je, že má i analogický protějšek [98]. Jedná se o reverzní *TetR* (*rTetR*), který je naopak aktivován za přítomnosti Tc. Výsledek byl získán náhodnými mutacemi v proteinu. Aminokyselina 71 je povrchovým zbytkem a byla vytvořena záměna: (Glu → Lys), Asp 95 spojuje čtecí hlavu DNA s jádrem proteinu (Asp → Asn), Leu 101 se podílí na dimerizaci podjednotky (Leu → Ser) a Gly 102 (Gly → Asp) sousedí s aminokyselinou, která kontaktuje Tc. Přítomnost transkripčních faktorů *TetR* a *rTetR* (reverse *TetR*) tak dává možnost regulovat dvě sady genů pomocí jednoho induktoru.



Obrázek 4.18: Ilustrace působení komplexu $[MgTc]^+$ na dimer *TetR* (z [94]).



Obrázek 4.19: Sekvence wild-type *TetR* operátoru s dvěma typy operátorových sekvencí (z [99]).

4.2 Výsledky in vivo

Na základě výstupů vybraného modelu neuronové sítě bylo vybráno pět promotorů a dva represibilní operátory pro sestavení experimentu. V následujících sekcích se bude postupně nacházet seznámení s přípravou experimentu, s jeho výsledky a na konec proběhne o získaných výsledcích diskuze shrnující vyvozené závěry.

4.2.1 Materiály a metody

Struktura experimentu byla vyhotovena na základě článku (Lee et al. [14]) probíraného v podsekci o Molecular Cloning 2.5.1. Zde se bude nacházet zkráceně popsaný průběh přípravy experimentů a měření. Podrobný popis se nachází v příloze *Materiály a metody* 5.

Zdroji pro vyhotovení experimentu byly kvasinkový kmen S0 (genotyp v tabulce 5.1), bakteriální vektor pro sestavení part-plazmidů, samotné part-plazmidy obsahující potřebné přírodní promotory, kódující sekvence pro proteiny a terminátory. Posledním potřebným zdrojem byly pro amplifikaci a úpravu DNA primery. Kromě primerů byly všechny použité materiály pro experimenty poskytnuty společností XENO Cell Inovations s.r.o.

Ze zdrojových part-plazmidů se sekvencemi promotorů byl vyamplifikován metodou PCR potřebný fragment DNA, který se domestikoval do bakteriálního vektoru, čímž vznikl nový part-plazmid. Metodou PCR byly ze dvou primerů vytvořeny sekvence operátorů, které byly také domestikovány. Potřebné part-plazmidy byly následně pomocí Golden Gate assembly sestaveny do genu, vloženého do klonovací kazety. Použité kazety obsahovaly informace pro namnožení plazmidu v bakteriích, což bylo po assembly využito. Po namnožení se vyčištěné plazmidy transformovaly do genomu kvasinkového kmenu S0.

Měřící protokol je také podrobně popsán v příloze *Materiály a metody* 5. V průběhu měřících experimentů bylo využito více nastavení parametrů. Hlavním rozdílem v měření byla hodnota $OD600$, na kterou se ředil vzorek před inkubací předcházející měření. Další úpravou pro finální nastavení bylo prodloužení inkubační doby před měřením ze 2 hodiny na 4 hodiny pro vzorky s variacemi promotorů *pPOP6* a *pREV1*.

4.2.2 Výsledky experimentu

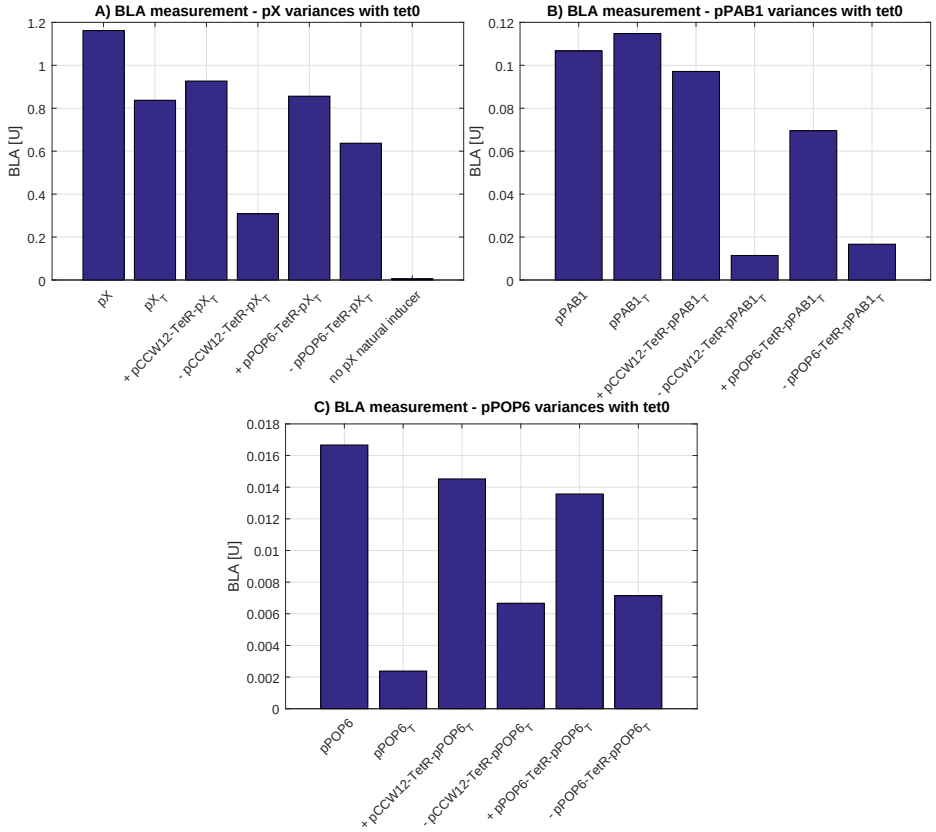
V průběhu měření experimentálních výsledků bylo využito řady nastavení měřícího protokolu, které vedly k získání mezivýsledků uvedených v příloze *Měření* 5. Zde budou pro přehlednost uvedeny pouze zajímavé výsledky práce s finálním nastavením měřícího protokolu.

Měření nejprve probíhalo pro všechny vzorky s totožným nastavením časů a ředění. To se ukázalo jako nevyhovující, neboť pro slabé promotory *pPOP6* a *pREV1* takto nebyl získán žádný signál. Po přenastavení hodnot ředění jednotlivých vzorků již byly získány reprezentativní hodnoty pro všechny měřené kmeny. Finální nastavení měřenícího protokolu bylo následující: vzorky s promotory *pX* a *pHHF2* byly naředěny na $OD600 = 0.02$, pro *pPAB1* na $OD600 = 0.1$ a pro slabé promotory *pPOP6* a *pREV1* na $OD600 = 0.2$. V případě slabých promotorů byla zároveň zvýšena inkubační doba ze 2 hodin na 4 hodiny.

Pro měření kmenů obsahujících regulaci *LacI* nastaly při tvorbě experimentu problémy, které způsobily znehodnocení výsledků. Tyto vzorky se tak zde nenacházejí a jsou uvedeny pouze v příloze *Měření* 5. Výsledky zde uvedené se týkají pouze kmenů s *TetR* regulací, ze kterých jsou vyfiltrovány měření pro promotory *pHHF2* a *pREV1*. U *pHHF2* byla důvodem nenaměřená represe a u *pREV1* došlo celkově k problémům se zaznamenáním měřitelného signálu. Výsledky pro *pHHF2* a *pREV1* jsou rovněž pouze v příloze *Měření* 5.

Výsledky experimentů jsou tak zaměřeny na promotory *pX*, *pPAB1* a *pPOP6*, do kterých se vkládal operátor *tet0*. Pro každou sadu reprezentovanou jedním typem upraveného promotoru probíhalo měření se šesti vzorky. Prvním byl přírodní promotor s *BLA* sekvencí, druhým pak upravený promotor s vloženým operátorem *tet0* následovaný *BLA* sekvencí. Zbylé vzorky již obsahovaly jak upravený promotor s *BLA*, tak gen obsahující regulační protein *TetR* exprimovaný promotorem *pCCW12* nebo *pPOP6*. Pro kmeny obsahující oba geny (*pCCW12/pPOP6-TetR+pYYY*$_T$) pak byly v měření vyhrazeny dva vzorky. Jeden s přidaným induktorem doxycyklin a druhý bez. V grafech je přítomnost induktoru ve vzorku označena přidáním do názvu '+', zatímco s nepřítomností induktoru byl do názvu přidán znak '-'. Kde nebylo přidávání induktoru relevantní se žádný z těchto znaků v názvu nevyskytuje. Očekávané pořadí naměřené hodnoty *BLA* [U] se shoduje s pořadím, jak jsou zde vzorky popsány (nativní promotor→upravený promotor→upravený promotor s regulačním genem a induktorem→upravený promotor s regulačním genem bez induktoru).

Měření sad promotorů probíhalo odděleně. Výstup experimentů je tak rozdělen do tří grafů, kde v každém grafu je měření pro jednu sadu promotorů (graf 4.20). Pro promotor *pX* se zde nachází i kontrola, kdy do vzorku nebyl přidán induktor potřebný pro zapnutí *pX*.



Obrázek 4.20: Graf finálního měření produkce *BLA* pro kmeny regulované proteinem *TetR*. A) Variance promotoru *pX*. B) Variance promotoru *pPAB1*. C) Variance promotoru *pPOP6*. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-' a kde nebylo přidání induktoru relevantní není v legendě navíc nic.

4.2.3 Diskuze

Znatelná represe byla experimentálně dosažena ve třech případech úpravy promotoru, zatímco ve zbylých sedmi ne. Nyní dojde k popisu výsledků s rozborem okolností, které ovlivnily získané výstupy.

Rozbor experimentálních výsledků

Zajímavé úpravy promotoru byly provedeny vkládáním *tet0* do *pX*, *pPAB1* a *pPOP6*. Ve všech případech se naplnil předpoklad z podsekce 4.1.2. Kmeny se silným promotorem *pCCW12* před kódující sekvencí pro regulační protein dosahují stejné nebo lepší represe než kmeny se slabým promotorem *pPOP6*.

Nejvíce byl tento efekt znatelný při represi silného promotoru *pX*. Míra represe naměřená pro sestavu s promotorem *pCCW12* předcházejícím regulačnímu proteinu *TetR* byla 66.6 %, zatímco pro *pPOP6-TetR* to bylo v porovnání jen 22.9

%. Všechny zaznamenané míry represe pro upravené promotory jsou sdruženy v tabulce 4.1. Zajímavé je zde i srovnání s kmenem S603. Jedná se o totožný kmen jako pPOP6-TetR-pX$_T$ s rozdílem, že operátor *tet0* byl do promotoru vložen expertně (graf 5.10). S expertním vložením bylo dosaženo 27.5% míry represe, přičemž pomocí odhadu neuronové sítě to bylo zmíněných 22.9 %.

