PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM TECNOLOGIA EM SAÚDE

JOSÉ ROBERTO GORSKI

UM ALGORITMO GENÉTICO PARA LOCALIZAÇÃO DE "*MOTIFS*" REGULATÓRIOS EM GENOMAS DE PROCARIONTES

CURITIBA
2007

JOSÉ ROBERTO GORSKI

UM ALGORITMO GENÉTICO PARA LOCALIZAÇÃO DE "*MOTIFS*" REGULATÓRIOS EM GENOMAS DE PROCARIONTES

Dissertação de mestrado apresentado ao Curso de Pós-Graduação em Tecnologia em Saúde, da Pontifícia Universidade Católica do Paraná, como pré-requisito à obtenção do título de Mestre.

Orientador: Prof. Dr. Humberto Maciel França Madeira.

CURITIBA
2007

JOSÉ ROBERTO GORSKI

UM ALGORITMO GENÉTICO PARA LOCALIZAÇÃO DE "*MOTIFS*" REGULATÓRIOS EM GENOMAS DE PROCARIONTES

Dissertação de mestrado apresentado ao Curso de Pós-Graduação em Tecnologia em Saúde, da Pontifícia Universidade Católica do Paraná, como pré-requisito à obtenção do título de Mestre.

COMISSÃO EXAMINADORA

_____

Prof. Dr. Júlio Cesar Nievola
Pontifícia Universidade Católica do Paraná

_____

Prof. Dr. Leandro dos Santos Coelho
Pontifícia Universidade Católica do Paraná

_____

Prof. Dr. Leonardo Magalhães Cruz
Universidade Federal do Paraná

Curitiba, 14 de Novembro de 2007.

À minha esposa Débora, por sua paciência, dedicação e compreensão
por todos estes meses de trabalho.

AGRADECIMENTOS

Aos meus pais, que sempre me apoiaram e me incentivaram em todas as etapas de minha vida.

Ao meu orientador que me auxiliou em todas as etapas deste trabalho.

Aos professores, pela paciência e disponibilidade de tempo nos esclarecimentos das dúvidas.

Aos meus colegas de sala de aula que colaboraram com este estudo.

Aos meus familiares e amigos pela compreensão em todos estes meses de que estive dedicado a esta atividade.

RESUMO

O presente trabalho propõe um algoritmo para predição de seqüências de regiões regulatórias em genomas bacterianos, utilizando uma técnica pouco empregada para esta finalidade, como alternativa às ferramentas de predição de regiões regulatórias já desenvolvidas. Foi desenvolvido um programa computacional baseando-se no uso de um Algoritmo Genético, batizado como GA_FIND_RR (Genetic Algorithm Find Regulatory Regions), cuja função de adaptação foi concentrada nas duas principais características conhecidas para estas regiões: a primeira característica é a "divisão" destas regiões em duas partes distintas, simbolicamente definida como $W_1N_xW_2$, onde W_1 e W_2 são oligômeros separados por uma distância variando entre 1 e 30 bases (N_x). O algoritmo foi testado em uma base de teste e com a bactéria *Bacillus subtilis*. Foram desenvolvidas cinco versões distintas do GA_FIND_RR, onde houve variações na função de adaptação e na forma como os dados eram pesquisados na matriz de busca. Em base de testes, o algoritmo desenvolvido foi capaz de localizar as regiões regulatórias artificialmente implantadas. Para o *Bacillus subtilis*, o algoritmo proposto conseguiu predizer cerca de 20% das regiões regulatórias descritas na literatura, sendo 83 "*motifis*" promotores, 35 "*motifis*" repressores e 6 "*motifis*" ativadores. Os testes foram executados usando 100 bases "*upstream*". Os algoritmos mais conhecidos e usados para esta finalidade possuem um índice de acerto em torno de 15% a 25% para 400 bases "*upstream*". Ou seja, pode-se afirmar que a utilização do Algoritmo Genético é mais uma alternativa as técnicas já usadas, ou servir de complemento a elas. Acredita-se que o resultado possa ser melhorado, em versões futuras, quando houver um conhecimento biológico aprofundado a respeito das regiões regulatórias e com a inclusão de técnicas de adaptação dinâmicas no algoritmo.

Palavras-chave: Algoritmo Genético, Bioinformática, Regiões Regulatórias.

ABSTRACT

The present work aimed at developing an algorithm for the prediction of regulatory sequences in bacterial genomes using a technique that has nor been widely used for this purpose, as an alternative to other available tools. A computer software based on a genetic algorithm was developed and named GA_FIND_RR, with a fitness function based on the two main features of such sequences: a region that split into two distinct parts, symbolically defined as $W_1N_xW_2$, W1 and W2 being oligomers separated by other bases ranging from 0 to 30 (Nx). The algorithm was tested against a test dataset and against a *Bacillus subtilis* dataset. Five different versions of GA_FIND_RR were developed with variations in the fitness function and in the way searches were made against the target dataset. For the test dataset, the algorithm was able to find the artificially implanted regulatory sequences. For the *B. subtilis* dataset, when the first 100 bases upstream of the transcription start site was used, the algorithm was able to predict 20% of the regulatory regions described in the literature, with 83 promoters, 35 repressors and 06 activators. Currently, widely used algorithms for this purpose possess a accuracy ranging from 15 to 25%, when the first 400 upstream bases are used. These results suggest that the proposed algorithm is a viable alternative to current techniques or can be a useful complement to them. Future improvements in the efficacy of algorithm can be envisioned, as a more thorough knowledge of regulatory regions are unveiled as well as by adding dynamic adaptation techniques to the algorithm.

Keywords: Algorithm Genetic, Bioinformatic, Regulatory Regions.

LISTA DE ABREVIATURAS

AG: Algoritmo Genético;

CZ: Tipo de Cruzamento;

DNA: Ácido desoxirribonucléico;

EL: Elitismo;

GA_FIND_RR: *Genetic Algorithm Find Regulatory Regions;*

GR: Gerações;

HMM: *Hidden Markov Model*;

PP: População;

RBS: *Ribossome Binding Site;*

RNA: ácido ribonucleico

RR: Região Regulatório;

TI: Tamanho do Indivíduo;

TM: Taxa de Mutação;

TS: Tipo de Seleção;

TSS: Transcription Start Site;

TX: Taxa de Seleção.

ÍNDICE DE FIGURA

ÍNDICE DE TABELA

SUMÁRIO

1. INTRODUÇÃO

A localização de regiões regulatórias (RR) em genomas de eucariontes e procariontes é um desafio computacional para a biologia molecular. Existem vários métodos computacionais atualmente usados para a predição destas regiões e entre eles, destacam-se: Matrizes de peso, Modelo Oculto de Markov e métodos de busca exaustiva. Algumas destas ferramentas que utilizam estes métodos foram desenvolvidas com o objetivo de poderem ser usadas para mais de um organismo e outras ferramentas destinam-se a localização de regiões regulatórias para apenas um organismo. Os algoritmos desenvolvidos, objetivando a localização de RR para mais de um organismo, na média, conseguem predizer, 15% a 25% de RR, ou seja, têm um fraco desempenho. Algoritmos desenvolvidos para um organismo específico, tendem a localizar um número maior de RR em relação aos algoritmos genéricos.

Os estudos indicam que diferentes algoritmos trabalhando em conjunto, são complementares entre si. Ou seja, RR preditas por um determinado algoritmo “A”, podem não ser encontradas por um outro algoritmo “B”, e RR preditas pelo algoritmo “B” podem não ser encontradas pelo algoritmo “A”. Sendo assim, a soma de todas as RR preditas por ambos os algoritmos tende ser maior que a execução de apenas um algoritmo.

A utilização de algoritmos de otimização global, que empregam uma estratégia de busca voltada em direção a pontos de "alta aptidão", podem ser usados como complemento às ferramentas já usadas na predição de RR. Estas ferramentas não visam encontrar todas as soluções para um determinado problema, porém, procuram boas soluções para este problema. O algoritmo genético (AG) é uma destas ferramentas que podem ser usadas para buscar boas soluções dentro de um determinado espaço de busca. Sendo assim, o AG pode ser usado como uma alternativa complementar as técnicas atuais de predição de RR.

Nesta dissertação de mestrado foi desenvolvida uma ferramenta computacional, baseada em AG, com a intenção de predizer os principais *motifs* regulatórios de genomas de procariontes. Com o objetivo de inferir uma melhor função de adaptação, foram desenvolvidas versões distintas do algoritmo, batizado de GA_FIND_RR. Cada função de adaptação foi escrita em um código fonte distinto, em ambiente Matlab, variando principalmente a forma de busca na base de dados e o cálculo do *fitmess* dos indivíduos candidatos a serem uma RR.

1.1 Objetivo Geral

Desenvolver um algoritmo para predição de *motif* regulatórios, baseando-se em uma técnica de computação evolucionária, denominada como Algoritmo Genético.

1.2 Objetivos específicos

- Propor um algoritmo para localizar regiões regulatórias de DNA(Ácido desoxirribonucléico**)** de procariontes, com a utilização de Algoritmo Genético, criando mais de uma versão para testar variações da função de aptidão;

- Testar o algoritmo desenvolvido, no ambiente Matlab, em um ambiente de testes;

- Validar o algoritmo desenvolvido para um organismo procarionte e comparar os resultados obtidos com os já publicados para este mesmo organismo.

