UNIVERSIDADE DE SÃO PAULO
FACULDADE DE MEDICINA DE RIBEIRÃO PRETO
DEPARTAMENTO DE BIOLOGIA CELULAR E MOLECULAR

ADRIANO GOMES SILVA

**Desenho de novos promotores sintéticos de leveduras baseados em
abordagens *in silico***

Ribeirão Preto

2019

ADRIANO GOMES SILVA

**Desenho de novos promotores sintéticos de leveduras baseados em
abordagens *in silico***

Versão corrigida

A versão original encontra-se disponível tanto na Biblioteca da Unidade que aloja o Programa, quanto na Biblioteca Digital de Teses e Dissertações da USP (BDTD)

Dissertação apresentada à Faculdade de Medicina de Ribeirão Preto da Universidade de São Paulo para a obtenção do título de Mestre em Ciências.

Área de concentração: Biologia Celular e Molecular

Orientador: Prof. Dr. Rafael Silva Rocha

Ribeirão Preto

2019

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

SILVA, Adriano Gomes

Desenho de novos promotores sintéticos de leveduras baseados em abordagens in silico, 2019.

51 f. : il.

Dissertação (Mestrado em Ciências) – Programa de Pós-Graduação em Biologia Celular e Molecular, Universidade de São Paulo, Ribeirão Preto, 2019

1. Biologia Sintética. 2. Promotores sintéticos. 3. Elementos cis-regulatórios
4. Leveduras. 5. Algoritmos Genéticos. 6. Bioinformática. 7. Regulação transcricional.

Aluno: SILVA, Adriano Gomes

Título: Desenho de novos promotores sintéticos de leveduras baseados em abordagens *in silico*

Dissertação apresentada à Faculdade de Medicina de Ribeirão Preto da Universidade de São Paulo para obtenção do título de Mestre em Ciências.

Área de concentração: Biologia Celular e Molecular.

Aprovado em: 06/12/2019

Banca Examinadora

Prof. Dr.: Rafael Silva Rocha

Instituição: Faculdade de Medicina de Ribeirão Preto, Universidade de São Paulo

Julgamento: Aprovado

Prof. Dr.: Renato Tinós

Instituição: Faculdade de Filosofia, Ciências e Letras, Universidade de São Paulo

Julgamento: Aprovado

Profa. Dra.: Elizabeth Bilsland

Instituição: Instituto de Biologia, Universidade Estadual de Campinas

Julgamento: Aprovado

AGRADECIMENTOS

Agradeço aos meus pais, Iraci e Florisvaldo, por me trazerem ao mundo e por todo apoio afetivo e financeiro dado por todos esses anos da melhor forma que puderam. Também agradeço a todos familiares que me apoiaram, principalmente ao meu irmão André por me introduzir ao mundo da computação e da programação de computadores.

Ao meu orientador Rafael, por todos os discursos motivadores, ensinamentos, broncas e puxões de orelha que me garantiram diversos aprendizados, por toda liberdade dada para as escolhas desde o programa de pós-graduação ao desenvolvimento de todas as partes do projeto e por tornar o ambiente de pesquisa muito amigável. Obrigado por me motivar tão bem a mim e a todos os alunos que já passaram por suas aulas ou pelo laboratório nessa enormemente promissora área, mas ainda pouco explorada, que é a biologia sintética.

Aos amigos membros e ex-membros tanto do SSB Lab (Ananda, Bárbara, Bianca, Badain, Cauã, Cláudia, Felipe, Greicy, Juliana, Kauan, Larwsk, Leonardo, Lummy, Maria Clara, Marina, Murilo e Ninna) quanto do Laboratório de Metagenômica Funcional (Gabriel, Guilherme, Ítalo, Luana, Lucas, Tiago e Vanessa) por todo o apoio acadêmico teórico e prático, todas as festas, cervejas, momentos constrangedores, histórias e risadas.

À Juliana por sempre estar ao meu lado me apoiando nas disciplinas impossíveis da BioCel, nos experimentos que dão errado e nas minhas crises existenciais pré-deadlines. Obrigado por sempre ter estado comigo nesses momentos difíceis e nos momentos alegres que compartilhamos.

Ao Leonardo por me ensinar muito sobre todo esse universo de experimentos em biologia celular e molecular, sempre com a maior paciência, didática e primor do mundo. Obrigado por me ajudar sempre, mesmo estando ocupado, e não desistir de me ensinar, mesmo vendo a minha gigantesca dificuldade com a bancada.

À Faculdade de Medicina de Ribeirão Preto e principalmente ao Departamento de Biologia Celular, Molecular e Bioagentes Patogênicos e aos seus funcionários e professores. Obrigado pela disponibilização do espaço, dos equipamentos, do suporte dos professores e a todos eventos sempre enriquecedores aos participantes.

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) pelo financiamento da minha pesquisa (Processo FAPESP 2018/06509-6), dos eventos que participei e principalmente pela bolsa na qual pude me manter pesquisando com dedicação exclusiva.

Agradeço a todos que de alguma forma contribuíram positivamente com esse projeto.

“There is an art to flying, [...] or rather a knack. The knack lies in learning how to throw yourself at the ground and miss. [...] Clearly, it is this second part, the missing, that presents the difficulties.”

- Douglas Adams, Life, the Universe and Everything

RESUMO

Silva, Adriano Gomes. **Desenho de novos promotores sintéticos de leveduras baseados em abordagens in silico**. 2019. 49 f. Dissertação (Mestrado em Ciências) – Faculdade de Medicina de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto, 2019.

Microrganismos do grupo das leveduras são de altíssima relevância para diversas aplicações biotecnológicas. Nos últimos anos, diversos estudos têm focado os seus esforços na engenharia de circuitos sintéticos em leveduras visando tanto expandir as capacidades metabólicas e biossintéticas desses organismos quanto aumentar a tolerância destes a condições ambientais agressivas como aquelas relacionadas a diferentes processos industriais. Nesses trabalhos, têm sido observado que uma grande limitação para construção destes circuitos é o reduzido número de elementos regulatórios modulares e customizáveis que respondam às diferentes condições de interesse. Neste sentido, nosso grupo de pesquisa vem desenvolvendo nos últimos anos uma série de abordagens computacionais e experimentais para a busca e desenvolvimento de promotores sintéticos em bactérias, organismos modelo para estudos em biologia molecular e computacional. Nesta dissertação, exploramos a expansão dessas abordagens visando o desenho *in silico* e caracterização *in vivo* de novos promotores sintéticos para a levedura *Saccharomyces cerevisiae*. Dessa forma, o principal foco deste trabalho foi a construção de novas aplicações utilizando algoritmos genéticos que permitissem a combinação de diferentes elementos *cis*-regulatórios para fatores de transcrição de *S. cerevisiae*, objetivando a obtenção de novos elementos sintéticos de relevância biotecnológica, bem como de uma nova metodologia para a sua obtenção. Junto a isso também desenvolvemos uma ferramenta de visualização de informações relacionadas aos fatores de transcrição e expressões regulares em um conjunto de genes. A validação do modelo computacional não foi obtida nos testes *in vivo*, mas as informações obtidas das metodologias aplicadas podem auxiliar em futuros testes de validação.

Palavras-chave: biologia sintética, promotores sintéticos, leveduras, algoritmos genéticos, bioinformática, regulação transcricional, elementos *cis*-regulatórios

ABSTRACT

Silva, Adriano Gomes. **Design of novel synthetic yeast promoters based on in silico approaches**. 2019. 49 f. Dissertação (Mestrado em Ciências) – Faculdade de Medicina de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto, 2019.

Microorganisms of the yeast group are very important for several biotechnological applications. In recent years, several studies have focused on the engineering of synthetic circuits in yeasts aiming to both expand the metabolic and biosynthetic capacities of these organisms and increase their tolerance to aggressive environmental conditions such as those related to different industrial processes. In these works, it has been observed that a major limitation for the construction of synthetic circuits in yeast is the reduced number of modular and customizable regulatory elements that respond to the conditions of interest. Our group has been developing in recent years a series of computational and experimental approaches to the development of synthetic promoters in bacteria. In this dissertation, we will cover the expansion of these approaches aiming at the *in silico* design and *in vivo* characterization of new synthetic promoters for *Saccharomyces cerevisiae* yeast. Thus, the main focus of this work was the construction of new genetic algorithms that allow the combination of *cis*-regulatory elements for *S. cerevisiae* transcription factors. It is expected that new synthetic materials of biotechnological relevance will be obtained, as well as a new methodology for obtaining them. We have also developed a tool for visualizing information related to transcription factors and regular expressions in a set of genes. Model validation was not obtained, but the information obtained from the applied methodologies may help in future validation tests.