Nejlepších výsledků bylo dosaženo pro kmeny obsahující *tet0* vložený do promotoru *pPAB1*. Hodnoty pro represovaný promotor jsou v tomto případě srovnatelné. Rozdílem však je vývoj při přidání induktoru doxycyklin. V přítomnosti induktoru by mělo zpětně dojít ke skoro plné expresi *BLA*, což se stalo pouze u kmenu *pCCW12-TetR-pPAB1*$_T$ a u *pPOP6-TetR-pPAB1*$_T$ nikoliv. Toto snížení exprese v přítomnosti induktoru bylo pro stejný kmen pozorováno i v předchozích měřeních. Vysvětlením chování může být bodová mutace v kmenu *pPOP6-TetR-pPAB1*$_T$. Mutace mohla nastat buď v upraveném promotoru *pPAB1-tet0*, kde by došlo k jeho narušení a celkově k horší funkcionalitě, nebo v kódující sekvenci pro *TetR* v oblasti, kam se váže induktor. Mutací v tomto místě by pak mohlo dojít horšímu vázání induktoru, čímž by se zvýšila šance, že regulační protein nasedne na vazebné místo v cílovém promotoru a bude ho represovat. I tak však byla míra represe v kmenu *pPOP6-TetR-pPAB1*$_T$ 76.0 %, zatímco pro *pCCW12-TetR-pPAB1*$_T$ to bylo 88.3 %.

Posledním promotorem se znatelnou represí byla verze *pPOP6* s vloženým operátorem *tet0*. Míra represe zde byla téměř shodná pro oba kmeny. Pro kmen *pCCW12-TetR-pPOP61*$_T$ dosahovala 53.8 % a pro *pPOP6-TetR-pPOP61*$_T$ 47.8 %. Promotor *pPOP6* je velice slabý, není tedy překvapením, že k jeho represi nebylo třeba mít silnější promotor před kódující sekvencí pro *TetR*. Zvláštní chování však bylo pozorováno pro kontrolní vzorek s upraveným promotorem a *BLA*, kde není přítomen regulační protein a hodnoty by se tak měly pohybovat například kolem hodnoty pro + *pPOP6-TetR-pPOP6*$_T$. Kmen byl však sestavován zvlášť a mohlo zde dojít k nějakému problému, který neodhalil test genomickou extrakcí.

Kmen	BLA [U] +	BLA [U] -	míra represe [%]	násobně nižší exprese při represi
pCCW12-TetR-pX$_T$	0.9267	0.3086	66.6 %	3.003x
pPOP6-TetR-pX$_T$	0.8256	0.6371	22.9 %	1.296x
pCCW12-TetR-pPAB1$_T$	0.0971	0.0114	88.3 %	8.518x
pPOP6-TetR-pPAB1$_T$	0.0695	0.0167	76.0 %	4.162x
pCCW12-TetR-pPOP6$_T$	0.0145	0.0067	53.8 %	2.164x
pPOP6-TetR-pPOP6$_T$	0.0136	0.0071	47.8 %	1.915x

Tabulka 4.1: Tabulka výsledných represí pro upravené promotory s úspěšnou regulací.

Provedenými experimenty bylo důležité otestovat také standardní funkčnost modifikovaných promotorů, zda-li úpravami nedošlo k jejich přílišnému poškození. Ve většině případů nedošlo k velkému narušení promotoru. V některých případech byly paradoxně naměřeny i vyšší hodnoty exprese pro upravený promotor, než pro přírodní. V první fázi experimentů bylo prováděno měření velkého množství vzorků

najednou, což mohlo danou chybu do měření zanést. Druhé možné vysvětlení se nachází níže u popisu potenciálních vylepšení modelů a technik trénování neuronových sítí. V jednom případě je však otázka funkčnosti upraveného promotoru nejistá. Pro kmeny s promotorem *pREV1* nebyly v původních měřeních zaznamenány žádné hodnoty. Při měření s finálním nastavení protokolu byl získán určitý nárůst měřené hodnoty pro nativní promotor s *BLA*, ale pro ostatní vzorky už v podstatě nic. Promotor *pREV1* je však tak slabý, že i drobné narušení mohlo způsobit v podstatě neměřitelné hodnoty *BLA*. Pro otestování variace *pREV1* promotorů by bylo lepší místo *BLA* použít citlivější signalizační protein. Jedná se například o luminescenční enzym luciferázu, která umožňuje produkovat světlo světluškám. Z časového hlediska však nebylo možné tento konstrukt v laboratoři otestovat.

Pro analýzu problematických promotorů je dobré se zpětně podívat na využité odhady neuronové sítě, ze kterých se experimenty sestavovaly (obr. 4.8). Je patrné, že umístění operátorů do *pHHF2* probíhalo v největší vzdálenosti od TSS. Tento fakt mohl způsobit, že operátor neměl vliv na míru transkripce a nijak tedy nerepresoval promotor. Pro promotory *pX, pPAB1, pPOP6* byly operátory umístěny v menší blízkosti nad TSS a mohly tak lépe blokovat navázání RNA polymerázy na promotor. V případě *pREV1* proběhlo umístění operátorů do oblasti za TSS, pro dané konstrukty však nebyla získána úplně průkazná data o jejich vlastnostech.

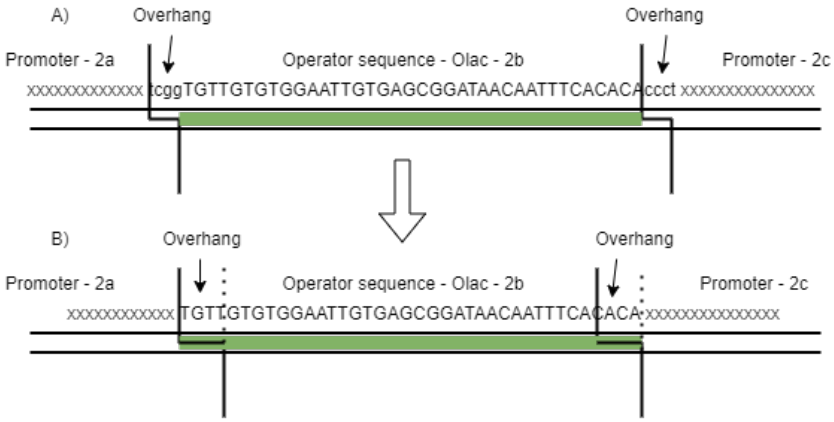
Experimentálně získané laboratorní výsledky ukázaly, že použitá metoda má jistě potenciál pro úspěšné navrhování genetických úprav. V této práci byla provedena první iterace experimentů tohoto druhu, která může ukázat směr pro případné další pokusy. Kvalita regulace zde nedosahovala úrovně expertního návrhu. Šlo však vyloženě o jeden pokus vložení operátoru do promotoru, přičemž při expertním návrhu se většinou provede několik různých umístění, ze kterých se poté vybírá ten nejlépe fungující konstrukt.

Možná vylepšení experimentů

V průběhu experimentu se ukázala dvě možná vylepšení, která by měla vést v případě opakování k získání lepších výsledků.

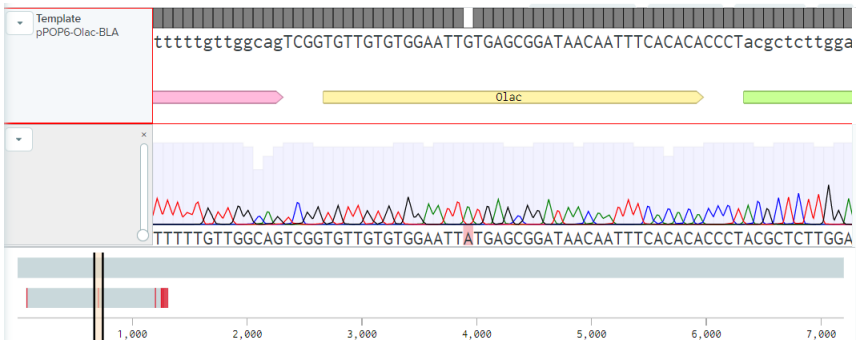
První nedostatek byl vytvořen při návrhu sestavení promotorů z částí 2a, 2b, 2c (viz obr. 4.21). Pro sestavení promotoru byly pro umístění operátorů využity overhangy, které se v původní sekvenci nevyskytovaly. Důvodem tohoto návrhu byla modularita vkládání pro oba typy operátorů bez zásahu do jedné z jejich sekvencí. Zároveň byly pro tento účel použity vyzkoušené overhangy z článku (Lee et al. [14]). Jednalo se tak zároveň i o bezpečnější řešení než při použití úplně nových overhangů. Přesto by však pravděpodobně bylo lepší mít overhangy rovnou jako součást sekvence operátoru. Čtyři přidané báze z každé strany operátoru mohly způsobit nepřirozené zkroucení DNA a bránit tak správnému přichycení regulačního proteinu na vazebné místo.

Druhým a pravděpodobně zásadním problémem pro většinu kmenů s vloženým operátorem *Olac* byla bodová mutace vzniklá v místě operátoru (viz obr. 4.22). Bohužel se tato mutace vyskytla v konstruktech se všemi promotory, které při vložení



Obrázek 4.21: A) Schéma použitého spojení částí 2a, 2b, 2c. B) Schéma možného spojení částí 2a, 2b, 2c bez rozšíření sekvence o báze v podobě overhangů.

operátoru *tet0* dosahovaly znatelných hodnot represe (*pX*, *pPAB1*, *pPOP6*). Paradoxně v promotorech *pHHF2* a *pREV1* k dané chybě nedošlo. Příprava všech upravených promotorů probíhala z jedné zkumavky s vyextrahovaným part-plazmidem z bakterií. K odhalení mutace pomocí sekvenace však došlo v době, kdy nebylo možné již způsobenou chybu v potřebném čase napravit.



Obrázek 4.22: Ukázka výsledku sekvenace promotoru *pPAB1-Olac*.

Kombinace obou problémů pravděpodobně vedla k tomu, že pro kmeny s regulací pomocí *LacI* nebyly získány žádné použitelné výsledky. Ukázkou potenciálu použité metody pro vkládání regulačních míst do promotorů však může být následující informace. Expertně navržený konstrukt pX_L-BLA_{lab}, který byl zde součástí kmenu S601, S602 (změřené hodnoty v grafeu 5.10) má vložený operátor *Olac* posunutý pouze o 12 bází ve směru od konce promotoru. Výsledky jakých dosáhly kmeny S601, S602 přitom byly cílem této práce.

Vyhodnocení trénování modelů neuronových sítí

Poslední zastávkou je ohlédnutí za trénováním modelů neuronových sítí. Metodika učení vybraného modelu pro experimenty (*Place-back-rewrite*) je snaha pochopit jazyk promotorů. Z výsledků experimentů se jeví, že způsobené zásahy nijak významně nenarušily funkčnost promotoru. Výsledek by tak odpovídal stylu učení, kdy se model neuronové sítě snaží umístit předloženou sekvenci na místo, kde mu to přijde na základě okolí přirozené. Teoreticky by ji tak mohl i vylepšit.

V podsekci 4.1.1 byly ukázány i výsledky pro modely, které pro experimenty využity nebyly. Jedná se například o *Place-back-rewrite* se skip-connections, který vracel po jednom pokusu natrénování zajímavé výstupy. S rozšířením modelu by tak mohlo dojít k odlišným návrhům pro umístění regulačních sekvencí do promotorů.