1.3 Estrutura da Dissertação

Esta dissertação esta organizada em quatro capítulos. No capítulo 2 apresentam-se noções básicas de biologia molecular, bioinformática e ferramentas para localizações de RR. Também são apresentados os principais conceitos do algoritmo genético. No capítulo 3 é descrito como foi desenvolvido o GA_FIND_RR. No capítulo 4 é apresentando os resultados obtidos com o GA_FIND_RR para uma base de testes e para o *Bacillus subtilis*, bem como a discussão e análise dos resultados. No capítulo 5 são expostas as conclusões do trabalho e sugeridas propostas de melhorias para o algoritmo desenvolvido.

2 Revisão bibliográfica

2.1 Noções de Biologia Molecular

O universo biológico consiste de dois tipos de células, as eucariontes e procariontes. As eucariontes possuem um núcleo celular bem definido ao passo que os procariontes não possuem. As células eucariontes estão presentes em organismos multicelulares (mamíferos, plantas, etc.) e no reino protista. As células procariontes compõem a maioria dos organismos unicelulares, como as bactérias.

Como um organismo vivo, as células podem crescer, se reproduzir, processar informações, responder a estímulos e processar uma surpreendente quantidade de reações químicas. A estas habilidades define-se como vida (HARVEY et al., 2003). Segundo Hunter (1993), os sistemas vivos processam matéria, energia e informação. O princípio básico da vida, a reprodução, é a transferência dos materiais encontrados em um organismo para um outro organismo, que por sua vez mantem as características similares ao seu progenitor, possuindo uma capacidade de adaptação às circunstâncias em mudança. Porém, alguns aspectos dos organismos vivos permaneceram o mesmo apesar de quase 4 bilhões de anos de evolução. Os conteúdos moleculares básicos para processar a matéria, energia e informação mudaram pouco neste período.

2.1.1 DNA e RNA

Com o avanço da biologia molecular, tem sido possível determinar a seqüência completa do DNA de diferentes organismos, conforme é demonstrado no Projeto Genoma Humano, além de dezenas de genomas de procariontes já concluídos (BENSON et al., 2004).

A estrutura do DNA foi elucidada em 1953 por James Watson e Francis Crick, abrindo caminho para compreensão da ação gênica e da hereditariedade em termos moleculares (GRIFFITHS, 2002). O DNA contém todas as informações necessárias para a construção das células e tecidos de um organismo. A replicação exata destas informações assegura um desenvolvimento normal de um organismo de geração para geração. A informação armazenada no DNA é organizada em unidades hereditárias conhecidas como genes que identificam as características de um organismo (HARVEY et al., 2003). O DNA é uma molécula que carrega todas as informações para a codificação das proteínas necessárias para

um determinado organismo. Esses genes estão presentes nos cromossomos, que são estruturas compostas de DNA e de outras proteínas, que estão em todas as células do corpo, entre as suas funções, pode-se destacar: carregar a informação genética para as células poderem se reproduzir e possuir as informações necessárias para a produção de proteínas (BROWN, 2002). O conjunto completo do material genético (todos os cromossomos) é chamado de Genoma. Em procariontes, em geral, existe apenas um cromossomo na célula (GRIFFITHS, 2002).

A molécula de DNA é formada por uma dupla fita de nucleotídeos. Cada nucleotídeo contém fosfato, pentose – no caso do DNA, desoxirribose – e uma das quatro bases nitrogenadas: Adenina (A), Timina (T), Guanina (G) ou Citosina (C). Os nucleotídeos são ligados entre si por ligações fosfodiéster e as bases nitrogenadas são ligadas através de pontes de hidrogênio entre uma fita e outra, conforme apresentado na figura 1 (GRIFFITHS, 2002). O fluxo unidirecional da informação contida no DNA até a síntese protéica é conhecido como "dogma central da biologia molecular" (CRICK, 1970), como mostrado na figura 2.

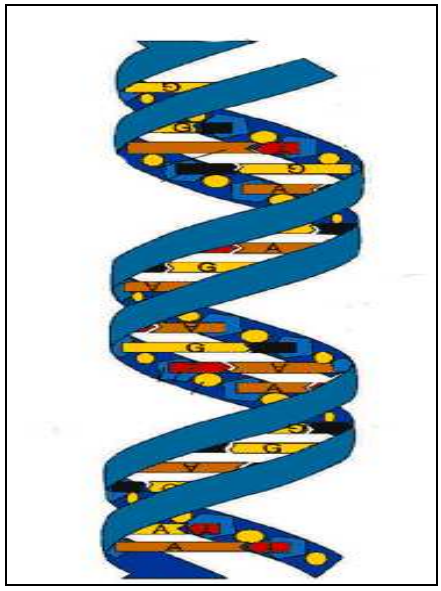

Figura 1 - DNA visto em três dimensões e esquema de ligações químicas. Duas cadeias de nucleotídeos orientam-se em direções opostas. Entre as bases ocorre o pareamento (A com T e C com G). As duas cadeias estão enroladas formando uma dupla hélice. Disponível em <http://www.mundovestibular.com.br/materias/imagens/DNA2.gif>. Acesso em 07/07/2007.

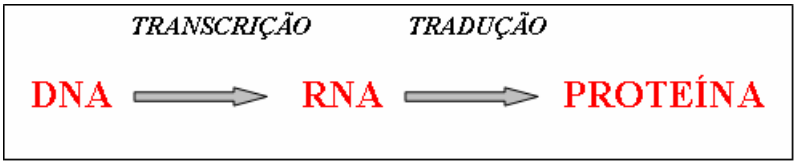

Figura 2 - Dogma Central da Biologia Molecular proposto por Crick (1970).
Através de uma molécula de DNA é gerada uma molécula de RNA que por sua vez codifica uma proteína. Disponível em <http://www.biotecnologia.com.br/revista/bio29/bioinfo2.jpg >. Acesso em 07/07/2007

As bases nitrogenadas unem-se em pares específicos – Adenina (A) liga-se com a Timina (T) e a Citosina (C) liga-se com a Guanina (G). A combinação dessas letras é a base do código genético, sendo a molécula de DNA uma matriz ou padrão para a produção das moléculas protéicas (HUNTER, 1993). A informação contida no DNA precisa ser transcrita em moléculas de RNA (ácido ribonucléico), síntese esta catalisada pela enzima RNA polimerase. As duas cadeias que compõem o DNA se separam e apenas uma delas orienta a formação de uma cadeia de RNA, para a qual é transcrita a informação codificada no gene. No RNA não há timina (T), mas em seu lugar encontra-se Uracila (U). A maioria dos genes transcreve suas informações para o mRNA (RNA mensageiro). Este comanda a síntese de proteínas. O mRNA contém sua informação disposta em trinca de bases, os códons (por exemplo, CTG). Cada códon corresponde a um aminoácido, e a seqüência de códons determina a seqüência de aminoácidos. O conjunto de aminoácidos forma a proteína. Existem 64 códons diferentes, correspondentes a 20 aminoácidos. Foi verificado que determinados aminoácidos podem ser codificados por dois ou mais códons diferentes. Cada códon, no entanto, codifica sempre o mesmo aminoácido, e certos códons servem como pontos iniciais e finais dos genes ou códon de início e terminação (GRIFFITHS, 2002).

As moléculas de DNA e RNA são quimicamente semelhantes, porém com tamanhos bem distintos. Enquanto a molécula de DNA pode conter milhões de nucleotídeos, a molécula de RNA contém de centenas a milhares de nucleotídeos (HARVEY et al., 2003).

Segundo Huerta e Collado-Vides (2003), nos procariontes, os genes que codificam proteínas ficam em locais próximos entre si, podendo formar "*operons*", que são transcritos simultaneamente gerando um mesmo mRNA para todos eles. Na maioria dos procariontes a transcrição é controlada por dois elementos da seqüência do DNA, que estão aproximadamente -35 bases e -10 bases, respectivamente, do início do local da transcrição, a base cujo número é 1 é a primeira base transcrita. Estes dois elementos da seqüência são

denominados seqüências do promotor, porque promovem o reconhecimento do local onde se inicia a transcrição pelo RNA polimerase. A seqüência de consenso para a posição -35 é “TTGACA”, e para a -10 é “TATAAT”. (a posição -10 é também conhecida como “Pribnow-box”). Estas seqüências (oligômeros) foram extraídas baseando-se no genoma da bactéria *Escherichia coli*. Um estudo da distribuição da região promotora desta bactéria em relação às seqüências e posições de consenso pode ser encontrado no trabalho realizado por Sivaraman et al. (2005).

A região promotora é uma área do cromossomo que determina onde a transcrição de um gene ou grupo de genes (operons) inicia e em que condições se dará este processo (GORDON et al., 2003). A atividade da RNA polimerase em um promotor, é regulada pela interação com proteínas acessórias, que afetam sua habilidade de reconhecer locais de inicio de transcrição. Estas proteínas regulatórias podem agir positivamente (ativadores) ou negativamente (repressores), desta forma, regulando a produção de proteínas (HARVEY et al., 2003). A figura 3 tem uma representação do processo de expressão gênica e a figura 4 ilustra um exemplo hipotético de uma RR em um gene de procarionte.