Keywords: synthetic biology, synthetic promoters, yeast, genetic algorithms, bioinformatics, transcriptional regulation, *cis*-regulatory elements

SUMÁRIO

1. INTRODUÇÃO	11
1.1. Biologia Sintética – Conceito, Aplicações e Limites	11
1.2. Redes de regulação gênica.....	13
1.2.1. Redes regulatórias eucarióticas como modelo	13
1.2.2. Regulação Transcricional e mecanismos gerais.....	13
1.2.3. Circuitos booleanos e promotores	14
1.3. Otimização	15
1.3.1. Inspiração biológica	16
1.4. Paralelização.....	18
1.5. Web Scrapping.....	18
2. OBJETIVOS	19
2.1. Objetivos específicos	19
3. Capítulo I: Método de evolução de elementos <i>cis</i> -regulatórios	20
3.1. Materiais e Métodos	20
3.2. Resultados	27
4. Capítulo II: Validação de promotores	29
4.1. Materiais e Métodos	29
4.1.1. Construção do vetor pLR1	29
4.1.2. Construção de promotores sintéticos	34
4.2. Resultados	37
5. Capítulo III: Desenvolvimento do software de visualização.....	39
6. CONCLUSÕES E DISCUSSÕES	47
7. SUGESTÕES PARA TRABALHOS FUTUROS	49
8. BIBLIOGRAFIA	50

1. INTRODUÇÃO

1.1. Biologia Sintética – Conceito, Aplicações e Limites

Nas últimas décadas, com o avanço nos conhecimentos de biologia molecular, foi possível perceber que na natureza são encontradas estruturas com funcionamento similar às utilizadas na computação [1, 2]. Neste ínterim, um exemplo disso são as sequências promotoras, sequências de DNA reconhecidas por proteínas específicas, chamadas de fatores de transcrição (FTs), responsáveis pelo auxílio na transcrição do DNA a fim de gerar uma nova fita de RNA, que por sua vez, pode ser traduzido em proteínas graças à maquinaria ribossomal. Sabendo disso, é possível modificar as sequências promotoras e FTs expressos em uma célula para formar estruturas semelhantes às portas lógicas com o objetivo de controlar funções biológicas [3, 4] (Figura 1), como já foi feito em nosso laboratório anteriormente utilizando algoritmos genéticos mais simples (sem cruzamentos, apenas mutações) e que foram aplicados a FTs e organismos procariotos.

Proteínas que diretamente ativam a transcrição, como FTs, ou indiretamente, como algumas proteínas fotorreceptoras ativadas por proteínas de sua via metabólica [5, 6], tornam possível a utilização destas biomoléculas como entradas de portas lógicas biológicas. As saídas destas portas lógicas podem ser proteínas não relacionadas à transcrição, como as proteínas repórteres, ou relacionadas, como FTs que agem na mesma célula ou até mesmo enzimas que geram moléculas capazes de atravessar a membrana celular e ativar FTs em células próximas [1, 7]. A possibilidade de desenhar sequências promotoras para a ativação ou inativação da expressão gênica é de extremo interesse científico e biotecnológico, pois tem a capacidade de melhorar o desempenho de diversos processos industriais e de ajudar a esclarecer diversas características biológicas relacionadas à transcrição [2, 3, 4].

Figura 1. Modelos de portas lógicas criados com módulos regulatórios procarióticos. As imagens no topo da figura simbolizam os atores no controle da atividade dos promotores: RNA polimerase (RNAP); FT (TFs); sequências de promotoras (Promoter DNA); sítio de ligação de fator de transcrição (TF site); e interação cooperativa (Cooperative interaction). (a) Porta lógica amplificadora (Amplifier) criada com um promotor que interage com TF ativador (A); (b) Porta lógica NOT criada com um promotor que interage com um TF inibidor de transcrição (B); (c) Porta lógica AND criada por um promotor que interage com um TF ativador (A) que precisa de uma molécula específica para ativação; (d) Porta lógica OR criada por dois TFs ativadores (A e B) que funcionam independentemente um do outro; (e) Porta lógica NAND criada por promotores que interagem com TFs inibidores (A e B) que impedem a ligação da RNA-polimerase (RNAP); (f) Porta lógica ANDN criada com um TF ativador (A) e um TF que impede a ligação (B) da RNA-polimerase (RNAP).

[1]

1.2. Redes de regulação gênica

O comportamento celular se baseia nas diversas interações entre suas moléculas. Todo o controle desse funcionamento é mediado por diversos reguladores moleculares que interagem entre si e com outras substâncias, principalmente na modificação da expressão de mRNA e tradução de proteínas [8]. O conjunto desses reguladores é chamado de rede de regulação gênica.

1.2.1. Redes regulatórias eucarióticas como modelo

Por conter uma série de organelas adicionais, como o retículo endoplasmático e complexo de Golgi, organismos eucariotos têm a capacidade de produzir proteínas de maior complexidade em comparação aos procariotos [9]. Neste sentido, microrganismos do grupo das leveduras são de altíssima relevância para diversas aplicações biotecnológicas. Nos últimos anos, diversos trabalhos são focados na engenharia de circuitos sintéticos em leveduras visando tanto expandir as capacidades metabólicas e biossintéticas desses organismos quanto aumentar a tolerância deste a condições ambientais agressivas como aquelas relacionadas a diferentes processos industriais [10, 11].

1.2.2. Regulação Transcricional e mecanismos gerais

A regulação da expressão eucariótica pode ocorrer em diversos pontos, desde a acessibilidade do DNA para a produção de mRNA, que pode ser regulada pela estrutura da cromatina, passando pela tradução, que podem ser aumentadas ou inibidas por RNAs reguladores, e o processamento de proteínas, que podem ter seu comportamento alterado por diversas modificações [12]. No estudo atual, nosso foco foi no principal ponto de controle, a regulação durante a transcrição feita por proteínas denominadas FTs ou elementos trans-regulatórios, que se ligam a sequências específicas de DNA, chamadas também de elementos *cis*-regulatórios, promovendo ou reprimindo a expressão de genes [13].

Os FTs se ligam a sequências específicas de DNA, facilitando (ativadores) ou bloqueando (repressores) que a RNA polimerase interaja com o promotor do gene. FTs se ligam geralmente nas proximidades do promotor, mas podem ser ocasionalmente encontrados distantes e mesmo assim afetarem a transcrição gênica. Mais de um fator pode controlar a ativação ou inativação de um gene, sendo chamada

de regulação combinatória [8, 15]. Tais FTs podem possuir parálogos, genes com funções distintas, como FTs de ativações ou sítios de ligação distintos, mas de mesma origem em uma duplicação genômica.

Para que um gene seja transcrito a enzima RNA polimerase deve se ligar com uma sequência de DNA chamada de promotor. Para se ligar ao promotor ela necessita de proteínas auxiliares chamadas de FTs gerais ou basais que quando ligadas aos seus elementos *cis*-regulatórios podem promover a montagem do complexo da RNA polimerase [14].

1.2.3. Circuitos booleanos e promotores

Os elementos básicos da computação são simples e utilizam a álgebra de dois valores, 0 e 1, conhecida como álgebra booleana. Circuitos fundamentais, como somadores e memórias, podem ser construídos da simples combinação de elementos básicos. Funções booleanas possuem N entradas, logo 2^N combinações de possíveis valores, e podem ser descritas em uma tabela denominada tabela-verdade ou por uma função booleana como, por exemplo, $F = A \cdot B$, uma porta AND [16].

1.2.3.1. Redesenho de comportamentos lógicos biológicos

Ao desenhar promotores é possível torná-los análogos às portas lógicas. Por exemplo, a porta OR funciona recebendo N entradas e caso alguma das portas esteja ativa sua saída será ativa, mas caso todas as portas estejam inativas a saída será inativa. No desenho de promotores isso seria similar a desenhar um promotor com capacidade de ligação a N diferentes FTs ativadores, tornando possível a expressão do gene relacionado ao promotor no caso da presença de qualquer um dos N FTs, mas não seria expressa se nenhum FT estivesse presente [1].

1.2.3.2. Matrizes de peso e posição (PWM)

Alinhando sequências predecessoras de genes obtidas por ChIP-seq ou predições computacionais é possível encontrar diversos padrões que representam os locais de ligação de determinados FTs no DNA. Esses padrões podem ser representados pelas matrizes de peso e posição (Figura 2). A partir de um conjunto de sequências alinhadas é possível definir uma matriz de peso e posição. As linhas são compostas pelo tipo de resíduo e as colunas às colunas do alinhamento [17]. Sabendo a PWM de um conjunto de FTs, é possível sugerir se a sequência pode se

ligar com grande afinidade a eles desde que a sequência se ajuste à todas as PWM do conjunto.

Figura 2. Metodologia de obtenção de uma PWM para um FT e do cálculo de Score. Sequências de DNA obtidas por chip-Seq alinhadas gerando uma Matriz de Frequência e Posição (PFM), que normalizada dá origem a uma Matriz de Proporção e Posição (PPM) e por fim é aplicado Log2, transformando a PPM em uma Matriz de Peso e Posição (PWM).

Sequências

1.3. Otimização

A teoria da otimização é o estudo quantitativo dos ótimos e dos métodos para buscá-los, que procuram melhorar o desempenho na resolução do problema sempre em direção aos pontos ótimos [18]. Os problemas de otimização possuem sempre uma função objetivo ou função de avaliação que queremos maximizar ou minimizar. Buscando os ótimos dessas funções de avaliação é possível encontrar os ótimos globais (máximo ou mínimo possível no espaço de busca) e os ótimos locais (máximo ou mínimo de uma região do espaço de busca e não do todo).

Dentre as diversas técnicas de otimização se encontram as técnicas metaheurísticas, utilizada para selecionar uma heurística que pode encontrar uma solução suficientemente boa para o problema de otimização com informação incompleta, imperfeita ou com limitada capacidade computacional. É possível encontrar soluções boas, mas não necessariamente ótimas, de forma muito mais eficiente que em métodos tradicionais, como a busca exaustiva, que faz a tentativa de todas as possibilidades, mas é inviável em tempo computacional para problemas com grandes quantidades de possibilidades.