Dále byl v práci zmíněn přístup nazvaný jako *Insert-fragment* (podsekce 3.1.3). Zde se nedostalo ani teoreticky k žádným výsledkům. Metoda se však snaží o rozdílný přístup vkládání operátoru do promotoru. Místo přepisu části promotoru by mělo dojít pouze k jeho vložení. Bylo by poté zajímavé srovnání, jak se výsledky této metody liší od metody *Place-back-rewrite*.

Kapitola 5

Závěr

Diplomová práce měla za cíl pomocí strojového učení vytvořit metodu vkládání regulačních míst do promotorů. Pro tyto účely byly navrženy tři rozdílné metody (*Classifier*, *Place-back*, *Insert-piece*), které se dále větvily na další přístupy. Z metody *Place-back* byl na základě testovacích výstupů vybrán přístup *rewrite*, pro nějž byl sestaven reálný experiment s dosaženou mírou represe až 88.3 %.

První část práce seznámila čtenáře s informacemi potřebnými pro pochopení řešeného problému. Nachází se zde shrnutí biologických pojmů, které se v práci využívají. Dále je zde sekce seznamující s genetickou regulací. Popsané jsou motivy a logické brány nacházející se přírodně v genomu organismů a expertně vytvořené regulační obvody. Následně se přechází k expertnímu návrhu změn v promotorech spolu s nastíněním, proč by bylo vhodné vytvořit nástroj navrhující změny automaticky. V pořadí je pak sekce s popisem potřebných modulů k natrénování zde probíraných neuronových sítí.

V následující kapitole byly uvedeny všechny zde vytvořené metody a datasety. Nejprve jsou zde zformulovány různé přístupy pro řešení úlohy pomocí neuronových sítí. Dále je seznámení s použitými datasety pro trénování neuronových sítí a nakonec detailní popis struktury jednotlivých natrénovaných modelů.

V poslední kapitole byly uvedeny získané výsledky. Nejprve pro modely neuronových sítí, na základě kterých byl sestaven experiment provedený v laboratoři. Následně jsou uvedeny postupy a metody pro vykonání experimentu, na které navazují získané výsledky. Z těch vyplývá, že navržené úpravy byly v daném provedení experimentů úspěšné skoro v jedné třetině případů. Kapitola je zakončena diskuzí nad naměřenými výsledky. Analyzují se zde postupy v souvislosti se získanými daty.

Tato diplomová práce byla mezioborová. Zahrnovala jak oblast umělé inteligence, tak oblast biokybernetiky a biochemie. Právě spojení zmíněných oborů začíná být zajímavým tématem. Dostáváme se do doby, kdy bude jednodušší získat množství velmi specifických dat, které do teď nebyly dostupné. Expertní znalosti v oblasti biologie a biochemie jsou rozvíjeny již velice dlouho, mají však jistá omezení přes která se člověk sám nedostane. Bylo by tak opravdu zajímavé se této možnosti chytit a být u zrodu nových poznatků prospěšných pro celé lidstvo.

Literatura

[1] J. D. Watson and F. H. C. Crick, "Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid," *Nature*, vol. 171, no. 4356, pp. 737–738, apr 1953.

[2] J. L. Alberts, *Molecular Biology of the Cell*, 2002.

[3] H. Norman. (2014) 3-prime and 5-prime structure. [Online]. Available: https://www.quora.com/What-do-3-and-5-in-a-DNA-structure-mean

[4] Purcell. (2017) Dna. [Online]. Available: https://basicbiology.net/micro/genetics/dna

[5] R. Hengge-Aronis, "Stationary phase gene regulation: what makes an escherichia coli promoter sigmas-selective?" *Current Opinion in Microbiology*, vol. 5, no. 6, pp. 591–595, dec 2002.

[6] G. Norris. (2012) Molecular biology of the gene. [Online]. Available: https://slideplayer.com/slide/7526755/

[7] C. Austin. (2021) Open reading frame. [Online]. Available: https://www.genome.gov/genetics-glossary/Open-Reading-Frame

[8] G. Karp, *Cell and molecular biology: concepts and experiments*, 2009.

[9] R. Nave. (2016) Translation or protein synthesis. [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/Organic/translation.html

[10] J. W. Samantha Fowler, Rebecca Roush, *Concepts of Biology*, 2013.

[11] J. D. Watson, *Molecular biology of the gene*, 2014.

[12] T. I. Lee and R. A. Young, "Transcription of eukaryotic protein-coding genes," *Annual Review of Genetics*, vol. 34, no. 1, pp. 77–137, dec 2000.

[13] S. M. Carr. (2021) Regulatory proteins. [Online]. Available: https://www.mun.ca/biology/scarr/bio4241_regulatoryproteins.htm

[14] M. E. Lee, W. C. DeLoache, B. Cervantes, and J. E. Dueber, "A highly characterized yeast toolkit for modular, multipart assembly," *ACS Synthetic Biology*, vol. 4, no. 9, pp. 975–986, may 2015.

[15] GoldBio. (2021) Pcr overview. [Online]. Available: https://www.goldbio.com/goldbios-pcr-overview

[16] E. Bianconi, A. Piovesan, F. Facchin, A. Beraudi, R. Casadei, F. Frabetti, L. Vitale, M. C. Pelleri, S. Tassani, F. Piva, S. Perez-Amodio, P. Strippoli, and S. Canaider, "An estimation of the number of cells in the human body," *Annals of Human Biology*, vol. 40, no. 6, pp. 463–471, jul 2013.

[17] T. H. Saey. (2018) A recount of human genes ups the number to at least 46,831. [Online]. Available: https://www.sciencenews.org/article/recount-human-genes-ups-number-least-46831

[18] D. B. L. Nguyen, Phan, "Effects of carriage and expression of the tn10 tetracycline-resistance operon on the fitness of escherichia coli k12." *Molecular Biology and Evolution*, may 1989.

[19] U. Alon, *An introduction to systems biology: design principles of biological circuits*, 2006.

[20] J.-B. Lugagne, S. S. Carrillo, M. Kirch, A. Köhler, G. Batt, and P. Hersen, "Balancing a genetic toggle switch by real-time feedback control and periodic forcing," *Nature Communications*, vol. 8, no. 1, nov 2017.

[21] J. L. Ramos, M. Martínez-Bueno, A. J. Molina-Henares, W. T. nad Kazuya Watanabe nad Xiaodong Zhang, M. T. Gallegos, R. Brennan, and R. Tobes, "The tetr family of transcriptional repressors," *Microbiology and Molecular Biology Reviews*, 2005.

[22] L. Glass and S. A. Kauffman, "The logical analysis of continuous, non-linear biochemical control networks," *Journal of Theoretical Biology*, vol. 39, no. 1, pp. 103–129, apr 1973.

[23] T. R. Thieffry, D., "Qualitative analys is of gene networks," *Biocomputing '98 - Proceedings Of The Pacific Symposium*, 1998.

[24] C. Yuh, "Genomic cis-regulatory logic: Experimental and computational analysis of a sea urchin gene," *Science*, vol. 279, no. 5358, pp. 1896–1902, mar 1998.

[25] Y. Pilpel, P. Sudarsanam, and G. M. Church, "Identifying regulatory networks by combinatorial analysis of promoter elements," *Nature Genetics*, vol. 29, no. 2, pp. 153–159, sep 2001.

[26] N. E. Buchler, U. Gerland, and T. Hwa, "On schemes of combinatorial transcription logic," *Proceedings of the National Academy of Sciences*, vol. 100, no. 9, pp. 5136–5141, apr 2003.

[27] A. E. Mayo, Y. Setty, S. Shavit, A. Zaslaver, and U. Alon, "Plasticity of the cis-regulatory input function of a gene," *PLoS biology*, 2006.

[28] R. Milo, "Network motifs: Simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, oct 2002.

[29] S. S. Shen-Orr, R. Milo, S. Mangan, U. Alon, S. S. Shen-Orr, R. Milo, and S. Mangan, "Network motifs in the transcriptional regulation network of escherichia coli," *Nature Genetics*, 2002.

[30] P. Erdos and A. Renyi, "Some further statistical properties of the digits in cantors series," *Acta Mathematica Academiae Scientiarum Hungaricae*, vol. 10, no. 1-2, pp. 21–29, mar 1959.

[31] T. A. Carrier and J. Keasling, "Investigating autocatalytic gene expression systems through mechanistic modeling," *Journal of Theoretical Biology*, vol. 201, no. 1, pp. 25–36, nov 1999.

[32] J. Demongeot, M. Kaufman, and R. Thomas, "Positive feedback circuits and memory," *Comptes Rendus de lAcademie des Sciences - Series III - Sciences de la Vie*, vol. 323, no. 1, pp. 69–79, jan 2000.

[33] T. S. Gardner, C. R. Cantor, and J. J. Collins, "Construction of a genetic toggle switch in escherichia coli," *Nature*, vol. 403, no. 6767, pp. 339–342, jan 2000.

[34] J. Monod and F. Jacob, "General conclusions: Teleonomic mechanisms in cellular metabolism, growth, and differentiation," *Cold Spring Harbor Symposia on Quantitative Biology*, vol. 26, no. 0, pp. 389–401, jan 1961.

[35] R. A. Weisberg, "A genetic switch: Phage lambda and higher organisms.mark ptashne," *The Quarterly Review of Biology*, vol. 69, no. 2, pp. 267–268, jun 1994.

[36] M. Ishiura, "Expression of a gene cluster kaiABC as a circadian feedback process in cyanobacteria," *Science*, vol. 281, no. 5382, pp. 1519–1523, sep 1998.

[37] A. Tiwari, J. C. J. Ray, J. Narula, and O. A. Igoshin, "Bistable responses in bacterial genetic networks: Designs and dynamical consequences," *Mathematical Biosciences*, vol. 231, no. 1, pp. 76–89, may 2011.

[38] A. Milias-Argeitis, S. Summers, J. Stewart-Ornstein, I. Zuleta, D. Pincus, H. El-Samad, M. Khammash, and J. Lygeros, "In silico feedback for in vivo regulation of a gene expression circuit," *Nature Biotechnology*, vol. 29, no. 12, pp. 1114–1116, nov 2011.

[39] C. Briat, A. Gupta, and M. Khammash, "Antithetic integral feedback ensures robust perfect adaptation in noisy biomolecular networks," *Cell Systems*, vol. 2, no. 1, pp. 15–26, jan 2016.

[40] P. L. Kapitza, "Dynamic stability of a pendulum with an oscillating point of suspension." *J. Exp. Theor. Phys*, 1951.

[41] S. Iyer, D. K. Karig, S. E. Norred, M. L. Simpson, and M. J. Doktycz, "Multi-input regulation and logic with t7 promoters in cells and cell-free systems," *PLoS ONE*, vol. 8, no. 10, p. e78442, oct 2013.

[42] M. Rong, B. He, W. T. McAllister, and R. K. Durbin, "Promoter specificity determinants of t7 RNA polymerase," *Proceedings of the National Academy of Sciences*, vol. 95, no. 2, pp. 515–519, jan 1998.