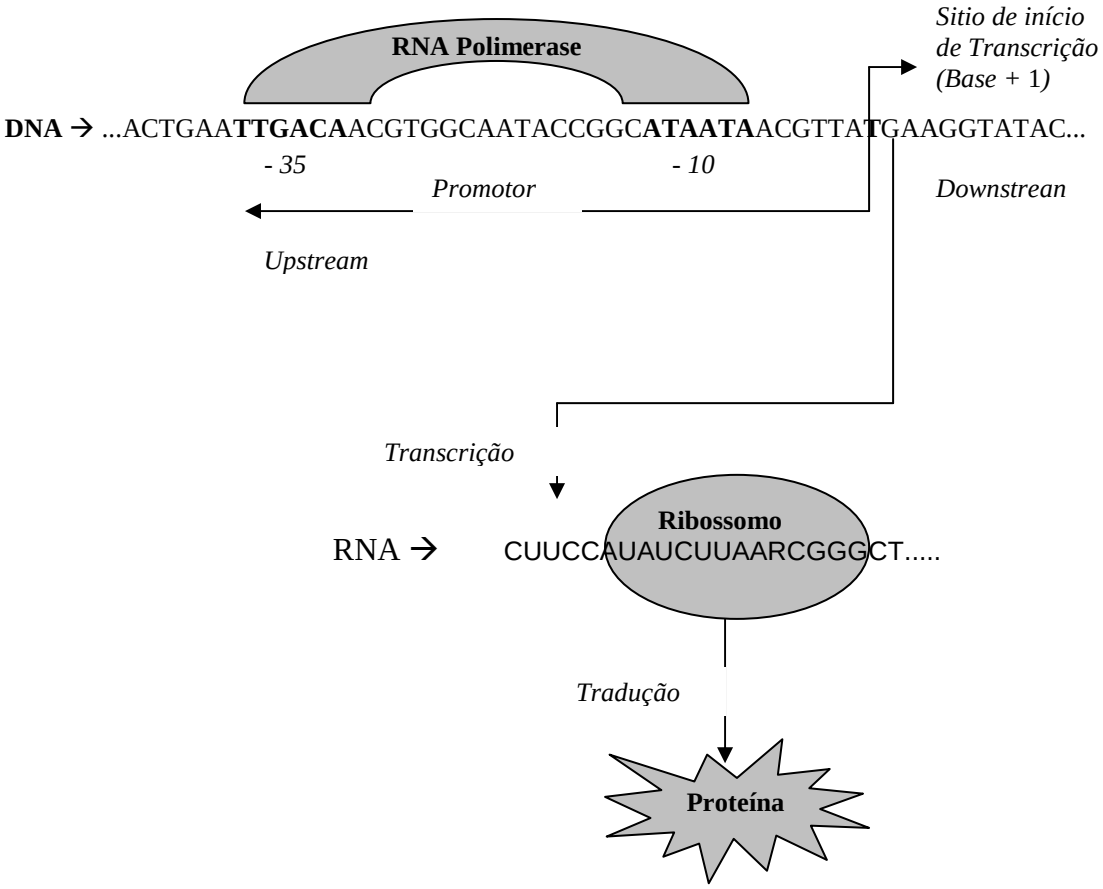

Figura 3 - Exemplo esquemático de uma proteína em procarionte.
O processo inicia-se com a ligação química da molécula RNA Polimerase com a região promotora (em negrito), que geralmente fica antes do sitio de transcrição, gerando uma molécula de RNA que por sua vez ira produzir uma proteína, com o auxilio de um aglomerado macromolecular conhecido como Ribossomo.

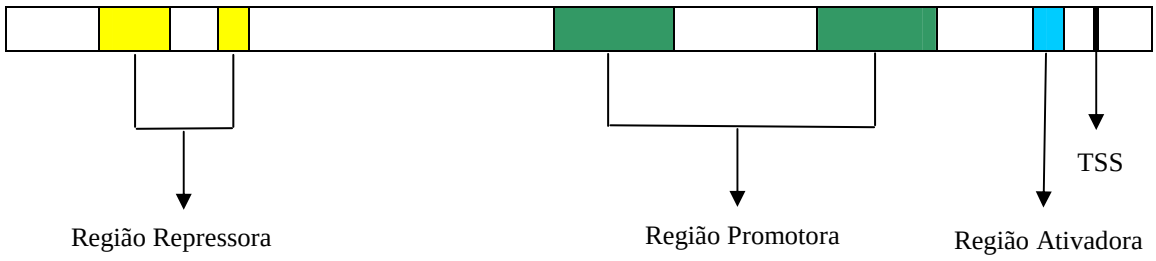

Figura 4 - Representação esquemática de uma RR de um gene de procarionte.
As regiões em amarelo representam à região repressora, as regiões em verde representam as regiões promotoras e a região em azul representa a região ativadora. TSS é o inicio do sítio de transcrição.

2.2 Bioinformática

A bioinformática é uma área de estudo com características multidisciplinares, abrangendo diversos ramos da ciência, destacando-se a estatística, matemática, física, ciência da computação e naturalmente a biologia molecular (BARNES e GRAY, 2003). O aprofundamento dos estudos referentes à bioinformática está associado diretamente ao início do projeto genoma humano, devido à necessidade de criar ferramentas computacionais que auxiliassem a manipulação dos dados gerados pelo projeto (BURLEY et al., 1999).

Segundo Finkelstein et al. (2004), a bioinformática é responsável por uma revolução na biologia molecular, devido ao profundo conhecimento adquirido das seqüências de DNA, das sínteses de RNA e da geração de proteínas, gerando um vasto conjunto de dados. Para a manipulação destes dados, foram necessários esforços significativos de cientistas da computação na criação de ferramentas computacionais, usando técnicas de mineração de dados, sistemas inteligentes, ferramentas de busca, ferramentas de comparação, entre outros.

Os dados e informações produzidos pelas ferramentas de bioinformática geralmente ficam armazenados em Bancos de Dados públicos, tais como o Genbank (2007), construído e distribuído pelo *National Center for Biotechnology Information* (NCBI) (BENSON et al., 2004) e o COG (2007) (TATUSOV et al., 2003), mantido pelo mesmo centro. O EMBL (2007), mantido pelo *European Bioinformatics Institute* (EBI) (KANZ et al., 2005) e o DDBJ, mantido pelo *Center for Information Biology and DNA Data Bank of Japan*, (MIYAZAKI et al., 2004), são bancos de dados que armazenam informações de diversos tipos de organismos, tanto de eucariontes como de procariontes.

Existem Bancos de dados mais específicos, como é o caso do *Comprehensive Microbial Resource* (CMR, 2007), mantido pelo *The Institute for Genomic Research* (TIGR), que é uma base de dados exclusiva de organismos unicelulares (PETERSON et al., 2001), e o RegTransBase (KAZAKOV et al., 2007) que é um banco de dados exclusivo das seqüências regulatórias de organismos procariontes, mantido por vários centros de pesquisa, entre eles: *Howard Hughes Medical Instutuite, Russian Academy of Sciences,* entre outros.

Com a grande quantidade de dados armazenados, conforme observado na figura 5, tornou-se importante fazer comparações entre as seqüências armazenadas, com o objetivo de inferir funções e relacionamento evolucionário entre organismos. Os programas de

comparação de seqüências são as ferramentas computacionais mais utilizadas na bioinformática. O programa mais conhecido entre eles é o BLAST (*Basic Local Alignment Search Tool*) (ALTSCHUL et al., 1997). O seu uso é de domínio público, podendo ser acessado diretamente pela internet (BLAST, 2007). Outro exemplo de ferramenta de comparação de seqüências, também de domínio público é o FASTA (PEARSON e LIPMAN, 1988), também podendo ser acessado diretamente pela internet (FASTA, 2007).

Crescimento do *GenBank* (1982 – 2005)

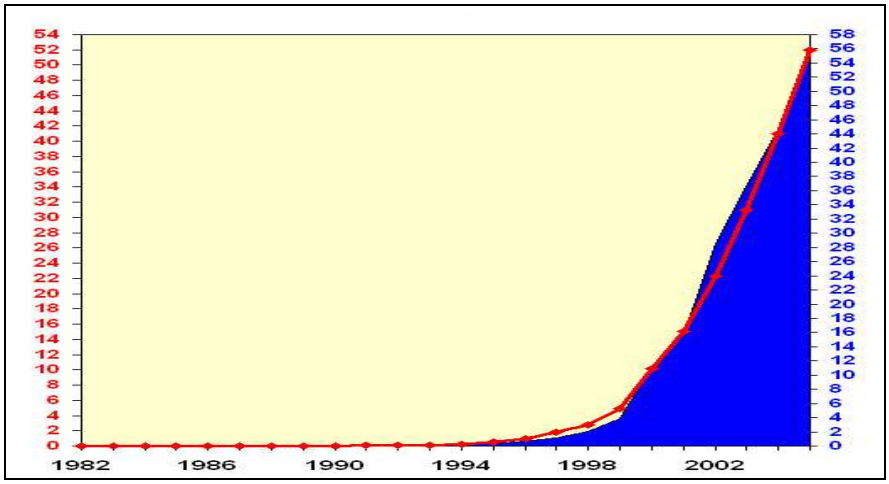

Figura 5 - Crescimento da base de dados do GenBank de 1982 até 2005. A cor azul representa a quantidade de pares de bases armazenadas e a linha vermelha a quantidades de seqüências armazenadas. Disponibilizado em <http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html>. Acesso em 28/04/07.

Atualmente existem várias ferramentas desenvolvidas para aplicações em bioinformática, muitas destas ferramentas estão disponíveis para uso público em vários portais na internet. Alguns destes portais estão descritos na tabela 1.