1.3.1. Inspiração biológica

A evolução natural pode ser interpretada como um processo de otimização. Ela é paralela, eficiente, robusta, resolve problemas extremamente complexos, com um número grande de variáveis e com muitos ótimos locais [19]. Na seleção natural, indivíduos e populações competem por recursos. Indivíduos melhor adaptados à função de avaliação do ambiente terão mais descendentes e indivíduos que são menos adaptados geram menos. Não se conhece a priori quais alterações genéticas produzirão genótipos melhor adaptados, mas os genes são alterados aleatoriamente por mutações e recombinados por reprodução sexuada. O resultado disso é que os descendentes se tornam cada vez mais adaptados aos seus ambientes. Pensando na utilização das ideias de seleção natural em problemas de otimização, surge na década de 50 a computação evolutiva que inclui os algoritmos genéticos, programação genética e estratégias evolutivas [20].

1.3.1.1. Algoritmos evolutivos

Algoritmos evolutivos são métodos adaptativos utilizados na resolução de problemas de otimização inspirados nos mecanismos de adaptação genéticos de organismos biológicos. São metaheurísticas populacionais baseadas em busca aleatória. Neles, é possível “evoluir” soluções para problemas que não necessariamente envolvem organismos biológicos. Representando um problema adequadamente e as formas de avaliar as soluções apresentadas é possível gerar uma população de soluções candidatas que vão sofrendo mutações e cruzamentos durante diversas gerações, cada vez mais adaptada à forma de avaliação (Figura 3).

Figura 3. Representação dos algoritmos evolutivos. O processo inicia com uma população que pode ser aleatoriamente criada. Depois disso a população passa por uma avaliação para decidir quais soluções serão mantidas, são selecionadas para a próxima geração ou para a reprodução e, então, sofrem variação (mutação ou crossover).

Imagem inspirada em [19].

1.3.1.2. Algoritmos genéticos

Os algoritmos genéticos foram inicialmente desenvolvidos por John Holland, começaram a serem aplicados em otimização por Keneth DeJong e foram popularizados por David Goldberg [20].

Os elementos principais dos algoritmos genéticos são:

- a) População: subconjunto de possíveis soluções de um determinado problema, geralmente em número fixo. Os indivíduos da população são codificados em estruturas de dados ajustadas ao problema, sendo chamadas de cromossomos. Cada elemento do cromossomo é chamado de gene;
- b) Codificação: conjunto de genes, informações que definem as características do indivíduo. Exemplo: Strings, vetores, tuplas ou objetos;
- c) Função de Avaliação: Mede o grau de aptidão da solução (indivíduo) de acordo com o problema a ser resolvido;
- d) Seleção: Métodos para a seleção de indivíduos para a fase de reprodução, como:
 - Elitismo: Indivíduos com maior aptidão são selecionados;

- Método da roleta: Seleciona por probabilidade, sendo os mais aptos os com maior chance. O nível de diversidade da população é maior que no elitismo, mas decresce no decorrer do processo de otimização;
 - Torneio: k indivíduos são escolhidos aleatoriamente e o de maior aptidão é escolhido.
- e) Reprodução: Aplicada após a seleção de indivíduos (pais), permite a obtenção de novos indivíduos (filhos). Os operadores genéticos mais comuns são:
- Crossover: recombinação das características dos pais;
 - Mutação: alteração aleatória de genes no cromossomo filho.

Após algumas gerações a população pode convergir para um ótimo global. A convergência para um ponto em que não há melhora pode ser usada como critério de parada, assim como o tempo de execução, número de gerações e valores de aptidão mínimos e máximos.

1.4. Paralelização

Como o avanço na arquitetura de computadores, computadores com um único núcleo de processamento capazes de executar uma única tarefa foram dando lugar a processadores com cada vez mais núcleos (multiprocessamento) e capazes de executar múltiplas tarefas por núcleo (multitarefa). Essa nova arquitetura possibilitou o surgimento da computação paralela, computação na qual múltiplos cálculos podem ser realizados paralelamente, tornando o tempo de processamento de diversos algoritmos muito mais curto. Problemas podem ser divididos em pequenas partes e processados por núcleos diferentes e seus resultados são reunidos no final.

1.5. Web Scrapping

Existem diversos bancos de dados online com informações biológicas relevantes. Apesar de disponibilizarem os dados de forma fácil para pessoas entrando em suas páginas, esses bancos de dados muitas vezes não disponibilizam formas facilitadas de acesso para que programas possam extrair informações de forma automatizada. A comunicação facilitada entre aplicações deveria ser feita por uma tecnologia chamada de Web Service, mas na sua ausência é necessário utilizar técnicas mineração de dados. Web data scraping é uma técnica que permite programar robôs de busca para extrair informações específicas de sites sem suporte a Web Service [21].

2. OBJETIVOS

Este estudo teve como objetivo a criação de uma ferramenta computacional que recebe como entrada um conjunto de elementos *cis*-regulatórios e é capaz de gerar um conjunto de novas sequências de nucleotídeos que possuam alto score para as matrizes de peso e posição alvo e baixo score para as não alvo, com o objetivo de gerar elementos *cis*-regulatórios que tenham alta e/ou baixa afinidade para múltiplos FTs.

2.1. Objetivos específicos

- a) Desenho *in silico* da combinação de elementos *cis*-regulatórios com alta afinidade para os FTs;
- b) Desenvolvimento de uma ferramenta de uso livre para busca de novos FTs em sequências promotoras;
- c) Caracterização *in vivo* dos promotores sintéticos gerados pelo algoritmo em linhagens de levedura *S. cerevisiae* contendo as fusões de interesse.

3. Capítulo I: Método de evolução de elementos *cis*-regulatórios

3.1. Materiais e Métodos

Foi desenvolvido um algoritmo genético em Python, utilizando a biblioteca de paralelização Numba [22]. Numba faz com que os algoritmos em Python, que por padrão da linguagem são interpretados, sejam convertidos para a linguagem C e sejam compilados, tornando a sua execução mais rápida e paralelizada entre os processadores. Neste algoritmo é necessário entrar inicialmente com os seguintes parâmetros:

- a) Matrizes dos FTs: matrizes de dimensão $N \times 4$ sendo N correspondente ao tamanho do FT e as 4 dimensões correspondentes às 4 bases (A, T, G, C);
- b) Pesos das matrizes de posição: Pesos que serão dados a cada matriz de posição durante o cálculo do fitness;
- c) População inicial: Conjunto de sequências de DNA, que por padrão é vazio e gerado aleatoriamente durante a inicialização do algoritmo. Outras sequências podem ser inseridas para dar um viés às sequências iniciais ou para continuar o algoritmo a partir de um conjunto otimizado anteriormente;
- d) Gerações: Número máximo de iterações do algoritmo;
- e) Parar se estável: Por padrão falso, mas se verdadeiro para o algoritmo após N gerações sem modificação da sequência de melhor adaptação;
- f) Direção dos FTs: direção em que os FTs alvos e não alvos devem ser considerados nos cálculos, podendo ser apenas senso, apenas anti-senso ou ambas as direções;
- g) Tamanho individual: Número de bases que devem ter as sequências geradas;
- h) Tamanho da população: Quantidade de sequências que o algoritmo deve gerar.
- i) Viés da população: Por padrão falso, mas se verdadeiro gera sequências de DNA baseadas nas melhores sequências de ligação de um dos alvos escolhido aleatoriamente para cada indivíduo da população e caso o tamanho do alvo seja menor que o tamanho desejado da sequência, ela é completada com nucleotídeos aleatórios;
- j) Taxa de mutação: Taxa em que as sequências provenientes do cruzamento entre outras sequências são mutadas. Varia entre 0 e 1;
- k) Tipo de seleção: lista contendo tuplas com o tipo de seleção e a quantidade de indivíduos selecionados pelo tipo de seleção, permitindo assim que múltiplos tipos de seleção possam ser aplicados num mesmo algoritmo genético. Os

tipos de métodos de seleção implementados foram: elitismo, roleta, *rank-selection* e torneio;

- l) Máximo de repetições de base: Filtro que diminui drasticamente os scores de sequências com bases sequencialmente repetidas para evitar que as sequências finais geradas com muitas repetições possam causar problemas na construção dos pré-prints de DNA ou primers;
- m) Estacionário ou Geracional: Por padrão Estacionário, pois utilizando o elitismo para pelo menos um indivíduo permite que o(s) indivíduo(s) gerado(s) com maior score não seja(m) perdido(s) entre iterações. Na forma geracional os indivíduos da geração anterior são perdidos, restando apenas os descendentes gerados por cruzamento;
- n) Delta: Quantidade de vezes que a população cresce durante o cruzamento;
- o) Taxa de cruzamento: Taxa em que indivíduos selecionados aleatoriamente sofrem cruzamento. Varia entre 0 e 1;
- p) Fora de alcance: Por padrão falso, mas se verdadeiro permite que FTs sejam buscados mesmo nas bordas das sequências completando-os com o score da pior base possível, tornando possível localizar sítios ligação apenas com uma sequência parcial da borda que será ligada à outra sequência qualquer e que no mínimo teria a ligação demonstrada;
- q) Margem: Distância mínima desejada que os FTs alvo tenham dos não alvos. Por padrão 0, que faz com que o algoritmo tente maximizar a diferença de scores, mas quando outro valor é especificado os fatores não alvo na direção senso que ultrapassam a margem dos alvos na direção senso causam uma punição no fitness das sequências;
- r) Reversa: Por padrão falsa, mas se verdadeira a margem de distância entre os FTs não alvos também será considerada na direção anti-senso levando a punições no score;
- s) Normalizada: Por padrão falsa, mas se verdadeira os scores são normalizados utilizando as Fórmulas 1-4.