[43] J. W. Dubendorf and F. Studier, "Controlling basal expression in an inducible t7 expression system by blocking the target t7 promoter with lac repressor," *Journal of Molecular Biology*, vol. 219, no. 1, pp. 45–59, may 1991.

[44] S. Oehler, M. Amouyal, P. Kolkhof, B. von Wilcken-Bergmann, and B. Müller-Hill, "Quality and position of the three lac operators of e. coli define efficiency of repression." *EMBO J.*, 1994.

[45] Y. Chen, S. Zhang, E. M. Young, T. S. Jones, D. Densmore, and C. A. Voigt, "Genetic circuit design automation for yeast," *Nature Microbiology*, vol. 5, no. 11, pp. 1349–1360, aug 2020.

[46] J. Brosius and A. Holy, "Regulation of ribosomal RNA promoters with a synthetic lac operator." *Proceedings of the National Academy of Sciences*, vol. 81, no. 22, pp. 6929–6933, nov 1984.

[47] J. Monod, "Recherches sur la croissance des cultures bacteriennes," *Agris*, 1942.

[48] K. E. Mayo, R. Warren, and R. D. Palmiter, "The mouse metallothionein-i gene is transcriptionally regulated by cadmium following transfection into human or mouse cells," *Cell*, vol. 29, no. 1, pp. 99–108, may 1982.

[49] L. Nover, *Heat shock response*, 1991.

[50] N. E. Hynes, U. Rahmsdorf, N. Kennedy, L. Fabiani, R. Michalides, R. Nusse, and B. Groner, "Structure, stability, methylation, expression and glucocorticoid induction of endogenous and transfected proviral genes of mouse mammary tumor virus in mouse fibroblasts," *Gene*, vol. 15, no. 4, pp. 307–317, dec 1981.

[51] M. Hu, "The inducible iac operator-repressor system is functional in mammalian cells," *Cell*, vol. 48, no. 4, pp. 555–566, feb 1987.

[52] J. Nakabayashi, "Optimal gene expression for efficient replication of herpes simplex virus type 1 (HSV-1)," in *Herpesviridae - A Look Into This Unique Family of Viruses*. InTech, mar 2012.

[53] T. R. Fuerst, M. P. Fernandez, and B. Moss, "Transfer of the inducible lac repressor/operator system from escherichia coli to a vaccinia virus expression vector." *Proceedings of the National Academy of Sciences*, vol. 86, no. 8, pp. 2549–2553, apr 1989.

[54] U. Deuschle, R. Hipskind, and H. Bujard, "RNA polymerase II transcription blocked by escherichia coli lac repressor," *Science*, vol. 248, no. 4954, pp. 480–483, apr 1990.

[55] L. Han, H. G. Garcia, S. Blumberg, K. B. Towles, J. F. Beausang, P. C. Nelson, and R. Phillips, "Concentration and length dependence of DNA looping in transcriptional regulation," *PLoS ONE*, vol. 4, no. 5, p. e5621, may 2009.

[56] J. Müller, S. Oehler, and B. Müller-Hill, "Repression oflacPromoter as a function of distance, phase and quality of an AuxiliarylacOperator," *Journal of Molecular Biology*, vol. 257, no. 1, pp. 21–29, mar 1996.

[57] M. Razo-Mejia, S. L. Barnes, N. M. Belliveau, G. Chure, T. Einav, M. Lewis, and R. Phillips, "Tuning transcriptional regulation through signaling: A predictive theory of allosteric induction," *Cell Systems*, vol. 6, no. 4, pp. 456–469.e10, apr 2018.

[58] M. Lewis, "The lac repressor," *Comptes Rendus Biologies*, vol. 328, no. 6, pp. 521–548, jun 2005.

[59] E. Volna, *NEURONOVÉ SÍTĚ 1*, 2008.

[60] S. Lawrence, C. Giles, A. C. Tsoi, and A. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, jan 1997.

[61] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, nov 1997.

[62] V. Myska, "Rekurentní neuronové sítě pro klasifikaci textů," Master's thesis, 2018.

[63] Google. (2021) Sentencepiece. [Online]. Available: https://github.com/google/sentencepiece

[64] J. Wong. (2020) Understanding sentencepiece. [Online]. Available: https://jacky2wong.medium.com/understanding-sentencepiece-under-standing-sentence-piece-ac8da59f6b08

[65] Benchling. (2021) Benchling - biology software. [Online]. Available: https://benchling.com

[66] T. Knight, "Idempotent vector design for standard assembly of biobricks," *MIT Synthetic Biology Working Group Technical Reports 2003.*, 2003.

[67] C. Engler, R. Kandzia, and S. Marillonnet, "A one pot, one step, precision cloning method with high throughput capability," *PLoS ONE*, vol. 3, no. 11, p. e3647, nov 2008.

[68] R. P. Shetty, D. Endy, and T. F. Knight, "Engineering BioBrick vectors from BioBrick parts," *Journal of Biological Engineering*, vol. 2, no. 1, apr 2008.

[69] S. University. (2021) Sgd. [Online]. Available: https://www.yeastgenome.org/

[70] P. T. Monteiro, J. Oliveira, P. Pais, M. Antunes, M. Palma, M. Cavalheiro, M. Galocha, C. P. Godinho, L. C. Martins, N. Bourbon, M. N. Mota, R. A. Ribeiro, R. Viana, I. Sa-Correia, and M. C. Teixeira, "Yeastract:: a portal for cross-species comparative genomics of transcription regulation in yeasts," *Nucleic Acids Research*, vol. 48, no. D1, pp. D642–D649, oct 2019.

[71] S. L. University. (2021) Yeastss. [Online]. Available: http://www.yeastss.org/

[72] R. Dreos, G. Ambrosini, R. Groux, R. C. Périer, and P. Bucher, "The eukaryotic promoter database in its 30th year: focus on non-vertebrate organisms," *Nucleic Acids Research*, vol. 45, no. D1, pp. D51–D55, nov 2016.

[73] E. W. Sayers, R. Agarwala, E. E. Bolton, J. R. Brister, K. Canese, K. Clark, R. Connor, N. Fiorini, K. Funk, T. Hefferon, J. B. Holmes, S. Kim, A. Kimchi, P. A. Kitts, S. Lathrop, Z. Lu, T. L. Madden, A. Marchler-Bauer, L. Phan, V. A. Schneider, C. L. Schoch, K. D. Pruitt, and J. Ostell, "Database resources of the national center for biotechnology information," *Nucleic Acids Research*, vol. 47, no. D1, pp. D23–D28, nov 2018.

[74] S. Sahoo. (2018) Residual blocks — building blocks of resnet. [Online]. Available: https://towardsdatascience.com/residual-blocks-building-blocks-of-resnet-fd90ca15d6ec

[75] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016.

[76] K. Shannon and I. Phillips, "beta-lactamase detection by three simple methods: Intralactam, nitrocefin and acidimetric," *Journal of Antimicrobial Chemotherapy*, vol. 6, no. 5, pp. 617–621, 1980.

[77] C. Chow, H. Xu, and J. S. Blanchard, "Kinetic characterization of hydrolysis of nitrocefin, cefoxitin, and meropenem by beta-lactamase from mycobacterium tuberculosis," *Biochemistry*, vol. 52, no. 23, pp. 4097–4104, may 2013.

[78] W. Gilbert and B. Muller-Hill, "ISOLATION OF THE LAC REPRESSOR," *Proceedings of the National Academy of Sciences*, vol. 56, no. 6, pp. 1891–1898, dec 1966.

[79] Y. Setty, A. E. Mayo, M. G. Surette, and U. Alon, "Detailed map of a cis-regulatory input function," *Proceedings of the National Academy of Sciences*, vol. 100, no. 13, pp. 7702–7707, jun 2003.

[80] T. Kuhlman, Z. Zhang, M. H. Saier, and T. Hwa, "Combinatorial transcriptional control of the lactose operon of escherichia coli," *Proceedings of the National Academy of Sciences*, vol. 104, no. 14, pp. 6043–6048, mar 2007.

[81] B. Otová, *Lékařská biologie a genetika : 1. díl*, 2019.

[82] E. Dekel and U. Alon, "Optimality and evolutionary tuning of the expression level of a protein," *Nature*, vol. 436, no. 7050, pp. 588–592, jul 2005.

[83] M. Eames and T. Kortemme, "Cost-benefit tradeoffs in engineered lac operons," *Science*, vol. 336, no. 6083, pp. 911–915, may 2012.

[84] W. S. Reznikoff, "The lactose operon-controlling elements: a complex paradigm," *Molecular Microbiology*, vol. 6, no. 17, pp. 2419–2422, oct 2006.

[85] M. Razo-Mejia, J. Q. Boedicker, D. Jones, A. DeLuna, J. B. Kinney, and R. Phillips, "Comparison of the theoretical and real-world evolutionary potential of a genetic circuit," *Physical Biology*, vol. 11, no. 2, p. 026005, apr 2014.

[86] F. Jacob and J. Monod, "On the regulation of gene activity," *Cold Spring Harbor Symposia on Quantitative Biology*, vol. 26, no. 0, pp. 193–211, jan 1961.

[87] P. J. Farabaugh, "Sequence of the lacI gene," *Nature*, vol. 274, no. 5673, pp. 765–767, aug 1978.

[88] D. Goodsell, "lac repressor," *RCSB Protein Data Bank*, mar 2003.

[89] W. Gilbert and A. Maxam, "The nucleotide sequence of the lac operator," *Proceedings of the National Academy of Sciences*, vol. 70, no. 12, pp. 3581–3584, dec 1973.

[90] C. P. Bahl, R. Wu, J. Stawinsky, and S. A. Narang, "Minimal length of the lactose operator sequence for the specific recognition by the lactose repressor." *Proceedings of the National Academy of Sciences*, vol. 74, no. 3, pp. 966–970, mar 1977.

[91] J. R. Sadler, H. Sasmor, and J. L. Betz, "A perfectly symmetric lac operator binds the lac repressor very tightly." *Proceedings of the National Academy of Sciences*, vol. 80, no. 22, pp. 6785–6789, nov 1983.

[92] S. Bourgeois and A. D. Riggs, "The lac repressor-operator interaction IV. assay and purification of operator DNA," *Biochemical and Biophysical Research Communications*, vol. 38, no. 2, pp. 348–354, jan 1970.

[93] A. Jobe and S. Bourgeois, "Lac represser-operator interaction," *Journal of Molecular Biology*, vol. 75, no. 2, pp. 303–313, apr 1973.

[94] W. Saenger, P. Orth, C. Kisker, W. Hillen, and W. Hinrichs, "The tetracycline repressor—a paradigm for a biological switch," *Angewandte Chemie International Edition*, vol. 39, no. 12, pp. 2042–2052, jun 2000.

[95] W. Hinrichs, C. Kisker, M. Duvel, A. Muller, K. Tovar, W. Hillen, and W. Saenger, "Structure of the tet repressor-tetracycline complex and regulation of antibiotic resistance," *Science*, vol. 264, no. 5157, pp. 418–420, apr 1994.