A Bioinformática não seria possível sem os avanços dos recursos computacionais em hardware, *software* e na utilização da internet. Estes recursos auxiliam no armazenamento, consultas e análises de uma grande quantidade de dados e informações disponibilizadas em grandes bancos de dados.

Com os atuais avanços tecnológicos, pode-se prever um futuro de grandes descobertas na área de biologia molecular que poderão auxiliar toda a humanidade com o emprego de ferramentas produzidas pela bioinformática (LESK, 2002).

Tabela 1 - Portais na internet com ferramentas computacionais para bioinformática.

Nome	Principais ferramentas	Site na Internet
NCBI	Ferramentas para análises de seqüências de nucleotídeos, análises de proteínas, visualização de estruturas, ferramentas para analise de genomas e análises da expressão gênica.	http://www.ncbi.nlm.nih.gov/Tools/
EMBNet	Conjunto de ferramentas para busca de seqüências, alinhamento de seqüências, analise estatística de seqüências de proteínas e base de dados de promotores de eucariontes.	http://www.ch.embnet.org/
ExPASy	Site dedicado a sistemas de analises de proteínas	http://expasy.org/tools/
BRC	Portal contendo diversas tabelas, com os *links* e descrições resumidas de várias ferramentas de bioinformática.	http://www.brc.dcs.gla.ac.uk/~mallika/bioinformatics-tools.html
CBS	Contem uma série de ferramentas, tais como, analise de DNA, reconhecimento de seqüências de proteínas, métodos de predição de seqüências de proteínas, anotação de seqüências, entre outros. Porém o acesso gratuito destas ferramentas e exclusiva para acadêmicos.	http://www.cbs.dtu.dk/biotools/
BioWiki	Contem uma série de ferramentas para analise de seqüências, localização de genes e promotores, análises de proteínas, Links para Banco de dados, analise de Microarrays e links para várias companhias e comunidades ligadas a bioinformática.	http://www.biodirectory.com/biowiki/Main_Page

2.3 Localização de Regiões Regulatórias

Um dos maiores desafios da biologia molecular é a compreensão dos mecanismos que regulam a expressão dos genes. Uma importante etapa neste desafio é a habilidade em identificar elementos regulatórios, que situam-se, tipicamente, antes do início da área de transcrição de um gene (chamada de região "*upstream*"), cuja função é ativar ou inibir o mecanismo de transcrição. Estas regiões podem ser chamadas de regiões regulatórias. A predição dos elementos regulatórios é um problema onde métodos computacionais oferecem grande esperança, e os "bioinformatas" têm investido um considerável esforço para a solução deste problema (TOMPA et al., 2005).

Para localizar regiões regulatórias em organismos procariontes, devem ser levadas em conta as características destas regiões, explicadas por Li et al. (2002) e Huerta e Collado-Vides (2003), onde pode-se representar estas seqüências por W1NXW2 (conhecido como dimer), onde, W1 e W2 são oligômeros (conjunto de bases) e NX é uma quantidade arbitrária

de bases separando W1 e W2, podendo variar entre 0 (zero) e 30 (trinta) bases, conforme o esquema da figura 6.

..... xxxxxxxxxxxPPPPPPxxxxxxxxxxPPPPPPxxxxxxxxxxxxxxxGGGGGGGGG.....

Figura 6 - Característica de uma região regulatória em procarionte.
A letra "P" é a indicação de um Promotor, a letra "G" é a indicação da região transcrita do gene e o "x" é uma seqüência de bases arbitrárias neste processo. Os símbolos "P", "G" e "x" , são representações de qualquer umas das bases nitrogenadas (A, C, G e T).

Todos os programas analisados partem do princípio que os oligômeros candidatos a serem regiões regulatórias aparecem várias vezes em uma seqüência de DNA, sendo estatisticamente significantes, ou seja, são "super-representadas" ("*over-represented"*).

Existem alguns programas de computadores já desenvolvidos para a localização de regiões regulatórias em procariontes. Entre as estratégias usadas, destacam-se os métodos que usam os conceitos de Matrizes de Peso, Busca Exaustiva e modelos estatísticos como o Modelo Oculto de Markov ou *Gibbs Sampling*.

2.3.1 Busca Exaustiva e Matriz de Peso

Busca exaustiva é um algoritmo de busca que procura encontrar uma solução para um determinado problema, testando todas as possibilidades. Também pode ser chamado de "força bruta" (BOCKHOLT, 2004). É uma técnica eficiente, mas dependendo do tamanho da área de busca a ser efetuada, pode acarretar em um alto custo para achar os melhores resultados, devido ao grande número de combinações possíveis de uma determinada seqüência (HAUPTY E HAUPTY, 2004).

Um exemplo de aplicação usando Busca Exaustiva com matriz de peso para predição de RR, é o algoritmo sugerido por Li et al. (2002) que consiste basicamente em três passos, como seguem abaixo:

- O primeiro passo é tabular as posições de todas as "*strings*" "W" (*dimers*) (tipicamente com 5 bases para aproximadamente 1 MB de seqüência), da seqüência analisada. Após a criação da tabela, esta é pesquisada para contar o número de ocorrências dos "*dimers"* $W_1N_XW_2$, onde o espaço x, varia tipicamente entre 0 e 30 bases. O valor encontrado para cada "*dimer"* é comparado com um valor estatisticamente calculado de "super-representatividade".

- O segundo passo é obter os "*dimers*" estatisticamente significantes e agrupá-los, criando "*clusters*" de todos os "*dimers*" similares. Estes clusters são agrupamentos de oligômeros parecidos, como no exemplificado na figura 7;

CTGTAxxxxxxxxxxxxxxxxxxTACAGx
CTGTxxxxxxxxxxxxxxxxxxxxxxCAGT

Figura 7 - Oligômeros similares.
Podem ser regiões conservadas de RR de um mesmo fator, sendo agrupados. O "x" é uma seqüência de bases arbitrárias neste processo, podendo ser qualquer base nitrogenada (A, C, G, T).

- A etapa final examina as seqüências do genoma real, que são combinadas por todos os membros de um "*cluster*" com a região "*upstream*", sendo executado um alinhamento múltiplo das seqüências para criar uma PSWN ("*Position Score Weight Matrix*"), para procurar por regiões regulatórias prováveis.

Uma melhoria do algoritmo descrito por Li et al. (2002), foi desenvolvido por Mwangi e Siggia (2003) e Studholme et al. (2004).

Uma matriz de peso ("*Weight Matrix*") é definida como uma matriz de números $W_{i,\,x}$ onde *i* são as colunas {1, 2, 3, 4,....,n} e *x* são os nucleotídeos {A,C,G,T} para um DNA. O "*score*" de uma "*string*" $X_1 ... X_n$ é dada por: $W_{1,x1} + W_{2,x2} + ... + W_{n,xn}$. O exemplo a seguir (tabelas 2, 3, 4 , 5 , 6), ilustra o processo de criação de uma matriz de peso (ATTESON, 1998):

1) Tabula-se o número de ocorrências de cada nucleotídeo de cada posição, como por exemplo:

Tabela 2: Primeira etapa para montar a matriz de peso.

...	**-3**	**-2**	**-1**	**1**	**2**	**3**	**4**	**5**	**6**	**…**
...	C	G	G	G	T	A	A	G	T	...
...	A	A	G	G	T	A	T	G	C	...
...	C	A	G	G	T	G	A	G	G	...
...	T	G	G	G	T	A	A	C	T	...
...	C	A	A	G	T	A	A	G	C	...
...	A	A	G	G	T	A	G	G	C	...
...	A	T	G	G	T	G	A	G	T	...
...	T	T	G	G	T	A	A	G	G	...
...	A	A	G	G	T	A	T	T	T	...
...	A	A	G	G	T	A	A	G	G	...

Tabela 3: Segunda etapa para montar a matriz de peso.

Totais encontrados por coluna										
Nucleotídeos	**-3**	**-2**	**-1**	**1**	**2**	**3**	**4**	**5**	**6**	**Total**
A	5	6	1	0	0	8	7	0	0	**27**
C	3	0	0	0	0	0	0	1	3	**7**
G	0	2	9	10	0	2	1	8	3	**35**
T	2	2	0	0	10	0	2	1	4	**21**
Total	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**10**	

2) Divide cada entrada (incluindo os totais) pelo valor total da respectiva linha da seqüência, como mostrado na tabela abaixo:

Tabela 4: Terceira etapa para montar a matriz de peso.

Nucleotídeos	**-3**	**-2**	**-1**	**1**	**2**	**3**	**4**	**5**	**6**	**Total**
A	0.5	0.6	0.1	0.0	0.0	0.8	0.7	0.0	0.0	0.300
C	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.3	0.078
G	0.0	0.2	0.9	1.0	0.0	0.2	0.1	0.8	0.3	0.389
T	0.2	0.2	0.0	0.0	1.0	0.0	0.2	0.1	0.4	0.233

3) Divide cada entrada para normalizar a coluna total:

Tabela 5: Quarta etapa para montar a matriz de peso.

Nucleotídeos	**-3**	**-2**	**-1**	**1**	**2**	**3**	**4**	**5**	**6**
A	1.67	2.00	0.33	0.00	0.00	2.67	2.33	0.00	0.00
C	3.86	0.00	0.00	0.00	0.00	0.00	0.00	1.29	3.86
G	0.00	0.51	2.31	2.57	0.00	0.51	0.26	2.06	0.77
T	0.86	0.86	0.00	0.00	4.29	0.00	0.86	0.43	1.71

4) Finalmente, aplica-se o logaritmo neperiano, fazendo os devidos arredondamentos:

Tabela 6: Quinta etapa para montar a matriz de peso.