Fórmula 1. Sendo s um vetor correspondente à sequência de bases convertidas para números e T a matriz de peso e posição.

$$ScoresNorm(s, T) = \begin{cases} -(Score(s, T)/BestScore(T)), & Score(s, T) < 0 \\ Score(s, T)/WorstScore(T), & Score(s, T) \geq 0 \end{cases}$$

Fórmula 2. Sendo s um vetor correspondente à sequência de bases convertida para números e T a matriz de peso e posição.

$$Score(s, T) = \sum_{i=1}^N T(i, s_i)$$

Fórmula 3. T a matriz de peso e posição.

$$BestScore(T) = \sum_{i=0}^N \max\{T(i, j) \mid j \in [1, 4]\}$$

Fórmula 4. T a matriz de peso e posição.

$$WorstScore(T) = \sum_{i=0}^N \min\{T(i, j) \mid j \in [1, 4]\}$$

O algoritmo segue os passos padronizados de outros algoritmos genéticos [18] (Figura 4). Com as sequências iniciais aleatórias ou enviesadas dadas, o algoritmo faz a avaliação destas atribuindo um valor para cada uma delas com base nos scores máximos para cada matriz de peso e posição e na diferença de score entre os alvos e os não alvos. Os parálogos dos alvos não são incluídos, pois na maioria das vezes possuem o sítio de ligação tão similar aos alvos que acabam atrapalhando a otimização do algoritmo. As matrizes de peso e posição utilizadas vieram do banco de dados YeTFaSCo [23], que possui 244 matrizes de peso e posição (PWM) de FTs de *S. cerevisiae* de alta confiabilidade. Em seguida, as sequências são filtradas utilizando os métodos de seleção, sendo retiradas as sequências que vão além do tamanho determinado da população. No cruzamento, são selecionados 2 indivíduos aleatórios da população e em seguida é sorteado se o cruzamento ocorrerá ou não de acordo com a taxa determinada. Mutações são aplicadas nos novos indivíduos gerados de acordo com a taxa de mutação. Os novos indivíduos passam pela avaliação e caso o número de iterações tenha alcançado o número de gerações definido, o algoritmo retorna as sequências encontradas e seus respectivos scores. Caso contrário o

algoritmo segue para uma nova iteração até alcançar a estabilização ou o número de gerações definidas.

Figura 4. Representação gráfica da sequência de operações implementadas utilizando algoritmos genéticos.

A função mais importante do algoritmo é a função de avaliação, conforme ilustram as Fórmulas 5-8 e o algoritmo 1. Com ela são gerados os scores tornando possível a seleção, que se mostra eficiente ao causar grande evolução dos scores médios durante as iterações das gerações (Figura 5). O objetivo final são sequências que apresentem score alto, geralmente na direção senso, para os alvos e scores baixos para os não alvos tanto na direção senso quanto anti-senso (Figura 6), assim como é visto na distribuição da melhor sequência motivo de um FT qualquer. Devido aos diferentes tamanhos dos sítios de ligação dos FTs, as sequências finais podem possuir tanto sítios de ligação com posicionamentos separados quanto sobrepostos, dependendo apenas do melhor score encontrado pelo algoritmo para cada conjunto de posições (Figura 7).

Fórmula 5. Sendo s um vetor correspondente à sequência de bases convertida para números e T a matriz de peso e posição dos alvos e NT a dos não alvos.

$$Fitness(s, T, NT) = \frac{\sum_{i=1}^n MaxScore(s, T_i) - \sum_{j=1}^l MaxScoreNT(s, NT_j, limit(s, T))}{n+l} + margin(s, T, NT)$$

Fórmula 6. Sendo s um vetor correspondente à sequência de bases convertida para números e T a matriz de peso e posição.

$$limit(s, T) = \min\{MaxScore(s, T_i)\}$$

Fórmula 7. Sendo s um vetor correspondente à sequência de bases convertida para números e T a matriz de peso e posição dos alvos e NT a dos não alvos.

$$margin(s, T, NT) = \begin{cases} 0, & \text{se } \min\{MaxScore(s, T)\} - \max\{MaxScore(s, NT)\} < 0 \\ \min\{MaxScore(s, T)\} - \max\{MaxScore(s, NT)\}, & \text{caso contrário} \end{cases}$$

Fórmula 8. Sendo s um vetor correspondente à sequência de bases convertida para números e T a matriz de peso e posição dos alvos e NT a dos não alvos.

$$\begin{aligned} MaxScoreNT(s, NT, limit) &= \sqrt{\left(MaxScore(s, NT) - (limit - MaxScore(s, NT)) \right)^2} \\ &= \sqrt{(2 * MaxScore(s, NT) - limit)^2} \end{aligned}$$

Algoritmo 1. Representa a função $MaxScore(s, T)$. Sendo Scores a função já descrita pela Fórmula 2.

Entrada: vetor s de tamanho n , matrix T de tamanho m

Saída: escalar Max

$Max = 0.0$

Para i de 1 até $(n-m+1)$ **faça**

$Aux = Score(s[i..(i+m-1)], T)$

Se $Aux > Max$ **então**

$Max = Aux$

Fim Se

Fim Para

Figura 5 – Exemplo de evolução de um conjunto de sequências sendo otimizadas para ACE2 (verde) e GCN4 (azul). Average Fitness (vermelho) é o score médio do conjunto de sequências e Best Fitness (rosa) é o score da sequência que possui o melhor score para os dois alvos e a melhor distância dos não alvos.

Figura 6 - Exemplo de distribuição de scores das matrizes de peso dos FTs alvo (vermelho) GCN4 e ACE2, o parálogo de ACE2, SWI5 (verde), e os não alvos (azul) em uma sequência (topo, amarelo) resultante do algoritmo genético. No eixo X estão distribuídos os maiores scores no sentido senso e no eixo Y estão distribuídos os maiores scores no sentido anti-senso.

Figura 7 - Posição dos melhores scores na direção senso de uma sequência de 20pb otimizada para os FTs ACE2 e GCN4.

A saída dos dados é na forma de uma tabela contendo as sequências e seus respectivos scores da função fitness e das matrizes de posição dos alvos (Tabela 1).

Tabela 1. Exemplo de sequências e respectivos scores gerados na otimização para os FTs ACE2, GCN4 e STP1. Secção superior da população final de uma única execução com população de 100 indivíduos.

Sequence	Best Fitness	ACE2	GCN4	STP1
TGCTGCTGTAGTACTAGCA	0.5850355712878957	0.591481477828952	0.5617413837905779	0.572104100722688
AAGTGACTAGCACGACTGTG	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
TTGTGACTAGCACGACTGAA	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
AAGTGACTAGCACGACTGTT	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
GTGTGACTAGCACGACTGAG	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
TTGTGACTAGCACGACTGTT	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
TTGTGACTAGCACGACTGAT	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
CAGTGACTAGCACGACTGTA	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
CAGTGACTAGCACGACTGGG	0.5550388359411811	0.591481477828952	0.5617413837905779	0.507962334944565
AGGTGACTAGCACGACTGTG	0.16613245933871792	0.591481477828952	0.5617413837905779	0.507962334944565
GGATGCTGCTTGTGACTAGC	0.16504284178245396	0.5318196094677495	0.5617413837905779	0.5514338306878777
TTATGCTGCTTGTGACTAGC	0.16504284178245396	0.5318196094677495	0.5617413837905779	0.5514338306878777
TGCTGCTGAAGTACTAGCG	0.16164556538311206	0.5709784769916537	0.5617413837905779	0.572104100722688
GAGTGACTAGCACGACTGTG	0.14800463730154742	0.591481477828952	0.5617413837905779	0.507962334944565
TGCTGCTTAAATGACTAGCG	0.14311465934799514	0.5709784769916537	0.5534960241463708	0.5514338306878777
TGCTGCTTATATGACTAGCG	0.13328426247840144	0.5709784769916537	0.5534960241463708	0.5514338306878777
TGCTGCTGTAATGACTAGCA	0.13162748289726495	0.591481477828952	0.5534960241463708	0.572104100722688
TGCTGCTGGTGTGACTAGCA	0.12739370700341335	0.591481477828952	0.5617413837905779	0.572104100722688
GTAAAGACGACTGACTAGCA	0.12115929886631255	0.591481477828952	0.4311695915838745	0.4210903753137295
GGAAAGACGACTGACTAGCA	0.12115929886631255	0.591481477828952	0.4311695915838745	0.4210903753137295
TAGTAGACGACTGACTAGCA	0.12115929886631255	0.591481477828952	0.4311695915838745	0.4210903753137295
TGGTAGACGACTGACTAGCA	0.12115929886631255	0.591481477828952	0.4311695915838745	0.4210903753137295
GGTAAGACGACTGACTAGCA	0.12115929886631255	0.591481477828952	0.4311695915838745	0.4210903753137295

3.2. Resultados

Diversas sequências com tamanho definido em 20 pares de bases foram geradas para as combinações possíveis dos 4 FTs, sendo STP1, SFP1 e GCN4 na direção senso e ACE2 na direção senso e anti-senso, pois na literatura é descrito como capaz de regular a transcrição tanto no sentido senso quanto anti-senso [33]. Foram utilizados os mesmos pesos para todos os FTs, população inicial aleatória e estacionária de 100 indivíduos, taxa de mutação de 0.1, máximo de repetições de nucleotídeos em sequência igual a 5, 2 indivíduos selecionados por elitismo e 98 indivíduos selecionados por rank-seleção, limite de 100 gerações, taxa de cruzamento de 0.5 e margem automática. Os algoritmos genéticos para cada combinação dos 4 FTs foram rodados 10 vezes e as populações finais para cada uma das dez iterações foram concatenadas e ordenadas como uma única população para cada combinação. Das sequências geradas e ordenadas para cada combinação, foram selecionadas alguns dos melhores scores que apresentavam possivelmente melhor viabilidade (Tabela 5 e legenda categórica).