[96] P. Orth, W. Saenger, and W. Hinrichs, "Tetracycline-chelated mg2 on initiates helix unwinding in tet repressor induction†,‡," *Biochemistry*, vol. 38, no. 1, pp. 191–198, jan 1999.

[97] J. L. Huffman and R. G. Brennan, "Prokaryotic transcription regulators: more than just the helix-turn-helix motif," *Current Opinion in Structural Biology*, vol. 12, no. 1, pp. 98–106, feb 2002.

[98] M. Gossen, S. Freundlieb, G. Bender, G. Muller, W. Hillen, and H. Bujard, "Transcriptional activation by tetracyclines in mammalian cells," *Science*, vol. 268, no. 5218, pp. 1766–1769, jun 1995.

[99] C. Sizemore, A. Wissmann, U. Gülland, and wolfgang Hillen, "Quantitative analysis of tn10tet repressor binding to complete set oftetoperator mutants," *Nucleic Acids Research*, vol. 18, no. 10, pp. 2875–2880, 1990.

[100] A. Roguev, J. Xu, and N. J. Krogan, "Transformation of schizosaccharomyces pombe in a 96-well format," *Cold Spring Harbor Protocols*, vol. 2018, no. 1, p. pdb.prot091942, jul 2017.

[101] C. B. Brachmann, A. Davies, G. J. Cost, E. Caputo, J. Li, P. Hieter, and J. D. Boeke, "Designer deletion strains derived fromSaccharomyces cerevisiae s288c: A useful set of strains and plasmids for PCR-mediated gene disruption and other applications," *Yeast*, vol. 14, no. 2, pp. 115–132, jan 1998.

[102] D. Georgiev, M. Cienciala, H. Kasl, L. Berne, and T. Puchrova, "Biological computing systems and methods for multivariate surface analysis and object detection," Patent WO2 020 051 534A2, 2020.

Příloha: Materiály a metody

Kvasinkový kmen

Použitý kvasinkový kmen (S0[1]) pochází z rodičovského kmenu BY4741, který je *MAT**a*** a má odstraněné selekční markerové geny *his3*, *leu2*, *met15* a *ura3*. S0 má dále knockoutované ještě další geny: *bar1*, *mfa*, *mfx* a *aga2*. Genotypový zápis se nachází v tabulce 5.1.

Bakteriální vektor

Pro domestikaci[2] a namnožení vytvořených fragmentů byl využit bakteriální vektor založený na bakteriálním plazmidu *ColE1*, který obsahuje *E.coli* marker *CamR* a *BsmBI* restrikčními místy ohraničený gen pro zelený fluorescenční protein (v originálu green fluorescent protein, dále ***GFP***). Bakteriální vektor byl součástí toolkitu z článku (Lee et al. [14]) zakoupeného na adrese www.addgene.org.

Klonovací kazety

Klonovací kazety využité v této práci byly vytvořeny společností XENO Cell Inovations s.r.o. Byly použity tři typy kazet: C001, C002, C003.

Kazeta C001 byla sestavena pro využití assembly metody MoClo a jejím obsahem je *leu2* marker, *leu2* 3‘ homologie, *NotI* restrikčními místy ohraničený gen pro rezistenci na ampicilin a *ColE1*, *leu2* 5‘ homologie a *BsaI* restrikčními místy ohraničený gen pro *GFP*.

C002 je kazeta vytvořená pro vlastní metodu assembly společnosti XENO Cell Inovations s.r.o. Obsahem kazety je *leu2* marker, *leu2* 3‘ homologie, *NotI* restrikčními místy ohraničený gen pro rezistenci na ampicilin a *ColE1*, *leu2* 5‘ homologie, *BsaI* restrikčními místy ohraničený gen pro zelený fluorescenční protein, terminátor *tSSA1* a gen pro signalizační protein *Venus*.

Kazeta C003 je rovněž vytvořena pro vlastní metodu assembly společnosti XENO Cell Inovations s.r.o. a obsahuje *his3* marker, *HO* 3‘ homologii, *NotI* restrikčními místy ohraničený gen pro rezistenci na ampicilin a *ColE1*, *HO* 5‘ homologii, *BsaI* restrikčními místy ohraničený gen pro *GFP*, *tENO2* terminátor a gen pro signalizační protein *Tuquoise*.

[1] Vytvořené kmeny budou začínat písmenem ’S’. Vychází to z anglického překladu slova ’kmen’ → ’strain’

[2] Domestikace je vložení fragmentu DNA do part-plazmidu.

Part-plazmidy

Part-plazmid se obecně skládá z fragmentu DNA vloženého do výše popsaného bakteriálního vektoru místo genu pro *GFP* a je následně využíván pro BsaI assembly k vytvoření genu. Part-plazmidy vznikly pomocí *BsmBI* Golden Gate assembly dle popsaného protokolu [67]. $10\mu l$ reakce obsahuje $6.5\mu l$ Nuclease-free H_2O, $0.5\mu l$ BsmBI-v2 restrikční enzym s koncentrací 10000 $units/ml$, $1\mu l$ T4 buffer 10x, $0.5\mu l$ T7 ligázy s koncentrací 3000000 $units/ml$ (vše zakoupeno u New England BioLabs Inc.), $1\mu l$ vektoru s hmotnostní koncentrací $100ng/\mu l$ a $0.5\mu l$ vkládaného fragmentu z PCR reakce. Teplotní program je poté nastaven na 30 cyklů se střídáním 8 minut: 50°C, 8 minut: 16°C, s následnými 20-ti minutami na 50°C a 20-ti minutami na 80°C.

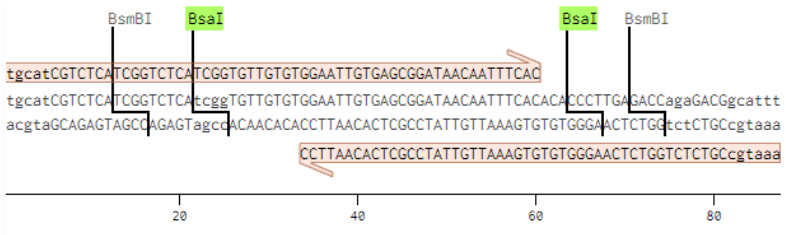
V práci byla využita řada již připravených part-plazmidů. První skupinou jsou plazmidy získané v rámci MoClo toolkitu a jedná se o promotory *pHHF2*, *pPAB1*, *pPOP6* a *pREV1*. Druhou skupinou jsou part-plazmidy, kde vložený fragment byl syntetizován společností Twist Bioscience. Takto sestavené part-plazmidy byly pro proteiny *BLA*, *LacI* a *TetR*. Posledními jsou part-plazmidy, kde byl fragment amplifikován z genomu *Saccharomyces c.* a takto získán byl promotor *pX* a *FLO11-PRE* sekvence, což je signální peptid, díky kterému se proteinu dostane ven z buňky.

Primery

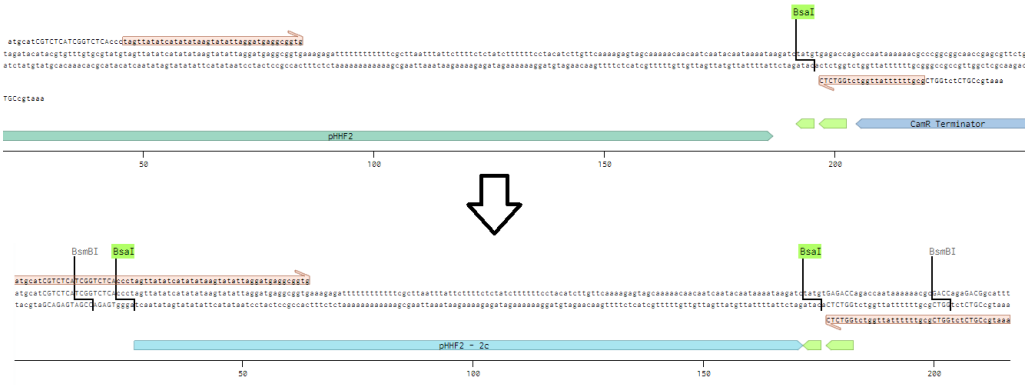
Primery použité v diplomové práci byly objednány od Integrated DNA Technologies. Primery byly navrženy na amplifikaci částí již domestikovaných promotorů a pro vytvoření operátorů. Oba procesy se prováděly pomocí metody PCR. Pro navázání primeru na RNA je potřeba, aby měly společných alespoň 15 bází, které se při správném návrhu a v odpovídajícím stavu prostředí na RNA nalepí. Ve volných koncích primerů pak mohou být sekvence obsahující informace pro restrikční enzymy nebo se jimi můžou vytvářet genetické modifikace.

Pro amplifikaci byly primery navrženy tak, aby z jedné strany nasedaly na Part-plazmid (který je společný pro všechny domestikované fragmenty) a vytvářely v dané oblasti *BsmBI* restrikční místo. Protichůdný primer se přichytával na oblast v promotoru a na svém konci vytvářel *BsaI* a *BsmBI* restrikční místo (viz obr. 5.2).

Druhým typem byly primery, které svým spojením vytvářeli operátory. Při dostatečném překrytí primerů (přes 15 bází) se během PCR doplní zbytek primerů a vzniknou fragmenty DNA (viz obr. 5.1).



Obrázek 5.1: Vznik fragmentů odpovídající použitým operátorům.



Obrázek 5.2: Schéma přípravy fragmentů pro následné sestavení part-plazmidů. Shodné oblasti primeru s DNA jsou barevně zvýrazněny, volné konce jsou bezbarvé.

Příprava part-plazmidů

Před sestavováním genů do kazet si bylo třeba připravit zbylé části, které nebyly předem dostupné. V této fázi si bylo zapotřebí vytvořit dva fragmenty z každého promotoru a fragmenty odpovídající vkládaným operátorům (obr.5.2, 5.1) a následně je vložit do bakteriálního vektoru (viz obr. 5.3).

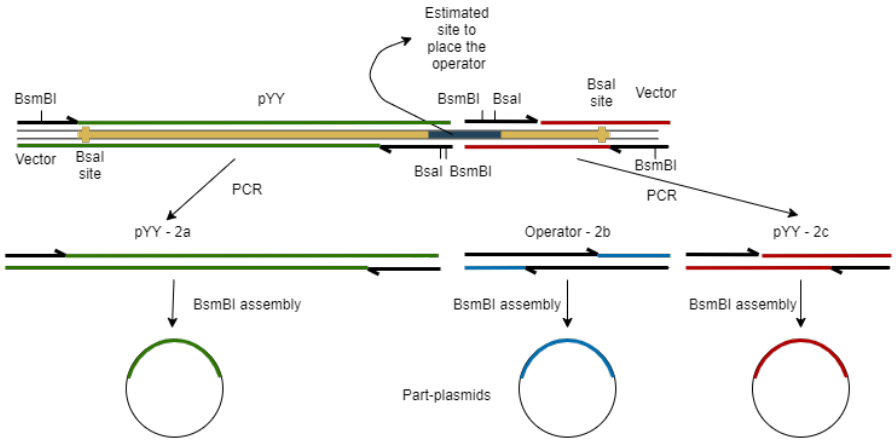
Pro získání částí promotorů bylo využito standardního PCR. 25μl reakce se skládala z 15.7μl Nuclease-free H_2O, 5μl Q5 buffer 5x, 0.5μl dNTPs 10mM, 0.3μl Q5 polymeráza s koncentrací 200$units/\mu l$ (vše zakoupeno u New England BioLabs Inc.), 1.25μl od obou primerů s koncentrací 5$ng/\mu l$ a 1μl vzorového DNA s koncentrací 100$ng/\mu l$.