Nucleotídeos	**-3**	**-2**	**-1**	**1**	**2**	**3**	**4**	**5**	**6**
A	0,51	0,69	-1,10	-10,00	-10,00	0,98	0,85	-10,00	-10,00
C	1,35	-10,00	-10,00	-10,00	-10,00	-10,00	-10,00	0,25	1,35
G	-10,00	-0,66	0,84	0,94	-10,00	0,66	-1,36	0,72	-0,26
T	-0,15	-0,15	-10,00	-10,00	1,46	-10,00	-0,15	-0,85	0,54

Após a geração da matriz de peso para a seqüência analisada, os oligômeros são submetidos à matriz para o cálculo de sua representatividade, como no exemplo a seguir:

Seqüência: **CAGGTAAGC**, substituindo pelos valores, tem-se: 1,35 + 0,69 + 0,84 + 0,94 + 1,46 + 0,98 + 0,85 + 0,72 + 1,35 = **9, 18,** este é o valor para esta seqüência.

Se a soma de uma determinada seqüência da matriz de peso for superior a um limiar esperado, esta seqüência é considerada como “super-representada” e pode ser uma seqüência de uma região regulatória (KIBLER e HAMPSON, 2001a). Alguns exemplos de geração de matrizes de pesos podem ser encontrados em (KIBLER e HAMPSON, 2001b).

2.3.2 Hidden Markov Model (HMM)

O HMM é uma ferramenta para representação de distribuições de probabilidade para observações de grandes seqüências de dados, sendo constantemente usada em modelagem de dados seriais. O HMM é usado em quase todos os sistemas de reconhecimento de linguagem, em inúmeras aplicações computacionais para biologia molecular, em compressão de dados e em outras áreas de inteligência artificial para reconhecimento de padrões (GHAHRAMANI, 2001).

O HMM pode ser usado sempre que se quer modelar a probabilidade de uma seqüência linear de eventos. A figura 8 representa um diagrama de estados usado no HMM, cuja explicação é dado por Freitas (2002):

> *"Podem ser representados por um diagrama de estados, no qual cada estado é nomeado e transições possíveis entre estados são representadas por setas, identificadas com a probabilidade da transição. As probabilidades dos arcos que saem de cada estado devem somar 1. Dessa forma, o modelo de Markov pode ser visto como um autômato finito (não determinístico), com probabilidades junto a cada arco. Cada estado tem uma tabela com probabilidades de emissão de símbolos, e existem probabilidades de transição para mover de um estado para outro. O caminho através do modelo começa de um estado inicial, e o próximo passo é escolher um novo estado com uma probabilidade de transição (por exemplo, ficar no estado 1 com probabilidade de transição t1, 1 ou mover para o estado 2 com probabilidade t1, 2). Depois, é gerado um resíduo com uma probabilidade de emissão específica daquele estado (escolher um "a" com probabilidade p(a), por exemplo). Este processo é repetido até que o estado final seja alcançado. Como resultado, tem-se uma seqüência oculta de estados percorridos (a qual não é observada) e uma seqüência de símbolos (que é observada)."*

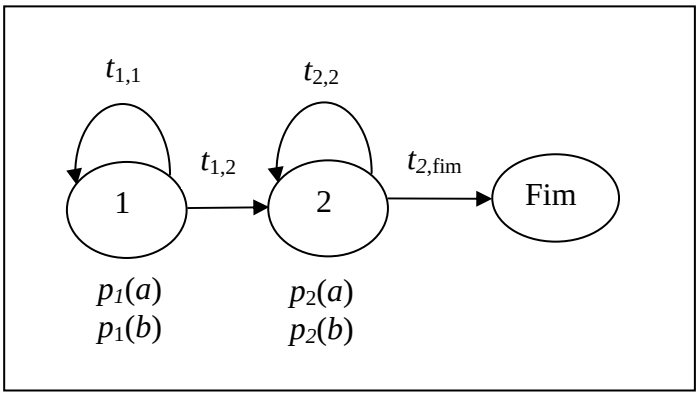

Figura 8 - Representação esquemática do HMM.

Para Eddy (2004) o HMM é um modelo probabilístico formal, que pode auxiliar na resolução de problemas de seqüências lineares. Ele provê o ferramental necessário para construir modelos complexos que podem resolver problemas de localização de genes, alinhamentos múltiplos, identificação de regiões regulatórias, entre outros.

Para a construção de um HMM são necessários quatro elementos: (i) um alfabeto com tamanho definido (por exemplo, A, C, G e T, neste caso o tamanho é quatro); (ii) o número de estados do modelo, estado pode ser definido como sendo a função que uma determinada posição em uma seqüência pode exercer (por exemplo: a posição esta em uma região intergene ou é um gene. Gene e intergene são estados); (iii) a probabilidade que cada base do alfabeto tem em pertencer a um determinado estado, onde a soma deve ser igual a 100% ou 1; (iv) a probabilidade que cada base em cada posição tem em alterar ou não de estado, onde a soma deve ser igual a 100% ou 1.

O HMM analisa a probabilidade de ocorrência de todas as possíveis configurações de seqüências para um determinado problema. No caso de busca por RR, pode ser criado um grupo de estados que caracterizem estas regiões intergênicas à procura de configurações de seqüências com taxas de probabilidade acima de um determinado limiar estabelecido. Um programa que faz uso do HMM para a localização de genes e regiões regulatórias é o GeneMarkS (BESEMER et al., 2001).

2.3.3 Gibbs Sampling

Gibbs Sampling é um modelo estatístico que faz uso de matrizes de peso para localizar as seqüências "super-representadas". A estratégia é usar seqüências longas, dividi-las em pequenos "pedaços" e transformá-las em matrizes de peso (TOMPA et al., 2005). O algoritmo de Gibbs não garante que a Matriz de Peso e o conjunto de ocorrências encontradas serão as melhores, mas converge para um máximo local (ao invés de um máximo global). Por outro lado, o método é rápido, o que o torna viável em várias aplicações.

Neuwald et al. (1995), explica uma técnica usando um algoritmo baseando em *Gibbs Sampling* para localizar regiões regulatórias, com a intenção de obter "*insight*" sobre estruturas de proteínas e predizer as suas funções. Segundo o autor, esta técnica é capaz de detectar as regiões regulatórias e otimizar o particionamento destas regiões.

2.3.4 Programas disponíveis para localização de regiões regulatórias

Existem várias ferramentas para predição de RR já desenvolvidas por várias instituições de pesquisa. As técnicas mais utilizadas são: Matrizes de Peso, *Gibbs Sampling* e Busca exaustiva. Na tabela 2 serão descritas, de forma resumida, algumas destas ferramentas.

Tabela 7 - Programas para predizer RR genéricos.
Resumo das principais características de ferramentas largamente usadas na predição de RR de DNAs.

Programa	Principal Operador	Técnica Utilizada	Referência
AlignACE	Usa um algoritmo baseado em *Gibbs Sampling*, que retorna uma série de *motifs* com matrizes de pesos que são "super representadas" em relação aos dados de entrada.	Avalia alinhamentos testados durante a execução do algoritmo usando um *score* máximo de probabilidade (a priore um logaritmo), que estima o grau de "super representatividade". Além disto, o algoritmo faz uso de mais uma mensuração que leva em conta a seqüência de entrada do genoma e destaca os "*motifs*" encontrados preferencialmente em associação com os genes anteriormente considerados.	Roth et al. (1998)
BioProspector	Um Algoritmo baseado no modelo de Markov, para localizar as Regiões regulatórias.	O programa usa como base um modelo de Markov. A significância de cada modelo encontrado tem seu *score* estimado baseando-se no método de Monte Carlo. Para agilizar o programa são previstas modelagem de *gaps* e em seqüências palíndromas.	Liu et al. (2001)
Consensus	Baseado em Matriz de peso, que procura as matrizes contendo os resultados com valores máximos.	O método de funcionamento basicamente consiste em encontrar primeiramente um par de seqüência que compartilha uma "super-representatividade", então procura-se por um terceira seqüência que pode ser adicionada as seqüências já encontradas, maximizando a possibilidade desta seqüência ser um "*motif*".	Hertz e Stormo (1999)
GLAM	Baseado em um algoritmo usando *Gibbs Sampling* onde as larguras dos alinhamentos são otimizados, pontuando os estatisticamente significantes.	Longas seqüências são fragmentadas em pequenas seqüências, e o alinhamento é transformado em Matrizes de pesos que são usadas para procurar prováveis regiões regulatórias.	Frith e Stormo (2004)
MDSCAN	Combina duas técnicas para buscar regiões regulatórias, o *"ChIP"* (Micro-arranjo) de cDNA com imunopreciptação da cromatina e matrizes de peso.	A partir de seqüências selecionadas com a técnica do "*ChIP*", o algoritmo localiza "*motifs*" que representam os sítios de ligação da interação protéica-DNA.	Liu et al. (2002)
MEME	Utiliza matriz de peso para prever as regiões regulatórias, também chamada para este caso de "Matrizes de posição".	As seqüências são fragmentadas em dois ou mais pedaços, usando um modelo estatístico para automaticamente escolher as melhores larguras para cada *motif*, sendo estas prováveis candidatas a regiões regulatórias.	Bailey e Elkan (1995)
MotifSampler	Algoritmo que encontra "*motifs*", baseado na criação de matriz usando *Gibbs Sampling*, sendo modelada com conhecimento baseado no modelo de Markov.	A estrutura probabilística é incrementada explorando a estimativa do número esperado de "*motifs"* da seqüência.	Thijs et al., (2002)

Programa	Principal Operador	Técnica Utilizada	Referência
QuickScore	Baseado em um algoritmo de Busca Exaustiva que estima a probabilidade de surgimento de oligômeros freqüentes no genoma.	Incorpora uma extensão do método "*consensus*" e usa expressões matemáticas para melhorar a eficiência computacional no cálculo dos "*scores*".	Régnier e Denise (2004)

Jacques et al. (2006) propõe uma ferramenta para localizar regiões regulatórias unindo técnicas já disponíveis, como Matrizes de Peso, HMM, entre outras e procura analisar as características individuais de cada organismo para então buscar por suas RR.