Tabela 5 - Sequências escolhidas para teste de resposta aos FTs.

<i>Alvos</i>	<i>Sequência</i>	<i>Categoria</i>	<i>Outliers</i>
<i>Negativa</i>	TGTATACAGTATACTGAATT	Muito Bom	0
<i>SFP1, ACE2</i>	ACCAGCATAAAAGTTTTTAG	Muito Bom	0
<i>SFP1, GCN4</i>	AGTGACTAGAAAAGTTTTTC	Mediano	0
<i>SFP1, STP1</i>	TACGGCGTAGAAAATTTTTTC	Bom	1
<i>ACE2, GCN4</i>	GACCAGCAGTGACTAAATGA	Bom	0
<i>ACE2, STP1</i>	GAAGACCAGCAATGCGGCGC	Excelente	0
<i>GCN4, STP1</i>	TGCGGCTTATGGGTGACACA	Bom	0
<i>SFP1, ACE2*</i>	TTGCTGGGAAAAAGTTTTTC	Bom	0
<i>ACE2*, GCN4</i>	TGCTGGGCTAAAATGAGTCG	Muito Bom	0
<i>ACE2*, STP1</i>	TGCGGCGTATGCTGGTCGTG	Excelente	0
<i>ACE2*, GCN4, STP1</i>	GTGAGTCGTTACGGCTGGTT	Muito Bom	0

*Anti-senso

Legenda categorias:

Excelente: $0.9 > \text{best score} \geq 0.8$

Muito bom: $0.8 > \text{best score} \geq 0.7$

Bom: $0.7 > \text{best score} \geq 0.6$

Mediano: $0.6 > \text{best score} \geq 0.5$

4. Capítulo II: Validação de promotores

4.1. Materiais e Métodos

4.1.1. Construção do vetor pLR1

Com o objetivo de caracterizar as propriedades de expressão moduladas pelos promotores descritos anteriormente, delineamos um plasmídeo para quantificação da expressão de um gene *repórter*. Para este fim, este plasmídeo foi construído com um gene *repórter* da proteína fluorescente verde (eGFP), proveniente do vetor pMCB17.apx e uma sequência terminadora do gene *tps1* de *S. cerevisiae*, apresentando as seguintes sequências, respectivamente. A escolha deste terminador foi baseada na literatura científica que mostrou que a utilização do mesmo aumenta consideravelmente a meia vida do mRNA e rendimento de produção de proteínas em *S. cerevisiae* [24]

eGFP (729 pares de base):

```
ATGAGTAAAGGAGAAGAAGAACTTTTCACTGGAGTTGTCCCAATTCTTGTTGAATTAGATGGTGATGTTAATGGGC
ACAAATTTTCTGTCAAGTGGAGAGGGTGAAGGTGATGCAACATACGGAAAACCTTACCCTTAAATTTATTTGCAC
TACTGGAAAACCTGTTCCATGGCCAACTTGTCACTACTTTTACCTATGGTGTTCATGCTTTTCAAGAT
ACCCAGATCATATGAAGCGGCACGACTTCTTCAAGAGCGCCATGCCTGAGGGATACGTGCAGGAGAGGACC
ATCTTCTTCAAGGACGACGGGAACTACAAGACACGTGCTGAAGTCAAGTTTGAGGGAGACACCCTCGTCAAC
AGGATCGAGCTTAAGGGAATCGATTTCAGGAGGACGAAACATCCTCGGCCACAAGTTGGAATACAACACTAC
AACTCCCAACGATATACATCATGGCCGACAAGCAAAAGAACGGCATCAAAGCCAACCTTCAAGACCCGCCAC
AACATCGAAGACGGCGGCGTGCAACTCGCTGATCATTATCAACAAAATACTCCAATTGGCGATGGCCCTGTC
CTTTTACCAGACAACCATTACCTGTCCACACAATCTGCCCTTTCGAAAGATCCCAACGAAAAGAGAGACCACA
TGGTCCTTCTTGAGTTTGTAAAGCTGCTGGGATTACACATGGCATGGATGAACTATACAAAGGCGCGCCGG
CTTAA
```

Terminador do gene *tps1* (211 pares de base):

```
TGAACCCGATGCAAATGAGACGATCGTCTATTCTGGTCCGGTTTTCTCTGCCCTCTCTTCTATTCACTTTTTT
TATACTTTATATAAAATTATATAAATGACATAACTGAAACGCCACACGTCCTCTCCTATTGTTAACGCCTGTCT
GTAGCGCTGTTACTGAAGCTGCGCAAGTAGTTTTTACCGTATAGGCCCTCTTTTCTCTCTCTTTCTTTCTC
TCCCGCGCTGATCTCTTCTTCGAAACA
```

Para a amplificação dos fragmentos, os oligonucleotídeos utilizados estão mostrados na Tabela 2. A amplificação de cada fragmento por PCR e os oligonucleotídeos usados em cada reação estão descritos na Tabela 3.

Tabela 2 - Oligonucleotídeos utilizados para a amplificação dos fragmentos.

Nº	ID	Sequência	T _m (°C)
1	5' KpnI_GFP	GCGCGGTACCATGAGTAAAGGAGAAGAAGCTTTTCACTGGAG	60
2	3' GFP_Ttps1	TCGTCTCATTTGCATCGGGTTCATTAAGCCGGCGCGCCTTT	60
3	5' Ttps1_GFP	AAAGGCGCGCCGGCTTAATGAACCCGATGCAAATGAGACGA	60
4	3' Ttps1_SacI	GCGCGAGCTCTGTTTCGAAGAAGAGATCAGCGCG	60
5	5' HindIII_tef1 promoter	GCGCAAGCTTCACACACCATAGCTTCAAAATGTTTCTAC	60
6	3' tef1 promoter BamHI	GCGCGGATCCCTTAGATTAGATTGCTATGCTTTCTTTCTAATGAGC	60

Tabela 3 - Tamanho dos fragmentos obtidos após amplificação com primers específicos.

Fragmento	Oligonucleotídeos	Tamanho (pb)*
GFP	1 e 2	729
Ttps1	3 e 4	211
GFP-Ttps1	1 e 4	940
Ptef1	5 e 6	419

*Não se considerou os sítios de restrição

A amplificação do fragmento GFP e *Ttps1* foi realizada utilizando-se os primers da Tabela 2 utilizando-se a enzima *Phusion DNA Polymerase (New England Biolabs)* de acordo com as recomendações do fabricante. As seguintes condições de ciclagem foram empregadas:

- a) 98°C – 0:30s
- b) 98°C – 0:10s
- c) 60°C – 0:30s
- d) 72°C – 0:30s
- e) 72°C – 10:00
- f) 12°C – ∞

A reação ocorreu com 35 ciclos de repetição de b-c-d. Os fragmentos foram quantificados e utilizados para um ensaio de equimolaridade e submetidos à uma PCR de sobreposição e posterior ligação ao vetor pUC-19, conforme ilustra a Figura 8.

A sequência GFP-*Ttps1* foi ligada ao vetor pUC-19 devido a este ser um vetor de alto número de cópias, o que permite que os passos de clonagem e verificação sejam feitos com maior facilidade. Para tais seleções, o vetor contendo a sequência

de interesse foi inserido em células bacterianas *Escherichia coli* DH5 α e selecionados em meio Luria Bertani (LB) na presença de Ampicilina (Concentração Final de 150 μ g/uL).

A fim de validar o plasmídeo construído e verificar seu funcionamento na linhagem de levedura *S. cerevisiae* SC9721 (*MAT α his3- Δ 200 URA3-52 leu2 Δ 1 lys2 Δ 202 trp1 Δ 63*) (Fungal Genetic Stock Center – www.fgsc.net), a sequência GFP-*Ttps1* foi extraída do vetor pUC19 e subclonado no vetor pRS426 (amp^R lacZ URA3) que contém uma origem de replicação em leveduras (2 μ). Para tal, o vetor pUC19 contendo o fragmento de interesse pRS426 foram digeridos com as enzimas HindIII e SacI para liberação do inserto e abertura da dupla fita para recepção deste inserto, respectivamente. A seguir, os fragmentos foram ligados um ao outro pela ação da T4 DNA Ligase (Thermo) conforme recomendações do fabricante. Foram utilizados 5 μ L da reação de ligação para a transformação genética de *E. coli* DH5 α que permitiu a correta seleção dos fragmentos que foram corretamente ligados e exibiram fenótipo resistente à ampicilina em meio LB. As colônias positivas foram isoladas e tiveram os plasmídeos extraídos (através de MiniPrep Extraction Kit – Promega) para posterior transformação genética de levedura. A este plasmídeo pRS426 contendo o fragmento GFP-*Ttps1* deu-se o nome pLR1 (Figura 9).