Při přípravě operátorů byl poměr materiálů pozměněn. Reakce byla na 40μl s obsahem 26.5μl Nuclease-free H_2O, 8μl Q5 buffer 5x, 1μl dNTPs 10mM, 0.5μl Q5 polymeráza s koncentrací 200$units/\mu l$ a 2μl od obou primerů s tentokrát nezředěnou koncentrací 100$ng/\mu l$.

Teplotní program byl poté pro oba typy stejný: 2 minuty 98°C, poté následoval třicetkrát cyklus 30 sekund na 98°C, 30 sekund 56°C a 1 minuta 72°C, po doběhnutí cyklu následovaly poslední 2 minuty na 72°C.

Procesem PCR byly tedy získány fragmenty, které se vkládaly do bakteriálního vektoru pomocí již popsané *BsmBI* assembly a tím byly získány nové part-plazmidy. Fragmenty a následně z nich vzniklé part-plazmidy jsou označené jako y - 2x, kdy y je nahrazeno názvem operátoru nebo promotoru, ze kterého byl fragment získán a 2x označuje pozici v assembly. Část MoClo assembly s označením 2 odpovídá promotoru. 2a je tak první část promotoru, 2b je operátor a 2c je konec promotoru. Výčet nově vytvořených part-plazmidů je následující: *pX-2a*, *pX-2c*, *pHHF2-2a*, *pHHF2-2c*, *pPAB1-2a*, *pPAB1-2c*, *pPOP6-2a*, *pPOP6-2c*, *pREV1-2a*, *pREV1-2c*, *Olac-2b* a *tet0-2b*.

Part-plazmidy byly po assembly vloženy pomocí chemické transformace do bakterií. S 10μl bakterií CC Turbo (z New England BioLabs Inc.) byly smíchány 4μl assembly part-plazmidů a nastavil se teplotní program: 30 minut 4°C, 15 sekund



Obrázek 5.3: Schéma vytvoření potřebných part-plazmidů. Přidružené označení 2x odpovídá umístění v MoClo assembly, kde je promotor částí číslo 2.

50°C, 2 minuty 4°C. Poté bylo přidáno $100\mu l$ média SOC následované hodinovou inkubací v 37°C na otáčky 1200rpm. Výsledný produkt se nanesl na chloramphenicolové misky, vůči kterým mají rezistenci pouze bakterie, které v sobě mají zaintegrovaný vkládaný part-plazmid. Po jednom dni v teplotě 37°C má mít miska podobu jako na obrázku 2.33. Z vybrané kolonie se potřela půlka nové misky, ze které se poté pomocí Miniprepu (toolkit a protokol od Fisher Scientific UK) vypreparovaly namnožené part-plazmidy, jež se nakonec naředily na $100ng/\mu l$. Vytvořené part-plazmidy byly následně poslány na ověření sekvenací[3] do Eurofins Genomics.

Assembly genů

Celkem bylo pro diplomovou práci složeno 22 genů. Dva geny se vkládaly do C001, dva do C002 a osmnáct do C003. Do kazety C001 bylo třeba vložit promotor, kódující sekvenci a terminátor, zatímco do kazet C002 a C003 pouze promotor a kódující sekvenci. Assembly genů probíhala pomocí *BsaI* restrikcí a v těchto kazetách se za restrikčním místem nachází již připravený terminátor, který tak není třeba přidávat. Ukázka procesu sestavení genu je na obrázku 5.4.

C001 kazeta sloužila pro vložení genu produkujícího regulační protein *LacI*. Části assembly byly kromě C001 promotor pPOP6/pCCW12, kódující sekvence LacI-nls a terminátor tENO1/tENO2. Vzniklé geny byly pPOP6-LacI-tENO1 a pCCW12-LacI-tENO2.

Do kazety C002 již nebylo třeba vkládat terminátor (již přítomen *tSSA1*) a vzniklé geny opět produkovaly regulační protein, tentokrát *TetR*. Do kazety se tak vkládaly pouze promotor a kódující sekvence. Výslednými geny v této kazetě byly pPOP6-TetR-tSSA1 a pCCW12-TetR-tSSA1.

Poslední kazeta C003 byla využita na sestavení všech genů se signalizačním proteinem *BLA*. Tato kazeta má také již integrovaný terminátor (tENO2). Schéma

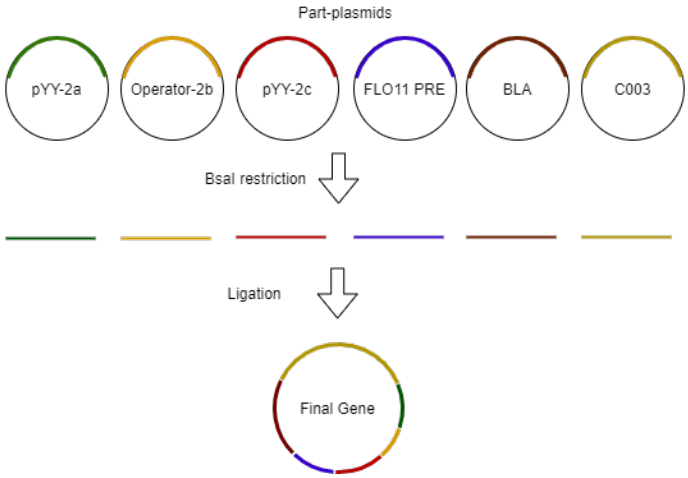
[3]Sekvenace je metoda zjišťující sekvenci nukleových bází.

této assembly se nachází na obrázku 5.4. Promotory byly použity buď přírodní nebo již dříve upravené (ucelená sekvence) nebo nově upravené (první část-2a, operátor-2b, koncová část-2c). Za promotorem se nachází *FLO11-PRE* peptid rozšiřující následující kódující sekvenci pro protein *BLA*, který umožní proteinu *BLA* opustit buňku. Díky tomu je poté možné jednoduše měřit množství *BLA* produkované buňkami. Takto vzniklé geny s původními promotory jsou *pX-BLA-tENO2*, *pHHF2-BLA-tENO2*, *pPAB1-BLA-tENO2*, *pPOP6-BLA-tENO2*, *pREV1-BLA-tENO2*, pX_L-BLA-$tENO2_{lab}$[4], pX_T-BLA-$tENO2_{lab}$, pZ_T-BLA-$tENO2_{lab}$ a pro upravené promotory pX_L-*BLA-tENO2*, pX_T-*BLA-tENO2*, $pHHF2_L$-*BLA-tENO2*, $pHHF2_T$-*BLA-tENO2*, $pPAB1_L$-*BLA-tENO2*, $pPAB1_T$-*BLA-tENO2*, $pPOP6_L$-*BLA-tENO2*, $pPOP6_T$-*BLA-tENO2*, $pREV1_L$-*BLA-tENO2* a $pREV1_T$-*BLA-tENO2*. Dále budou sestavené geny uváděny bez názvů terminátorů a *BLA* (například pZ_T-BLA-$tENO2_{lab}$ →pZ_T, $pHHF2_L$-BLA-tENO2→$pHHF2_L$).

Obsah $10\mu l$ reakce *BsaI* Golden Gate assembly je následující: $1\mu l$ T4 buffer 10x, $0.5\mu l$ restrikční enzym *BsaI-HF-v2* s koncentrací $20000units/ml$, $0.5\mu l$ T4 ligázy $400000units/ml$ (z New England BioLabs Inc.), $0.5\mu l$ každé části assembly s koncentrací $100ng/\mu l$ a zbytek do $10\mu l$ se doplní Nuclease-free H_2O.

Teplotní program začínal 30-ti násobným cyklem 8 minut na 37°C a 8 minut na 16°C, následuje 20 minut na 37°C a 15 minut na 80°C.

Vzniklé plazmidy se stejně jako v případě part-plazmidů transformovali do bakterií CC Turbo. Bakterie se poté nedávaly na chloramphenicolovou misku, ale na ampicilinovou, neboť součástí kazet je gen vracející bakterii rezistenci právě proti ampicilinu. Po výběru kolonie se opět potře půl misky a poté co bakterie narostou se udělá Miniprep a vyextrahované plazmidy se naředí na $100ng/\mu l$. Sestavené geny byly následně poslány na ověření sekvenací do Eurofins Genomics.



Obrázek 5.4: Schéma procesu Golden Gate assembly.

[4] pX_L je zkrácený zápis pro *pX-Olac*, analogicky pro promotor *pX-tet0* bude zkrácený zápis pX_T. X_{lab} znamená, že použitý promotor není přírodní, ale už byl sestaven v laboratoři dříve někým jiným. Promotor *pZ* je jako v případě *pX* duševním vlastnictvím XENO Cell Inovations s.r.o.

Transformace genů do kvasinek

Dle popsané stavby kazet, mezi restrikčními místy *NotI* je *AmpR* gen a *ColE1*. Jedná se tak o oblasti důležité pro namnožení v bakteriích, které dále nejsou třeba. Zvenku *NotI* míst jsou pak konce homologií, díky kterým se při správném nastavení může potřebná část zaintegrovat do genomu kvasinky.

Prvním krokem před transformací je *NotI* restrikce. $10\mu l$ reakce obsahuje $3\mu l$ nuclease-free H_2O, $1\mu l$ *NotI-HF* (z New England BioLabs Inc.) s koncentrací $1unit/\mu l$, $1\mu l$ CutSmart buffer 10x (z New England BioLabs Inc.) a $5\mu l$ plazmidu s koncentrací $100ng/\mu l$. Teplotní program byl nastavený na dvě hodiny v 37°C a následně 20 minut v 65°C.

Následný program pro kvasinkovou transformaci byl převzat z (Roguev et al. [100]). Po transformaci se kvasinky dají na Petriho misku s médiem bez odpovídající aminokyseliny pro vkládaný konstrukt. Pro konstrukty s kazetou C001 a C002 byla použita miska bez *Leu2* a pro C003 bez *His3*. Po dvou dnech inkubace ve 30°C miska poroste koloniemi (viz obr. 2.33), ze kterých se jedna vybere a rozetře se na novou misku se shodným médiem. Z vybraného vzniklého singletu na nové misce se poté udělá overnight. Do tekutého živného média se přidá nabraný singlet z misky a kultura se přes noc nechá růst. Další den je možné pro kontrolu provést genetickou extrakci a pomocí PCR ověřit, zda-li se požadovaná genetická informace dostala do kvasinky. V případě pozitivních testů je možné kulturu pro pozdější využití zamrazit v -80°C. Kvasinkovou transformací byly vytvořeny kmeny kvasinek uvedené v tabulce 5.1 (kromě rodičovských kmenů BY4741 a S0). Jako test úspěšného vložení genu do kvasinky byla v tomto případě extrakce genomu s následným PCR.
Pro lepší orientaci v označeních kmenů je zde popsáno použité rozdělení co skupiny kmenů obsahují za geny:

- S001-S005 - Upravený promotor s operátorem *Olac* před kódující sekvencí pro *BLA*.
- S006-S010 - Upravený promotor s operátorem *tet0* před kódující sekvencí pro *BLA*.
- S100/S200 - Promotor *pCCW12/pPOP6* s kódující sekvencí pro regulační protein *LacI*.
- S300/S400 - Promotor *pCCW12/pPOP6* s kódující sekvencí pro regulační protein *TetR*.
- S501-S505 - Nativní promotor před kódující sekvencí pro *BLA*.
- S1xx/S2xx/S3xx/S4xx - Kmeny s dvěma vloženými geny. Jedním je gen z S100/S200/S300/S400, druhým je gen z S001-S010.
- S601-S605 - Referenční kmeny s expertně navrženými úpravami promotorů.