Existem algumas ferramentas desenvolvidas para a localização de regiões regulatórias que são específicas para um determinado organismo. Estas ferramentas tendem a serem mais precisas que as de uso geral. Algumas destas ferramentas estão descritas na tabela 3.

Tabela 8 - Programas para predizer regiões regulatórias específicos.
Algumas ferramentas computacionais para predição de regiões regulatórias específicas para um determinado organismo.

Organismo Analisado	Técnica Utilizada	Referência
Bacillus subtilis	Método praticamente manual, usando ferramentas como o Editor de texto MS Word para poder fazer os alinhamentos necessários e fazer as análises para predição das regiões regulatórias.	Helmann (1995)
Escherichia coli	Utiliza-se de um método que incorpora propriedades biológicas de cada pedaço do DNA, associado a uma implementação gramatical usando a linguagem Prolog, refinando os resultados obtidos usando-se matriz de peso.	Rosenblueth et al., (1996)
Escherichia coli	Utiliza-se de matriz de peso para predição das regiões regulatórias da bactéria analisada	Robison et al., (1998)
Escherichia coli	Combinação de busca exaustiva com matriz de peso para predizer as regiões regulatórias da bactéria analisada.	Li et al., (2002)
Escherichia coli	Utiliza-se de uma técnica conhecida como "*Support Vector Machines*"(SVM), que tem por finalidade fazer uma combinação de duas seqüências e submete-las a uma pontuação pré-estabelecida para verificar a possibilidade de ser uma região regulatória.	Gordon et al., (2003)
Bacillus subtilis	Utiliza-se do método de busca exaustiva, "clusterizando" os oligômeros similares em matrizes de peso para predição de regiões regulatórias.	Mwangi e Siggia, (2003)
Escherichia coli	Faz uma "varredura" da posição -500 até a 500, calculando a possibilidade de cada base pertencer a uma região regulatória mediante equações matemáticas pré-estabelecidas. Se dois "sinais" separados por 25 bases responderem de forma positiva as equações pré-estabelecidas e estes sinais estiverem entre as posições -150 e 50, será apontada como uma provável região regulatória.	Kanhere e Bansal, (2005)
Escherichia coli	Método de busca que procura identificar oligômeros de tamanho 6 (seis) que repetem-se em posições semelhantes entre as regiões intergênicas da bactéria analisada.	Sivaraman et al., (2005)

2.3.5 Limitações e Potencialidades dos algoritmos para localização de regiões regulatórias.

Trabalhos que analisam as ferramentas citadas na tabela 2 foram realizados por TOMPA et al. (2005) para predição de RR para eucariontes e procariontes, e o trabalho de Hu et al. (2005) é específico para procariontes.

Ambos os trabalhos chegaram a determinadas conclusões semelhantes, principalmente em relação à capacidade de predição das regiões regulatórias, que ainda é considerada baixa. Ou seja, os algoritmos desenvolvidos para predizer RR com o objetivo de serem usados para vários organismos distintos, têm taxas de acerto baixa. Em média, a taxa de predição fica em torno de 15% a 25% , dependendo da ferramenta e do organismo analisado.

Os principais aspectos que ainda dificultam a criação de algoritmos genéricos capazes de predizer uma maior quantidade de regiões regulatórias, segundo os artigos mencionados, TOMPA et al. (2005) e HU et al. (2005) são:

- O mais importante entre todos os aspectos é a falta de uma completa compreensão biológica de todos os mecanismos envolvidos no processo regulatório. Esta falta de compreensão dificulta a criação de ferramentas computacionais mais precisas;

- As ferramentas para extração das bases de dados para os testes podem conter alguns erros nas seqüências, dificultando uma análise mais precisa dos resultados. Estas bases recebem revisões constantes, ou seja, com o passar do tempo, elas irão ficar cada vez mais confiáveis;

- Pode ocorrer que RR de um determinado organismo estejam próximas umas das outras, dificultando a localização por ferramentas computacionais;

- A taxa de acerto dos algoritmos decai significativamente quanto maior for o tamanho do oligômero analisado e quanto maior for o tamanho da região "*upstream*" analisada. Porém, quanto menor o tamanho do oligômero usado, maior a probabilidade de encontrar resultados falsos e quanto menor for o tamanho da região "*upstream*", pode ocorrer de algumas RR relevantes não serem localizadas;

- Outro fator que deve ser levado em consideração é a heurística escolhida pelo usuário na configuração dos parâmetros usados nos algoritmos. Como ainda não existe um perfeito conhecimento de todas as características das RR, as escolhas ficam na dependência de cada usuário, não havendo ainda um padrão cientificamente comprovado para a melhor parametrização.

Os mesmos autores que relatam as principais limitações dos algoritmos já desenvolvidos sugerem algumas propostas de melhorias, sendo as mais relevantes:

- Com o avanço do conhecimento na área da biologia molecular, a criação de ferramentas computacionais mais precisas para a predição de regiões regulatórias será facilitada;

- A melhoria no tratamento dos algoritmos para usar oligômeros mais longos na busca por RR, melhorando desta maneira a taxa de acerto na predição dessas regiões.

No trabalho desenvolvido por Tompa et al. (2005) percebeu-se que a junção de algoritmos distintos para predição de RR melhorou a taxa de acerto. Os algoritmos acabam se complementando, ou seja, cada algoritmo prediz certos candidatos a serem RR, assim sendo, a soma das predições de cada algoritmo pode gerar melhores resultados.

Hu et al. (2005), também destacam que a junção de várias técnicas distintas em um mesmo algoritmo pode melhorar o desempenho global. A execução de vários algoritmos de predição de RR para um mesmo organismo, para posterior comparação e junção das informações obtidas, pode ser um caminho promissor para melhorar a eficiência dos algoritmos.

Observa-se, portanto, que este é um campo de pesquisa que ainda necessita de muita exploração e descobertas, abrindo a possibilidade de desenvolvimento de novas ferramentas computacionais que possam auxiliar neste trabalho.

2.4 Computação Evolutiva

A teoria evolutiva inspira a computação evolucionária, sendo um nome genérico, dado a métodos computacionais. Na computação evolucionária os algoritmos mais conhecidos são os algoritmos evolucionários (AEs) (BARRETO, 1996; AZEVEDO, 1999).

O algoritmo genético é um dos tipos de algoritmo evolucionário, onde encontram-se também a programação genética (PG), programação evolucionária (PE) e a estratégia evolutiva (EE). Todos partilham de uma base conceitual comum, que consiste na simulação da evolução de estruturas individuais, via processo de seleção e os operadores de busca, referidos como operadores genéticos (OG), tais como mutações e cruzamento. Todo o processo depende da aptidão que cada solução tem frente a um determinado ambiente.

Os paradigmas da computação evolutiva (ou computação evolucionária) são também denominados algoritmos evolutivos (ou algoritmos evolucionários). Os algoritmos evolutivos (AEs) são sistemas computacionais para resolução de problemas baseados nos princípios da teoria evolutiva e na genética. Uma variedade de algoritmos evolutivos têm sido desenvolvida e todos dividem uma base conceitual comum, através de procedimentos de seleção, mutação e recombinação. Entre os algoritmos de busca destaca-se o AG.

Algumas das principais características do AG são descritas em Goldberg (1989) e Coelho e Coelho (1999), que são:

- Operar em uma população de pontos, que podem ser possíveis soluções para o problema proposto;
- Não requerer cálculos de derivadas e informações sobre o gradiente da função objetivo;
- Trabalhar com a codificação de um conjunto de parâmetros, podendo variar de acordo com o problema a ser resolvido;
- Realizar transições probabilísticas, em vez de regras determinísticas;
- Necessitar apenas da informação sobre o valor da função objetivo de cada indivíduo da população para poder evoluir;
- Apresentar simplicidade conceitual.

2.4.1 Algoritmo Genético

Em meados da década de 70, John H. Holland propôs a técnica de algoritmo genético (AG), com a publicação do livro: "*Adaptation in Natural and Artificial Systems*". (SRINIVAS e PATNAIK, 1994b).

O AG é considerado um método robusto, utilizado para resolver problemas em pesquisas numéricas, otimização de funções e aprendizagem de máquina, dentre outras áreas (FOGEL, 1995). A aplicação do AG é destacada em sistemas classificadores de dados para ordenação destes dados em um determinado propósito, como, por exemplo, para a simples recuperação ou para efetuar uma análise de dados (FURTADO, 1998). Whitley (1994) presume que os AG são freqüentemente descritos como um método de busca global, não utilizando gradientes de informações e podem ser combinados com outros métodos para refinamento de buscas quando há aproximação de um local máximo ou mínimo.