Como controle positivo de que o plasmídeo pode estar fazendo as células expressarem, clonamos o promotor constitutivo do gene *tef1* de *S. cerevisiae* no vetor *pLR1*. Amplificamos, através de PCR, utilizando os primers 5 e 6 (Tabela 2), a sequência correspondente a 419 pb deste promotor. Esses primers eram flanqueados por sítios de enzimas de restrição *HindIII* e *BamHI* para clonagem no vetor *pLR1* *upstream* à sequência de GFP previamente clonada. Os ensaios de clonagem foram realizados conforme descrito anteriormente e verificados por digestão quanto às corretas inserções após transformação bacteriana e seleção em LB e ampicilina. O vetor contendo este promotor é ilustrado na Figura 10.

Figura 8 - PCR de sobreposição utilizada na construção dos fragmentos de eGFP e Ttps1 para posterior ligação no vetor pUC19.

Figura 9 - pLR1. Fragmento de GFP-Ttps1 inserido no vetor pRS426.

Figura 11. Placa do teste da expressão das colônias. (A) Foto das colônias em campo claro e (B) na presença de UV, apresentando fluorescência nas colônias transformadas com *pLR1+P_{tef1}*.

Como controle, as células SC-9721 sem receber nenhum plasmídeo foram plaqueadas no meio, mostrando que a ausência de auxotrofia inviabiliza o crescimento das colônias. Ademais, podemos observar pela Figura 11 (parte A – campo claro e parte B – visualização de fluorescência) que apenas as células transformadas com o vetor pLR1 contendo o promotor *P_{tef1}* apresentaram fluorescência, mostrando que o plasmídeo é funcional e não permite vazamento de expressão na ausência de um promotor.

4.1.2. Construção de promotores sintéticos

Para a construção dos promotores sintéticos baseados na estratégia estabelecida *in silico*, utilizamos a sequência denominada de core 1 (elemento básico para transcrição) determinada por Heidi Redden [26]. Este core totalmente sintético foi otimizado para fins de expressão em levedura e possui alta força para recrutamento da maquinaria de transcrição da célula, possibilitando a transcrição mesmo sem a presença de elementos ativadores *upstream* (UAS) a ele. O core apresenta a seguinte sequência de nucleotídeos, disposta na orientação 5'-3'.

TATAAAAGAGCACTGTTGGGCGTGAGTGGAGGCGCCGGAAAAAGCATCGAAAAAAG

Além disso, este DNA foi sintetizada de modo a conter um sítio de *BamHI* flanqueando a extremidade 3' para posterior ligação ao vetor pLR1, disponível na coleção de vetores e plasmídeos do laboratório de Biologia Sistêmica e Sintética. Cada um dos elementos UAS escolhidos com base nas predições do programa (Tabela 4) foram sintetizados e ligados ao core promoter utilizando a metodologia de oligonucleotídeos fosforilados em sua extremidade 5' [27,28].

Tabela 4 - Elementos UAS escolhidos com base nas predições do programa

Nome	Sequência
5' CORE	TATAAAAGAGCACTGTTGGGCGTGAGTGGAGGCGCCGGAAGCATCGAAAAAG
3' CORE	GATCCTTTTTTCGATGCTTTTTCCGGCGCCTCCACTCACGCCAACAGTGCTCTTT
5' SFP1, ACE2	AGCTTGGCGCGCCACCAGCATAAAAGTTTTAGTTAATTAACCTGTAATATTCTACCCAAGCT
3' SFP1, ACE2	TATAAGCTTGGGTAGAATATTACAAGTTAATTAACATAAACTTTATGCTGGTGGCGCGCCA
5' ACE2, GCN4	AGCTTGGCGCGCCGACCAGCAGTGACTAAATGATTAATTAACCTGTAATATTCTACCCAAGCT
3' ACE2, GCN4	TATAAGCTTGGGTAGAATATTACAAGTTAATTAATCATTAGTCACTGCTGGTGGCGCGCCA
5' ACE2, STP1	AGCTTGGCGCGCCGAAGACCAGCAATGCGGCGCTTAATTAACCTGTAATATTCTACCCAAGCT
3' ACE2, STP1	TATAAGCTTGGGTAGAATATTACAAGTTAATTAAGCGCCGATTGCTGGTCTTCGGCGCGCCA
5' GCN4, STP1	AGCTTGGCGCGCCTGCGGCTTATGGGTGACACATTAATTAACCTGTAATATTCTACCCAAGCT
3' GCN4, STP1	TATAAGCTTGGGTAGAATATTACAAGTTAATTAATGTGTCACCCATAAGCCGCAGGCGCGCCA
5' SFP1, ACE2, GCN4	AGCTTGGCGCGCCAGTGACTAGCAAAATTTCTTTAATTAACCTGTAATATTCTACCCAAGCT
3' SFP1, ACE2, GCN4	TATAAGCTTGGGTAGAATATTACAAGTTAATTAAGAAAAATTTGCTAGTCACTGGCGCGCCA
5' NEGATIVA	AGCTTGGCGCGCCTGTATACAGTATACTGAATTTTAATTAACCTGTAATATTCTACCCAAGCT
3' NEGATIVA	TATAAGCTTGGGTAGAATATTACAAGTTAATTAATTCAGTATACTGTATACAGGCGCGCCA

O ensaio de ligação foi realizado utilizando-se T4 DNA ligase conforme as recomendações do fabricante. Após ligação, 50 µL de células *E. coli* DH10β foram utilizadas para transformação com 5 µL de DNA plasmidial advindo da ligação. A transformação aconteceu por 3 min a 37°C e as células foram recuperadas em placas de meio LB na presença de Ampicilina (150 µg/mL).

Para confirmar a correta ligação dos fragmentos no vetor, colônias isoladas foram utilizadas para procedimento de extração de DNA plasmidial usando o kit Wizard® Plus SV Miniprep Purification Kit System (Promega). Após extração de DNA plasmidial, os vetores foram digeridos com as enzimas EcoRV e PacI, para verificar se os promotores sintéticos foram montados corretamente. O perfil de restrição esperado consistia no aparecimento de duas bandas, de tamanho aproximados de 5301 pb e 1463 pb, o qual pode ser visualizado no gel (Figura 12).

Figura 12 – Resultado da digestão por EcoRV e PacI dos vetores de pLR1 com os promotores inseridos.

Para todos os promotores, pelo menos uma colônia apresentava a correta combinação de DNA vetorial e promotor sintético montada corretamente. Como os promotores foram construídos corretamente, os vetores então foram utilizados para transformações utilizando o método de Acetato de Lítio, descrito por Gietz [29]. As colônias resultantes foram plaqueadas em Meio Sintético Completo (YNBG – Yeast Nitrogen Base + Glicose 2%) na ausência de uridina para seleção auxotrófica a 30°C.

Após 4 dias de crescimento as colônias resultantes de cada um dos 4 quadrantes da placa de transformação foram recuperadas com meio YNBG e crescidas por 48h a 30°C para obtenção de massa celular a fim de se proceder com experimentos de extração de DNA. 3 colônias diferentes de cada placa, ou seja, contendo o vetor portador de promotores sintéticos diferentes, foram inoculadas em 5 mL de YNBG (YNB+Glicose 2%) a 30°C por 16h. Após este período, as leveduras foram centrifugadas e ressuspensas em meio YNBG preparado nos tampões ácido cítrico-citrato de sódio (100 mM) nos diferentes pH: 3, 4 e 5. Todas elas, em seguida, foram cultivadas por 8h na presença dos indutores (ácidos) e após este tempo foram utilizadas para medições de fluorescência. Como controles experimentais, utilizamos o vetor pLR1+Ptef1 como controle positivo de expressão e por fim, o vetor pLR1 não contendo nenhum promotor *upstream* ao gene da proteína GFP (a fim de verificar se o plasmídeo possuía *leaking* de expressão). Os resultados obtidos foram expressos como o *fold-change* da média de fluorescência de cada um dos promotores avaliados em relação ao controle negativo. Quando aplicados, testes de significância estatística foram realizados utilizando considerações de variância por ANOVA one-way ou ANOVA two-way com pós teste de múltiplas comparações de Dunnet ou Tukey, ambos com p-value inferior a 0.05.

4.2. Resultados

Utilizamos 200 μ L de cada replicata biológica para leitura de GFP em comprimento de onda $\sim$ 545 nm no Victor X3, uma leitura única e pontual após essas 8h. Os resultados foram obtidos com replicatas técnicas de cada replicata biológica (3). O valor do resultado é GFP/OD e os gráficos são apresentados em relação ao fold-change do controle negativo (pLR1 vazio, sem promotor).

Os promotores não apresentaram diferença estatística significativa na expressão de GFP entre os diferentes pHs em relação ao controle positivo, o promotor forte P_{tef1} (Figuras 21). Os promotores também não apresentaram diferença significativa em relação ao controle na ausência de indução (Figura 22). Um aumento da expressão em relação ao controle YNBG ocorre em todos os promotores, esse aumento, por também ocorrer em P_{tef1}, pode ser atribuído à algum mecanismo celular e não à responsividade dos promotores sintéticos ao pH.

Figura 21 – Estatística referente a um teste ANOVA-one-way com Pós teste de Dunnet (múltiplas comparações) com $p < 0.05$ em relação ao controle positivo P_{tef1}. * para $0.01 < p < 0.05$; ** para $0.001 < p < 0.01$; *** para $0.0001 < p < 0.001$; e **** para $p < 0.0001$.