V práci však jsou kmeny uváděny pro lepší orientaci místo jejich názvů převážně pomocí zkráceně zapsaného vloženého obsahu. Z názvů složených genů jsou vynechány terminátory a *BLA*. Například kmen S309, který obsahuje geny *pCCW12-TetR-tSSA1* a *pPOP6*$_T$*-BLA-tENO2* bude pojmenován jako *pCCW12-TetR-pPOP6*$_T$.

Kmen	Genotyp	Reference
BY4741	*MAT**a** his3Δ1 leu2Δ0 met15Δ0 ura3Δ0*	[101]
S0	BY4741 *bar1Δ mfaΔ mfxΔ aga2Δ*	XENO Cell Inovations s.r.o.
S001	S0 *his3::pX$_L$-BLA*	
S002	S0 *his3::pHHF2$_L$-BLA*	
S003	S0 *his3::pPAB1$_L$-BLA*	
S004	S0 *his3::pPOP6$_L$-BLA*	
S005	S0 *his3::pREV1$_L$-BLA*	
S006	S0 *his3::pX$_T$-BLA*	
S007	S0 *his3::pHHF2$_T$-BLA*	
S008	S0 *his3::pPAB1$_T$-BLA*	
S009	S0 *his3::pPOP6$_T$-BLA*	
S010	S0 *his3::pREV1$_T$-BLA*	
S100	S0 *leu2::pCCW12-LacI*	
S200	S0 *leu2::pPOP6-LacI*	
S300	S0 *leu2::pCCW12-TetR*	
S400	S0 *leu2::pPOP6-TetR*	
S501	S0 *his3::pX-BLA*	
S502	S0 *his3::pHHF2-BLA*	
S503	S0 *his3::pPAB1-BLA*	
S504	S0 *his3::pPOP6-BLA*	
S505	S0 *his3::pREV1-BLA*	
S101	S0 *leu2::pCCW12-LacI* *his3::pX$_L$-BLA*	
S102	S0 *leu2::pCCW12-LacI* *his3::pHHF2$_L$-BLA*	
S103	S0 *leu2::pCCW12-LacI* *his3::pPAB1$_L$-BLA*	
S104	S0 *leu2::pCCW12-LacI* *his3::pPOP6$_L$-BLA*	
S105	S0 *leu2::pCCW12-LacI* *his3::pREV1$_L$-BLA*	
S201	S0 *leu2::pPOP6-LacI* *his3::pX$_L$-BLA*	
S202	S0 *leu2::pPOP6-LacI* *his3::pHHF2$_L$-BLA*	
S203	S0 *leu2::pPOP6-LacI* *his3::pPAB1$_L$-BLA*	
S204	S0 *leu2::pPOP6-LacI* *his3::pPOP6$_L$-BLA*	
S205	S0 *leu2::pPOP6-LacI* *his3::pREV1$_L$-BLA*	
S306	S0 *leu2::pCCW12-TetR* *his3::pX$_T$-BLA*	
S307	S0 *leu2::pCCW12-TetR* *his3::pHHF2$_T$-BLA*	
S308	S0 *leu2::pCCW12-TetR* *his3::pPAB1$_T$-BLA*	
S309	S0 *leu2::pCCW12-TetR* *his3::pPOP6$_T$-BLA*	
S310	S0 *leu2::pCCW12-TetR* *his3::pREV1$_T$-BLA*	
S406	S0 *leu2::pPOP6-TetR* *his3::pX$_T$-BLA*	
S407	S0 *leu2::pPOP6-TetR* *his3::pHHF2$_T$-BLA*	
S408	S0 *leu2::pPOP6-TetR* *his3::pPAB1$_T$-BLA*	
S409	S0 *leu2::pPOP6-TetR* *his3::pPOP6$_T$-BLA*	
S410	S0 *leu2::pPOP6-TetR* *his3::pREV1$_T$-BLA*	
S601	S0 *leu2::pCCW12-LacI* *his3::pX$_L$-BLA$_{lab}$*	
S602	S0 *leu2::pPOP6-LacI* *his3::pX$_L$-BLA$_{lab}$*	
S603	S0 *leu2::pPOP6-TetR* *his3::pX$_T$-BLA$_{lab}$*	
S604	S0 *leu2::pCCW12-TetR* *his3::pZ$_T$-BLA$_{lab}$*	
S605	S0 *leu2::pPOP6-TetR* *his3::pZ$_T$-BLA$_{lab}$*	

Tabulka 5.1: Tabulka genotypů *Saccharomyces c.*

Protokol měření

Měřící protokol byl až na drobné výjimky stejný pro měření všech vytvořených kmenů. Ze zmražených buněk se dal na noc do $1ml$ živného média YPD overnight, který se inkuboval v teplotě 30°C v otočném zařízení. Dopoledne se buňky naředily dle absorbance na $OD600 = 0.1$ opět do $1ml$. Buňky se takto nechaly dál inkubovat ve 30-ti°C po dobu čtyř hodin. Po uplynutí stanovené doby se odebralo $500\mu l$ vzorku do uzavíratelné $1.7ml$ plastové zkumavky a v centrifuze se stočilo na $3000rcf$ po dobu jedné minuty. Následně se odebral supernatant a zkumavka se novým médiem doplnila zpět na $500\mu l$. Poté se měřila optická hustota a došlo k naředění vzorku do nové zkumavky na $OD600 = 0.05$. Objem naředěného vzorku byl tentokrát $100\mu l$. Pro měření kmenů, kde byl upravený promotor, byly vytvořeny dva vzorky, kdy do jednoho byl přidán odpovídající induktor. Takto připravené zkumavky se daly na dvě hodiny inkubovat a poté se stočily po dobu jedné minuty na $3000rcf$. Do měřícího platu se přendalo $85\mu l$ supernatantu, do kterého se zamíchalo $10\mu l$ nitrocefinu $1mM$ s $5\mu l$ destilované H_2O. Po přidání nitrocefinu začne docházet k reakci s *BLA* a měřící plate se tak dává hned do čtecího zařízení. Po dobu dvaceti minut se v minutových intervalech měřila fluorescence na vlnových délkách 486nm a 700nm[5]. Výsledné grafy zobrazené v podsekci 4.2.2, jsou získány z vypočtených hodnot $D4860$ pro jamku X v čase tn na základě přepočtu:

$$X_{D486}(tn) = X_{486}(tn) - X_{700}(tn), \tag{5.1}$$

$$X_{D4860}(tn) = X_{D486}(tn) - X_{D486}(t0), \tag{5.2}$$

kdy hodnoty t odpovídají časovému pořadí měření. Hodnota X_{D4860} v čase tn je tedy rozdílem první vypočtené hodnoty X_{D486} od X_{D486} v čase t_n, čímž graf začíná na hodnotě $X_{D4860}(t0) = 0$. $X_{D486}(tn)$ je rozdílem měření hodnot na vlnových délkách 486 a 700nm. Hodnota X_{486} je hlavní měřená hodnota odpovídající vlnové délce pro měření fluorescence červené barvy a odečtením X_{700} dochází ke korekci emisí při reakci *BLA* s nitrocefinem. Podoba vývoje jednotlivých metrik v čase je vyobrazena v grafu 5.5-A. Vývoj $D4860$ je do určité fáze lineárně rostoucí. Tato oblast je proložena lineární funkcí. Sklon lineární funkce je poté vydělen koeficientem $\Delta_{1U} = 0.03$, který odpovídá kompletní degradaci $1nmol$ nitrocefinu pomocí substrátu jednotek *BLA* [102]. Jedna jednotka *BLA* (1U) je definována jako množství enzymu potřebného pro degradaci $1nmol$ nitrocefinu za jednu hodinu. Sloupcové grafy produkce *BLA* (viz obr. 5.5-B) budou v jednotkách *BLA* obsažených v použitém substrátu:

$$amount\_of\_BLA = \frac{a}{\Delta_{1U}}[U], \tag{5.3}$$

kde a je sklon lineární funkce prokládající lineárně rostoucí hodnoty $D4860$.

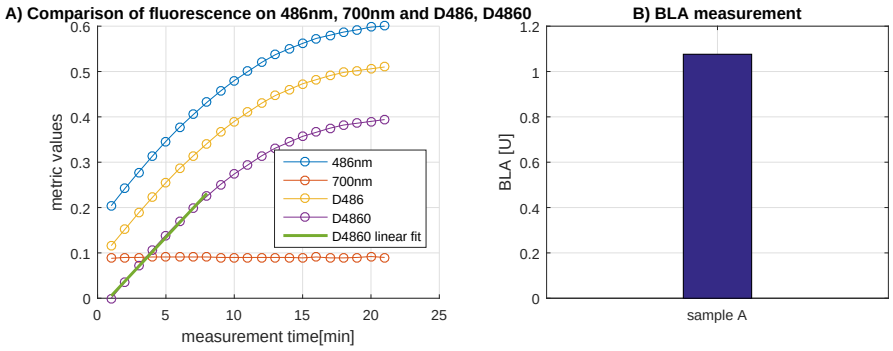
Základní nastavení měření se však ukázalo po prvních pokusech jako nevyhovující. Zatímco kmeny s integrovanými verzemi promotorů *pX/pHHF2* před *BLA* měly vysoký signál, pro *pPAB1* už bylo nastavení pro získání výsledků hraniční

[5]Jednotky měření fluorescence jsou takzvané relativní fluorescenční jednotky.

a pro *pPOP6/pREV1* nebylo naměřeno nic.

Protokol se tak upravoval na míru jednotlivým promotorům. Pro silné promotory *pX/pHHF2* byla pouze snížena cílová optická hustota při posledním ředění na $OD600 = 0.02$. Pro *pPAB1* naopak došlo ke zvýšení hustoty pro poslední ředění na $OD600 = 0.1$. Pro zbylé dva promotory *pPOP6/pREV1* nejprve došlo při posledním ředění ke zvýšení cílové optické hustoty na $OD600 = 0.15$. Později s dalším zvýšením cílové hustoty na $OD600 = 0.2$ došlo i ke zvýšení doby inkubace před měřením na čtyři hodiny. Měření samotné se v tomto případě protáhlo na 30 minut po minutových intervalech měření.