Para Goldberg (1994):

> *"Os AGs são técnicas não-determinísticas de busca, otimização e aprendizagem de máquina, que manipulam um espaço de soluções potenciais utilizando mecanismos inspirados nas teorias de seleção natural de C. Darwin e na genética de G. Mendel. Os AGs são robustos e eficientes em espaços de procura irregulares, multidimensionais e complexos."*

Atualmente o AG é empregado na resolução de problemas de cunho genético, como exemplos, na predição de estruturas de RNA (FOGEL et al., 2002), para a predição de "*non-coding*" RNA (ncRNA) (SATROM et al., 2005), para a construção de mapas de DNA (WALKER et al.,1994) e para a descoberta de regiões regulatórias (AERTS et al., 2004; FOGEL et al., 2004).

2.4.2 Algoritmo Genético e sua Inspiração Biológica

A primeira teoria sobre evolução das espécies foi proposta em 1809, pelo naturalista francês Jean Baptiste Pierre Antoine de Monet, conhecido como Lamarck. Para Lamarck as características que um animal adquire durante sua vida podem ser transmitidas hereditariamente. Este estudo ficou conhecido pela ciência como a "lei do uso e desuso" (DARWIN, 2004).

Charles Darwin vem debater a teoria de Lamarck de forma agressiva, tentando de forma científica explicar como as espécies evoluem. A seleção natural é parte do processo evolutivo,

geralmente aceito pela comunidade científica como a melhor explicação para a adaptação, onde o meio ambiente seleciona os seres mais aptos. Em geral, só estes conseguem reproduzir-se e os menos dotados são eliminados. Assim, só as diferenças que facilitam a sobrevivência são transmitidas à geração seguinte (STEARNS, 2003).

A seleção natural depende muito das condições ambientais, podendo selecionar características de um determinado organismo ajudando na reprodução e sobrevivência deste. Os organismos que não possuem tais características podem vir a morrer antes que se reproduzam ou serem menos prolíficos que os organismos mais aptos. (FUTUYAMA, 2003; STEARNS, 2003).

Para Darwin (2004):

> *"Pode-se dizer que a seleção natural realiza seu escrutínio dia-a-dia, hora-a-hora, pelo mundo, de qualquer variação, mesmo as mais sutis; rejeitando aquelas que são ruins e preservando e fazendo prosperar todas as que são boas; trabalhando silenciosa e imperceptivelmente, onde e quando surgir a oportunidade na melhora de cada ser orgânico em relação a suas condições orgânicas e inorgânicas de vida. Não vemos nada desse lento progresso, até que os ponteiros do relógio das eras tenham marcado um longo lapso de tempo e assim tão imperfeita é a nossa visão sobre o profundo passo das eras geológicas que tudo o que podemos ver é que as formas de vida são agora diferentes daquelas que existiam antes."*

Alguns pesquisadores buscaram na natureza a inspiração para novas técnicas de busca de soluções para determinados problemas. Na natureza, o processo de seleção natural demonstra que seres mais preparados (aptos) competem com os recursos naturais impostos, tendo assim maiores probabilidades de sobreviver, conseqüentemente, disseminam o seu código genético (SRINIVAS e PATNAIK, 1994b). Com o passar das gerações, através dos cruzamentos e das mutações que ocorrem com as espécies, estes tendem a estar cada vez mais adaptados ao meio ambiente em que vivem.

O AG trabalha com uma população no qual cada elemento pode ser a solução para o problema. A função de otimização representa o ambiente no qual a população inicial encontra-se. Emprega-se no AG a mesma terminologia e os mesmos princípios da teoria evolutiva e da genética conforme exemplificado na figura 9 (DIAS e BARRETO, 1998).

Biologia	Algoritmo Genético
Cromossomo	Indivíduo (“*string*”)
Gene	Bit
Alelo	Valor do bit
Lócus	Posição de um bit específico no indivíduo ou “*string*”
Genótipo	Indivíduo candidato a solução – *x*
Fenótipo	Valor da função para um dado indivíduo – *f(x)*

Figura 9 - Ligação terminológica entre AG e biologia.

Tratando do AG e tentando demonstrar um pouco a relação com a seleção natural, pode-se expressar como seguinte lei geral (DARWIN, 2004).

1 – SE há organismos que se reproduzem e

2 – SE os descendentes herdam as características de seus genitores e

3 – SE há variação nas características e

4 – SE o ambiente não suporta todos os membros de um população em crescimento,

5 – ENTÃO aqueles membros da população com características menos adaptativas (determinadas pelo ambiente) terão menores chnces de sobreviver e

6 – ENTÃO aqueles membros mais adaptados (determinadas pelo ambiente) prosperarão, tendo como resultado a evolução das espécies.

2.4.3 Componentes do Algoritmo Genético

Geralmente existem apenas dois componentes principais utilizados no AG, que dependem do problema a ser resolvido: a representação do problema e a função de adaptação (WHITLEY, 1994). Os outros componentes que completam este processo são: população, seleção, cruzamento e mutação (HAUPTY e HAUPTY, 2004).

Fundamental para a estrutura do AG é o mecanismo que será utilizado para a representação do problema, dependendo essencialmente de sua natureza para ser resolvido. Esta representação pode usar números binários (0 e 1), números inteiros ou reais (SRINIVAS & PATNAIK, 1994b).

Geralmente, cada solução possível (indivíduo ou cromossomo), possui um tamanho fixo. Por exemplo, se for usada a representação binária (0 e 1, representada por *K*) e o

indivíduo tiver tamanho (*S*) e *S* = 6 , todos os indivíduos terão o mesmo tamanho, como, por exemplo, 011101 e 111010. Conclui-se que a quantidade de indivíduos possíveis para esta representação será 64, ou generalizando K^S, onde *K* pode variar de acordo com o alfabeto utilizado. O importante desta representação é que cada indivíduo representa um ponto de busca no espaço das possíveis soluções para o problema (KOZA, 1995).

A função de adaptação é a função que deve ser otimizada. Ela possui o mecanismo de evolução para cada indivíduo (SRINIVAS & PATNAIK, 1994b), também conhecida como função objetivo. Esta função avalia cada indivíduo da população, gerando uma pontuação, dando a ele a chance de participar do processo reprodutivo para as próximas gerações. A avaliação é independente, mas o seu grau de adaptação ao ambiente vai depender dos demais indivíduos da população (WHITLEY, 1994). Dias e Barreto (1998), escrevem sobre a função de adaptação chamando de função custo como:

> *"A Função custo é uma função matemática representativa do problema (ambiente onde a população de indivíduos está inserida). A função custo não precisa ser o modelo do processo a ser otimizado, até mesmo porque se o modelo existisse e fosse bem comportado, os métodos clássicos de resolução seriam mais eficientes e eficazes. Contudo, quanto mais representativa do problema for a função custo, maiores são as chances de sucesso da otimização do AG."*

A população é um grupo de indivíduos candidatos à resolução do problema. A população é uma matriz com duas dimensões, podendo ser representada como N_p x N_T, onde N_p é a quantidade de indivíduos de uma determinada população e N_T é o tamanho de cada indivíduo (HAUPTY e HAUPTY, 2004). O tamanho da população irá depender do problema. Quanto maior a população, maior a chance de encontrar a solução para o problema. Porém, quanto maior a população, maior será o tempo de processamento, ou seja, a escolha do tamanho de uma população irá depender de alguma heurística utilizada pelo usuário e de sua experiência (DIAS e BARRETO, 1998).

Na natureza, segundo Darwin, o processo de seleção garante que os indivíduos mais adaptados tenham maiores chances de sobrevivência. No AG, o conceito usado é o mesmo (SRINIVAS e PATNAIK, 1994b). Para Whitley (1994), a seleção é aplicada na população corrente para criar uma população intermediária que irá passar pelos processos de cruzamento e mutação (que serão explicados mais adiante) para a geração da próxima população.

Existem várias técnicas de seleção, sendo a utilizada no algoritmo clássico, sugerido por Holland (1975) conhecida como "*roulette wheel*" (MICHALEWICZ, 1996). Esta atribui a

cada indivíduo uma probabilidade de passar para a próxima geração, proporcional a sua adaptação ao ambiente, em relação à somatória da adaptação de todos os indivíduos, sendo maior a probabilidade dos indivíduos mais adaptados serem sorteados. Um dos problemas deste método é a forte pressão seletiva, ou seja, há uma tendência de todos os indivíduos convergirem rapidamente para um mesmo ponto, que não necessariamente seja o máximo global, principalmente se um dos indivíduos tiver um valor de "*fitness*", muito maior que os demais (DEB, 1997).

Existem outros métodos mais eficientes que o modelo "*roulette wheel*", pois tendem a diminuir a pressão seletiva. Um destes métodos é conhecido por "*Rank*", segundo (GREFENSTETTE, 1997). Esta estratégia utiliza as posições dos indivíduos quando ordenados de acordo com o "*fitness*", para determinar a probabilidade de seleção para a próxima geração. Mesmo existindo um indivíduo com um "*fitness*" muito elevado em relação aos demais, o processo de seleção por "*Rank*" irá auxiliar a evitar a prematura convergência para um determinado ponto, porque este "super-indivíduo" sempre terá a mesma probabilidade de seleção, independente da função de adaptação.