Figura 22 - Estatística referente a um teste ANOVA-two-way com Pós teste de Tukey (múltiplas comparações) com $p < 0.05$ em relação ao controle sem induzir (YNBG Control pH 5.6). a para $0.01 < p < 0.05$; b para $0.001 < p < 0.01$; c para $0.0001 < p < 0.001$; e d para $p < 0.0001$.

5. Capítulo III: Desenvolvimento do software de visualização

Para melhorar a visualização e facilitar a análise dos dados gerados pelo algoritmo genético, decidimos desenvolver também um visualizador, pois seria útil não apenas ao nosso propósito, mas à visualização de sequências sintéticas ou mesmo de sequências naturais. O software visualizador foi desenvolvido em Python utilizando como biblioteca de interface a biblioteca nativa Tkinter, evitando assim a necessidade de os usuários terem de instalar novas bibliotecas visuais. O software é compatível com os sistemas operacionais Windows, Linux e Mac OS. Os gráficos do software são gerados automaticamente utilizando as bibliotecas Canvas e Matplotlib [30].

Inicialmente o usuário pode adicionar uma sequência manualmente ou adicionar um arquivo do tipo FASTA ou GBFF ou um arquivo FASTA e sua anotação GFF (Figura 13). Os arquivos GBFF ou FASTA com anotação são separados em genes, permitindo a seleção da quantidade de pares de base *upstream* e *downstream* tendo a posição de início do gene como referência. As sequências inseridas no programa podem ser exportadas no formato FASTA caso o usuário sinta a necessidade, bastando que o usuário selecione as sequências escolhidas e clique no botão Export. Também é possível filtrar os genes selecionados por regulador (Figura 13b), sendo atualmente a seleção de reguladores pré-carregada com reguladores de *S. cerevisiae* vindos do banco de dados YeastGenome [31], obtidos por meio de técnicas de web scraping, mas pretendemos futuramente inserir dados de outros bancos de reguladores e permitir a inserção de reguladores ou grupos de genes determinados pelos próprios usuários.

Após a seleção das sequências que o usuário deseja que sejam analisadas é possível selecionar o conjunto de FTs pelos quais as sequências terão os scores calculados (Figura 14a). Os FTs são representados utilizando as matrizes de peso e frequência (PFM) do banco de dados YeTFaSCo que são normalizadas (Fórmula 9) e transformadas em matrizes de peso e posição (PWM) (Fórmula 10) de acordo com os dados de background determinados pelo usuário, que podem se apresentar sob os seguintes padrões: *Global*, calculado com base na frequência de bases do genoma inserido; *Local*, baseado no cálculo das bases apenas no intervalo upstream e downstream determinado pelo usuário; ou *Custom*, no qual o usuário digita cada uma das frequências de bases. É possível também determinar a direção na qual as matrizes serão calculadas, o limiar mínimo (Threshold) para a contabilização do sítio

de ligação e o tipo de valor utilizado, sendo o Score padrão (Fórmula 2) ou o Score normalizado (Fórmula 1). Também é possível realizar a busca por sequências de bases no formato IUPAC e/ou de expressões regulares (Figura 14b).

Figura 13. Programa (a) antes e (b) depois do carregamento dos de um arquivo GBFF. (c) Filtragem de genes por regulador.

Fórmula 9. Sendo F a matriz de posição e frequência (PFM) e P a matriz de posição e proporção (PPM).

$$P_{i,j} = \frac{F_{i,j}}{\sum_{k=1}^4 F_{i,k}}$$

Fórmula 10. Sendo W a matriz de peso e posição (PWM), P a matriz de posição e proporção (PPM) e b o vetor de backgrounds.

$$W_{i,j} = \log_2 \left(\frac{P_{i,j}}{b_j} \right)$$

Ao calcular os scores é possível visualizá-los calculados de diversas maneiras. Uma delas é a *Global view* que permite que sejam marcados os supostos sítios de ligação dos FTs em cada uma das sequências escolhidas (Figura 15) de acordo com os parâmetros definidos para os FTs, como direção, *threshold*, tipo de score e *background*. Outra maneira de visualização é a *Global distribution*, que permite gerar histogramas dos dados relacionados aos scores, como histogramas de posição (Figura 16), scores (Figura 17) e ocorrências (Figura 18) nas regiões definidas do total de sequências selecionadas. Nos histogramas é possível escolher as ocorrências em quais fitas devem ser mostradas (senso ou anti-senso), se serão empilhadas ou somadas. Os mesmos gráficos podem ser feitos para expressões regulares e/ou IUPAC.

Figura 14. Abas de busca por: (a) FTs e (b) bases no formato IUPAC e/ou expressões regulares.

Sequences
Position Matrix
IUPAC/Regex

Database: YeTFaSCo
Specie:

Background: A: 0.30926067513606237 T: 0.30926067513606237
Global G: 0.19073932486393758 C: 0.19073932486393758

Direction: Both
Threshold: 0.8
Normalized Score

Search: 244 found (0 selected)

☐	Motif ID	Gene Name	Locus Tag	Thr.	±
☐	2192	YKL222C	YKL222C	-	
☐	1506	ARG81	YML099C	-	
☐	2203	FHL1	YPR104C	-	
☐	400	STE12	YHR084W	-	
☐	962	GAT1	YFL021W	-	
☐	2174	STP2	YHR006W	-	
☐	541	ABF2	YMR072W	-	
☐	494	TOS8	YGL096W	-	
☐	830	FKH2	YNL068C	-	
☐	2239	UME6	YDR207C	-	
☐	2144	MIG3	YER028C	-	
☐	2226	RDS2	YPL133C	-	
☐	757	RDS2	YPL133C	-	
☐	2233	STB3	YDR169C	-	
☐	67	FLO8	YER109C	-	
☐	486	UGA3	YDL170W	-	

Reset

Sequences
Position Matrix
IUPAC/Regex

Add
Edit
Remove
Reset
Export

ID

Expression

New expression

ID:

Expression:

Cancel

Insert

(a)

(b)

Figura 15. Visualização dos supostos posicionamentos dos sítios de ligação nas sequências de acordo com os FTs e parâmetros selecionados.

Figura 16. Histogramas de posição de ocorrência para (a) um conjunto de FTs em ambos as direções somadas sobre todas as sequências do genoma de *S. cerevisiae* e (b) do FT AZF1 nas direções senso e anti-senso separadas.

Figura 17. Histograma de scores por sequência dos FTs selecionados sobre 1000 pb *upstream* de todos os genes de *S. cerevisiae* a partir dos scores maiores que 0.2 na direção senso.

Figura 18. Histograma de quantidade de ocorrências por sequência dos FTs selecionados sobre 1000 pb *upstream* de todos os genes de *S. cerevisiae*.

Figura 19. Histograma de posições de ocorrência de (a) ATG e (b) TATA-box (TATATWTR) sobre 1000 pb *upstream* de todos os genes de *S. cerevisiae*.

(a)

(b)

Outros gráficos possíveis de serem gerados são os gráficos individuais de distribuição (Figura 20a). Estes gráficos permitem ter noção dos FTs com maiores scores nos sentidos senso e anti-senso no intervalo selecionado que podem ter influência na transcrição do gene selecionado, e de ocorrências na sequência (Figura 20b), que permite visualizar as ocorrências e sobreposições dos FTs também durante todo o intervalo selecionado e podem ter influência na transcrição do gene selecionado. A primeira versão do artigo descrevendo a nova ferramenta foi

depositada na plataforma bioRxiv [32] e o software disponibilizado na plataforma GitHub (github.com/adri4nogomes/TFound).

Figura 20. Gráficos individuais. (a) Distribuição dos maiores scores por FT em cada direção e (b) ocorrências na sequência, com detalhamento das sobreposições e dos scores em cada posição.

(a)

(b)

6. CONCLUSÕES E DISCUSSÕES

Os resultados desta dissertação nos levam às seguintes conclusões:

- a) É possível gerar sequências de DNA com alto score para um conjunto de matrizes de peso e posição ao mesmo tempo que possuem baixo score para um segundo conjunto de matrizes de peso e posição utilizando algoritmos genéticos. Desta otimização foi possível perceber que quanto menor o tamanho da sequência e/ou maior a quantidade de alvos e não-alvos menor será a chance de obtenção de sequências otimizadas.
- b) Não foi possível obter elementos *cis*-regulatórios responsivos ao pH utilizando a metodologia de construção de promotores aqui descrita. Tal fato abre um leque de possibilidades que poderão ser testadas futuramente:
 - O core escolhido apresenta ativação muito forte, mesmo na ausência de UAS, podendo estar atrapalhando a visualização de variação na presença de UAS que causam pouca variação de expressão. Neste caso a troca do elemento core poderia evidenciar se esta seria a falha;
 - Os FTs gerais que se ligam aos sítios de ligação do core escolhido e fazem a montagem do aparelho de transcrição podem não interagir com os FTs alvo do projeto. Neste caso a troca por outros FTs alvo seriam uma nova possibilidade de testes;
 - A variação de pH não influencia fortemente na ativação dos FTs alvo do projeto. Esta possibilidade pode ser testada utilizando outras condições descritas na literatura como ativadoras dos FTs alvo escolhidos;
 - O score obtido para os elementos *cis*-regulatórios não é suficientemente alto para a ligação com o FT ou as matrizes de peso e posição não são confiáveis o suficiente para o uso. Isto poderia ser testado trocando os UAS gerados por um UAS de score alto para apenas um FT obtido a partir da matriz de peso. Caso houvesse variação na expressão, isso evidenciaria a confiabilidade das matrizes de peso e posição e o score insuficiente dos UAS gerados. Caso contrário, evidenciaria a não confiabilidade das matrizes de peso e posição ou a não interação do FT com a maquinaria relacionada ao core escolhido;

- Outros FTs podem ainda estar interferindo ou bloqueando a ligação dos FTs alvo ou por ainda não terem sido descritos por matrizes de peso e posição ou por terem uma prioridade pela ligação aos elementos cis-regulatórios gerados. Ex.: Parálogos que ainda não foram bem caracterizados;
- O tempo de indução pode ter sido muito grande (8 horas) podendo fazer com que a diferença na expressão tenha sido perdida, pois a célula teve tempo o suficiente para se adaptar à diferença de acidez. O processo pode ser repetido com um menor tempo de indução (1 a 4 horas) e a indução possa ser melhor visualizada. A medição da cinética em intervalos periódicos entre o início e o fim da indução também seriam de grande utilidade;
- Mudança do terminador T_{TPS1} , pois por ter uma meia vida grande demais possa estar impedindo a visualização de rápidas mudanças de expressão;
- O uso de algum promotor responsivo à acidez como controle positivo.