Kmeny obsahující *BLA* gen i gen s transkripčním faktorem byly měřeny vždy ve dvou jamkách. V jedné byl přidaný induktor[6] (v grafech vzorky se znaménkem '+' v názvu) a v druhé nebyl (vzorky se znaménkem '-' v názvu). Tím byly získány rozdíly způsobené represí promotoru předcházejícího sekvenci pro *BLA*. Finální měření s posledně popsaným nastavením protokolu byly následně prováděny simultánně ve dvou replikátech, které se do grafů průměrovaly. Při odlišném finálním naředění vzorků od $OD600 = 0.05$ budou získané hodnoty přenásobeny odpovídajícím koeficientem tak, aby byly v grafech zobrazené hodnoty vztažené k $OD600 = 0.05$.

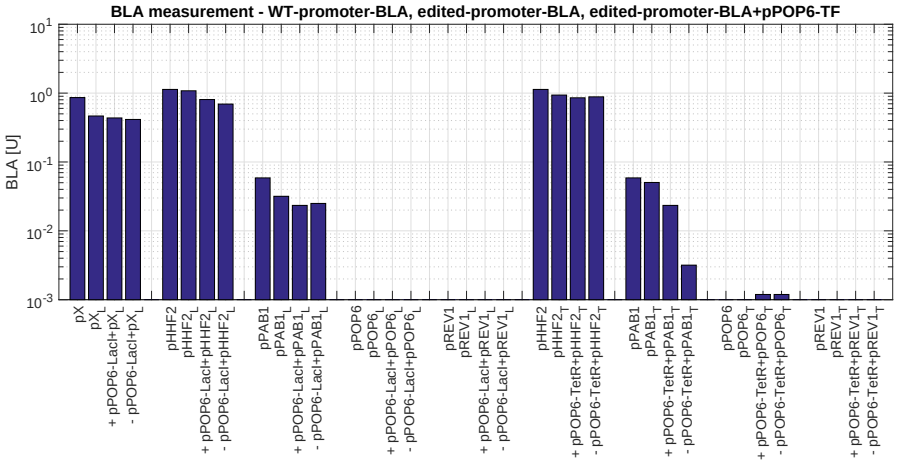


Obrázek 5.5: A) Ukázka vývoje popsaných metrik v čase včetně proložení lineární oblasti *D*4860 lineární funkcí. B) Sloupcový graf pro měření *BLA*.

[6]Ve zkumavce $0.1mM$ IPTG pro kmeny s *LacI*, $2\mu l$ doxycyklinu s koncentrací $20\mu g/ml$ pro kmeny s *TetR*.

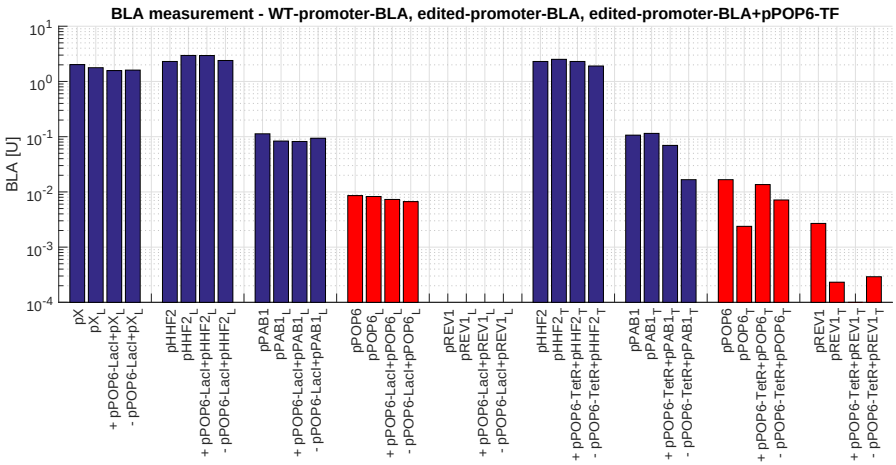
Příloha: Měření

Úvodní měření byly prováděny se shodným nastavením pro všechny vzorky. Kmeny obsahující před *BLA* variance promotoru *pX* a *pHHF2* byly naměřeny bez problémů. Pro *pPAB1* již došlo k očekávanému znatelnému poklesu signálu. Pro promotory *pPOP6* a *pREV1* však byl signál úplně neměřitelný. Na úvod měřený experiment je zobrazený v grafu 5.6, kde osa y je pro viditelnost malých signálů logaritmická.



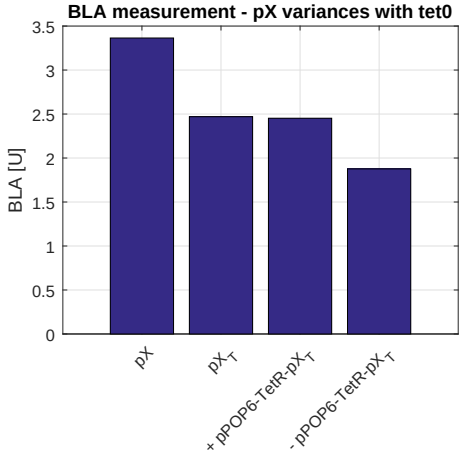
Obrázek 5.6: Graf pro úvodní měření produkce *BLA*. Graf obsahuje přírodní promotory s *BLA*, upravené promotory s *BLA* a upravené promotory s *BLA*, kde byl v kmenu i gen s regulačním proteinem. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-' a kde nebylo přidání induktoru relevantní není v legendě navíc nic.

Pro srovnání jsou v grafu 5.7 zobrazeny shodné kmeny, které byly měřeny již pomocí finálního nastavení měřícího protokolu. Kmeny s promotory *pPOP6* a *pREV1* byly inkubovány před měřením po dobu 4 hodin. V grafu jsou zvýrazněny červenou barvou, aby bylo odlišeno, že zde byl proveden pouze přepočet kompenzující rozdíl v naředění. Kompenzace doby inkubace nebyla prováděna. Kmeny s promotorem *pREV1* a s regulací *LacI* nebyly upraveným měřícím protokolem měřeny. V grafu je v této oblasti prázdné místo aby bylo zanecháno rozložení vzorků z předchozího grafu 5.6. Výsledky ukazují, že úprava měřícího protokolu napomohla k měřitelnosti i slabých promotorů *pPOP6* a *pREV1*.



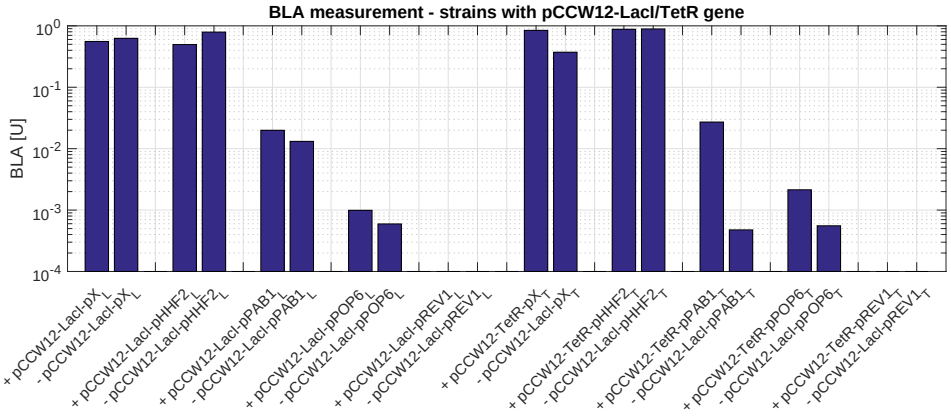
Obrázek 5.7: Graf měření produkce *BLA* s finálním nastavením měřícího protokolu. Graf obsahuje přírodní promotory s *BLA*, upravené promotory s *BLA* a upravené promotory s *BLA*, kde byl v kmenu i gen s regulačním proteinem. Červeně jsou označeny vzorky inkubované před měřením po dobu 4 hodin. Zbytek byl inkubován standardně nastavené 2 hodiny. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-' a kde nebylo přidání induktoru relevantní není v legendě navíc nic.

Z grafu 5.7 je vidět, že represe je pro kmeny s regulačním genem *pPOP6-LacI/TetR* patrná pouze u dvou konstruktů s *TetR*. Pro doplnění měření je dále uveden graf s promotorem pX_T 5.8,který z technických důvodů nebyl součástí předchozího měření. První vybraná bakteriální kolonie obsahovala chybnou assembly a kmeny s pX_T tak byly vytvořeny se zpožděním. Promotor pX_T je třetím z promotorů s patrnou represí.



Obrázek 5.8: Graf měření produkce *BLA* pro kmeny s integrovanými verzemi promotoru *pX* před *BLA* a *TetR* regulací. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-' a kde nebylo přidání induktoru relevantní není v legendě navíc nic.

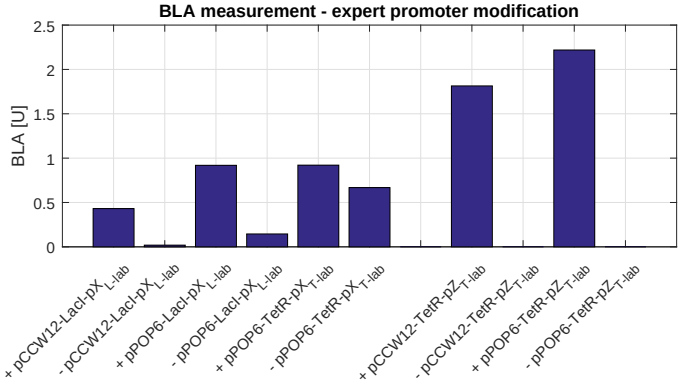
Následující graf obsahuje doplnění výsledků o kmeny, které obsahovaly konstrukt *pCCW12-LacI/TetR* (graf 5.9). Měřící protokol v této fázi ještě neobsahoval prodlouženou inkubaci slabých promotorů a ředění těchto vzorků bylo nastaveno na $OD600 = 0.15$. Pro promotor *pREV1* v tomto případě nebylo stále nic naměřeno.



Obrázek 5.9: Graf obsahující měření produkce *BLA* pro kmeny s integrovanými geny *pCCW12-LacI/TetR*. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-'.

Na základě výše uvedených grafů došlo k redukci kmenů pro závěrečný experiment, kde byly použity kmeny obsahující promotory pX_T, $pPAB1_T$ a $pPOP6_T$. Ostatní kmeny nejevily známky požadovaného represibilního chování a byly tak pro závěrečné měření popsané v podsekci 4.2.2 vynechány.

Posledním zde uvedeným měřením je experiment obsahující expertně upravené promotory *pX* a *pZ* (graf 5.10). Měření sloužilo jako kontrola, jestli sestavené geny *pCCW12/pPOP6-LacI/TetR* správně produkují regulační protein. Toto měření zároveň slouží jako srovnání s kvalitou expertně navržené represe oproti návrhu neuronovou sítí.



Obrázek 5.10: Graf obsahující měření produkce *BLA* pro expertně *pX* a *pZ*. Vzorky inkubované s induktorem mají v názvu '+', vzorky bez induktoru '-'.