7. SUGESTÕES PARA TRABALHOS FUTUROS

As principais implementações no algoritmo genético seriam: a possibilidade de variação de tamanho das sequências geradas, possibilitando que o algoritmo se auto-ajuste para gerar sequências mínimas dentro de um intervalo estabelecido com as características de afinidade especificadas pelo usuário; a criação de uma interface amigável aos usuários para a sua fácil utilização; e a possibilidade de especificar não apenas os alvos, mas a arquitetura que os alvos devem seguir, utilizando expressões regulares aplicadas às matrizes de peso.

Já no software de visualização as principais implementações serão: o cálculo de p-valor para matrizes de peso e posição [34], pois é considerado mais confiável que a utilização do score [35]; abertura de dados externos e salvamentos dos dados gerados nos formatos mais comuns, como GFF, BED, SGA, FPS, CSV e tabular; e a criação de um algoritmo de encontro de padrões e busca de expressões regulares de matrizes de posição.

Na questão experimental, algumas das possibilidades sugeridas nas Conclusões e Discussões poderiam ser testadas sem muitas dificuldades, dentre elas: diminuição do tempo de indução; o teste das células em outras condições de ativação dos FTs, além da mudança de pH; a troca do core atual por um core mais fraco; e o teste de elementos cis-regulatórios de score alto para uma única matriz de peso e posição.

8. BIBLIOGRAFIA

1. Silva-Rocha R, de Lorenzo V. Mining logic gates in prokaryotic transcriptional regulation networks. *FEBS Lett*. 2008. doi:10.1016/j.febslet.2008.01.060
2. Silva-Rocha R, Castro LDS, Antoniêto ACC, Guazzaroni ME, Persinoti GF, Silva RN. Deciphering the cis-regulatory elements for XYR1 and CRE1 regulators in *Trichoderma reesei*. *PLoS One*. 2014. doi:10.1371/journal.pone.0099366
3. Amores GR, Guazzaroni ME, Silva-Rocha R. Engineering Synthetic cis-Regulatory Elements for Simultaneous Recognition of Three Transcriptional Factors in Bacteria. *ACS Synth Biol*. 2015. doi:10.1021/acssynbio.5b00098
4. Guazzaroni ME, Silva-Rocha R. Expanding the Logic of Bacterial Promoters Using Engineered Overlapping Operators for Global Regulators. *ACS Synth Biol*. 2014. doi:10.1021/sb500084f
5. Levskaya A, Chevalier AA, Tabor JJ, et al. Synthetic biology: engineering *Escherichia coli* to see light. *Nature*. 2005.
6. Fernandez-Rodriguez J, Moser F, Song M, Voigt CA. Engineering RGB color vision into *Escherichia coli*. *Nat Chem Biol*. 2017. doi:10.1038/nchembio.2390
7. Tamsir A, Tabor JJ, Voigt CA. Robust multicellular computing using genetically encoded NOR gates and chemical “wiresg.” *Nature*. 2011. doi:10.1038/nature09565
8. Reece JB, Urry LA, Cain ML, Wasserman SA, Minorsky P V., Jackson RB. *Campbell Biology*.; 2010. doi:10.1007/s13398-014-0173-7.2
9. Vellai T, Vida G. The origin of eukaryotes: The difference between prokaryotic and eukaryotic cells. *Proc R Soc B Biol Sci*. 1999. doi:10.1098/rspb.1999.0817
10. Chen Y, Sheng J, Jiang T, Stevens J, Feng X, Wei N. Transcriptional profiling reveals molecular basis and novel genetic targets for improved resistance to multiple fermentation inhibitors in *Saccharomyces cerevisiae*. *Biotechnol Biofuels*. 2016. doi:10.1186/s13068-015-0418-5
11. Godinho CP, Mira NP, Cabrito TR, et al. Yeast response and tolerance to benzoic acid involves the Gcn4- and Stp1-regulated multidrug/multixenobiotic resistance transporter Tpo1. *Appl Microbiol Biotechnol*. 2017. doi:10.1007/s00253-017-8277-6
12. Purves M, Sadava J, Orians L, Heller T. Life: The Science of Biology 7th Edition. *McGraw Hill*. 2004.
13. Gilbert SF. Developmental Biology. Osteogenesis: The Development of Bones. *Dev Biol* 6th Ed. 2000. doi:10. 0-87893-243-7
14. Wray GA. The evolutionary significance of cis-regulatory mutations. *Nat Rev Genet*. 2007. doi:10.1038/nrg2063
15. Reményi A, Schöler HR, Wilmanns M. Combinatorial control of gene expression. *Nat Struct Mol Biol*. 2004. doi:10.1038/nsmb820
16. Tanenbaum AS, Austin T, Cossio MLT, et al. *Structured Computer Organization (6th Edition)*.; 2012. doi:10.1007/s13398-014-0173-7.2
17. Staden R. Methods for calculating the probabilities of finding patterns in sequences. *Bioinformatics*. 1989. doi:10.1093/bioinformatics/5.2.89
18. Eiben AE, Smith J. From evolutionary computation to the evolution of things. *Nature*. 2015. doi:10.1038/nature14544

19. Bäck T, Hammel U, Schwefel HP. Evolutionary computation: Comments on the history and current state. In: *Evolutionary Computation: The Fossil Record.* ; 1998. doi:10.1109/9780470544600.ch1
20. Coello Coello CA. A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. *Knowl Inf Syst.* 1999. doi:10.1007/bf03325101
21. Glez-Peña D, Lourenço A, López-Fernández H, Reboiro-Jato M, Fdez-Riverola F. Web scraping technologies in an API world. *Brief Bioinform.* 2013. doi:10.1093/bib/bbt026
22. Lam SK, Pitrou A, Seibert S. Numba: A LLVM-based python JIT compiler. In: *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC - LLVM '15.* ; 2015. doi:10.1145/2833157.2833162
23. De Boer CG, Hughes TR. YeTFaSCo: A database of evaluated yeast transcription factor sequence specificities. *Nucleic Acids Res.* 2012. doi:10.1093/nar/gkr993
24. Yamanishi M, Katahira S, Matsuyama T. TPS1 terminator increases mRNA and protein yield in a *saccharomyces cerevisiae* expression system. *Biosci Biotechnol Biochem.* 2011. doi:10.1271/bbb.110246
25. Colot H V., Park G, Turner GE, et al. A high-throughput gene knockout procedure for *Neurospora* reveals functions for multiple transcription factors. *Proc Natl Acad Sci U S A.* 2006. doi:10.1073/pnas.0601456103
26. Redden H, Alper HS. The development and characterization of synthetic minimal yeast promoters. *Nat Commun.* 2015. doi:10.1038/ncomms8810
27. Cox RS, Surette MG, Elowitz MB. Programming gene expression with combinatorial promoters. *Mol Syst Biol.* 2007. doi:10.1038/msb4100187
28. Kinkhabwala A, Guet CC. Uncovering cis regulatory codes using synthetic promoter shuffling. *PLoS One.* 2008. doi:10.1371/journal.pone.0002030
29. Gietz RD. Yeast transformation by the LiAc/SS carrier DNA/PEG method. *Methods Mol Biol.* 2014. doi:10.1007/978-1-4939-0799-1_4
30. Hunter JD. Matplotlib: A 2D graphics environment. *Comput Sci Eng.* 2007. doi:10.1109/MCSE.2007.55
31. Cherry JM, Hong EL, Amundsen C, et al. *Saccharomyces* Genome Database: The genomics resource of budding yeast. *Nucleic Acids Res.* 2012. doi:10.1093/nar/gkr1029
32. Gomes-Silva A, Silva-Rocha R. TFound: A Software to Map Cis-Regulatory Elements in Yeast. *bioRxiv.* January 2018:474957. doi:10.1101/474957
33. Mostovoy Y, Thiemicke A, Hsu TY, Brem RB. The role of transcription factors at antisense-expressing gene pairs in yeast. *Genome Biol Evol.* 2016. doi:10.1093/gbe/evw104
34. Touzet H, Varré JS. Efficient and accurate P-value computation for position weight matrices. *Algorithms Mol Biol.* 2007. doi:10.1186/1748-7188-2-15
35. Erill I, O'Neill MC. A reexamination of information theory-based methods for DNA-binding site identification. *BMC Bioinformatics.* 2009. doi:10.1186/1471-2105-10-57