Outro método de seleção utilizado é conhecido como "torneio". Segundo Blickle (1997), um grupo de "*q*" (*q >= 2)* indivíduos são selecionados aleatoriamente da população, onde este grupo participa de um torneio, sendo o vencedor, o indivíduo que tiver o melhor "*fitness*". Este indivíduo será selecionado para a próxima etapa do AG que é o cruzamento. Este processo é repetido "n" vezes até se obter a nova população.

Quanto maior o tamanho *"q"*, maior será a pressão seletiva. Para a maioria dos programas que usam o método de seleção por torneio em AG, recomenda-se que o valor de *"q" ∈ {6,...,10}* indivíduos (BLICKLE, 1997).

Durante o processo de seleção pode-se perder um indivíduo com um alto grau de adaptação. Para evitar este problema, usa-se um conceito conhecido como elitismo, onde o melhor indivíduo, ou os melhores indivíduos, são passados diretamente para a próxima geração (ZUBEN, 2000).

Segundo Srinivas e Patnaik (1994b), após a seleção vem o cruzamento, sendo esta operação essencial para o AG. Indivíduos pré-selecionados formam pares aleatórios para o processo de cruzamento, que é a criação de um ou mais indivíduos (filhos) dos indivíduos selecionados (pais) pela seleção. No final deste processo a população geralmente permanece

do mesmo tamanho da população anterior (HAUPTY e HAUPTY, 2004). Como esclarecem Dias & Barreto (1998) e Zuben (2000), existem várias formas de cruzamento, dentre elas destacam-se: cruzamento de 1-partição, de 2-partições e cruzamento com (*n*)-partições. Na figura 10, segue um exemplo de cruzamento de 1-partição que é a escolha aleatória de um ponto de corte nos pais onde será processada a troca de material genético.

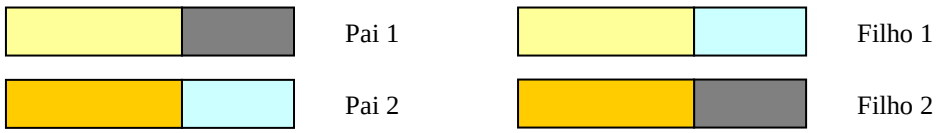

Figura 10 - Cruzamento com uma-partição.
Um ponto aleatório dos indivíduos (divisão de cores) é escolhido e seu material genético é trocado.

Outra forma de cruzamento é o cruzamento uniforme, conforme exposto em Syswerda (1989), onde para cada bit do filho é decidido qual pai vai contribuir com o valor para aquela posição.

A mutação permite que indivíduos da nova geração sofram pequenas alterações, permitindo assim uma possibilidade de busca maior no espaço do problema. O processo inicia-se com a escolha de um ponto aleatório de um indivíduo, dentre um grupo de outros indivíduos, depois é aplicado uma taxa de probabilidade de troca deste ponto (bit) por um outro bit (KOZA, 1995). Para Michalewicz (1997), este processo de mutação é conhecido como mutação uniforme.

Zuben (2000), esclarece que:

> *"A probabilidade de ocorrência de mutação de um gene é denominada taxa de mutação. Usualmente, são atribuidos valores pequenos para a taxa de mutação. A idéia intuitiva por traz do operador de mutação é criar uma variabilidade extra na população, mas sem destruir o processo já obtido com a busca".*

Whitley (1994), explica que a taxa de mutação geralmente é pequena, na ordem de 1%, sendo que para Srinivas e Patnaik (1994b), a mutação é um operador secundário que pode restaurar material genético perdido de gerações anteriores. Na Figura 11, têm-se um exemplo de uma mutação em um indivíduo.

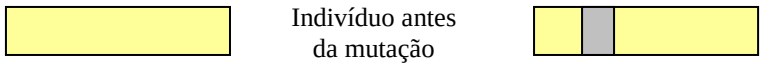

Figura 11 - Exemplo de mutação.
Um indivíduo sofreu a mutação em um ponto (bit), observa-se pela troca de cor o ponto de mutação.

2.4.4 Teorema Fundamental do Algoritmo Genético

Os conceitos anteriormente explicados são as bases para a equação matemática que explica o funcionamento do AG, também conhecida como Teoria do Esquema que foi proposta por Holland em 1975.

Um esquema pode ser definido como sendo um subconjunto de um indivíduo com certas posições similares (HOLLAND, 1975). Um exemplo é apresentado na tabela 4.

Tabela 9 - Exemplo de esquema.
Um indivíduo e 3 possíveis esquemas, o "*" representa que qualquer bit pode ser colocado entre os bit padrões.

Indivíduo	**11011**
Esquema (E1)	1***1
Esquema (E2)	11**1
Esquema (E3)	**01*

Os esquemas têm duas propriedades que os quantificam: A ordem do esquema (O (E)) e o comprimento do esquema (δ (E)). A ordem de um esquema E, O (E), representa o número de 0's e 1's fixos no esquema, por exemplo, no esquema 11**1, a ordem O (E) = 3. O comprimento do esquema E; δ (E) representa a distância entre a primeira e a última posição de interesse no esquema. Por exemplo, o esquema 1***1 tem comprimento $\delta(E) = 5 - 1 = 4$, pois a última posição é 5 e a primeira é 1 (DIAS e BARRETO, 1998).

Srinivas e Patnaik (1994b) representam o teorema fundamental do AG como segue abaixo:

$$N(h,t+1) >= N(h.t)\frac{f(h,t)}{f'(t)}\left[1 - pc\frac{\delta(h)}{l-1} - pmo(h)\right]$$

Onde

$f(h,t)$: valor médio da adaptação do esquema h na geração t;

$f'(t)$: valor médio da adaptação da população na geração t;

pc: probabilidade de cruzamento;

pm: probabilidade de mutação;

$\delta(h)$: comprimento definido de um esquema;

$o(h)$: Ordem de um esquema h;

$N(h,t)$: Número esperado de instâncias de um esquema h na geração t;

l: É o número de bits em uma "*string*".

Segundo Dias e Barreto (1998):

> *"A Teoria fundamental do algoritmo genético ou teorema dos esquemas, segundo o qual os esquemas que tiverem adaptação superior a adaptação média da população crescerão exponencialmente, enquanto que os que tiverem adaptação média inferior decrescerão exponencialmente.*
> *Essa característica é altamente promissora. No entanto, ela depende de fatores ainda com forte predomínio heurístico, tais como a probabilidade de ocorrer cruzamento, pc, e a probabilidade de ocorrer mutação, pm. Há também a influência do número de indivíduos necessários à composição da população ou espaço de busca e do inter-relacionamento desses parâmentros entre si em função do problema a ser otimizado."*

2.4.5 Funcionamento do Algoritmo Genético

Este tópico foi descrito com base nas seguintes referências da literatura: Srinivas e Patnaik (1994b), Jong et al. (1997), Whitley (1994), Michalewicz (1996), Dias e Barreto (1998), Zuben (2000) e Haupty e Haupty (2004).

O algoritmo genético básico envolve seis etapas: (i) geração da população; (ii) avaliação da população; (iii) teste de convergência ou critério de término para a otimização; (iv) seleção e (v) aplicação dos operadores do AG; e (vi) criação de uma nova geração. Na figura 12, tem-se um fluxograma do AG e a seguir tem-se o algoritmo básico do AG:

- ✓ *Definir a função de adaptação;*
- ✓ *Definir as variáveis e parâmetros do AG;*
- ✓ *Gerar população inicial;*
- ✓ *Enquanto critério de término:*
  - *Avaliar cada indivíduo;*
  - *Selecionar os indivíduos;*
  - *Processar a operação cruzamento;*
  - *Processar a operação mutação;*
  - *Gerar nova população;*

- ✓ *Fim enquanto;*
- ✓ *Imprime os valores.*

A primeira etapa é a definição de qual será a função para representar o problema. Esta é a “chave” para um resultado satisfatório do AG. Quanto melhor for a representação do problema, maior será a probabilidade do AG encontrar bons resultados. Um dos cuidados a serem observados na criação da função de adaptação é a pontuação (“*fitness*”) a ser dada para cada indivíduo. Isto porque, dependendo da formulação usada, alguns indivíduos podem ter um “*fitness*” alto em relação aos demais indivíduos, podendo acarretar no aumento da pressão seletiva da população, gerando com isto uma perda rápida de diversidade.

Após a definição da função de adaptação, pode-se acrescentar a parametrização do sistema, ou seja, é neste momento que as variáveis são iniciadas. Alguns exemplos de variáveis que devem ser iniciadas são: tamanho da população; quantidade de gerações; taxa de seleção; taxa de mutação; tamanho do indivíduo; número de indivíduos que passarão automaticamente para a próxima geração (elitismo); critério de término; entre outros.

O próximo passo é a geração da população inicial, com distribuição uniforme pelo espaço de busca. Quanto melhor for esta distribuição pelo espaço de busca, maior será a tendência da função de adaptação “encontrar” bons candidatos na resolução do problema proposto.

Após a definição da função de adaptação e dos parâmetros a serem utilizados no processo, inicia-se o ciclo da evolução da população. Inicialmente, cada indivíduo será submetido à função de adaptação do problema, sendo atribuído a ele um valor de acordo com a sua aptidão. Quanto maior é o valor recebido pelo indivíduo, maior é o seu grau de adaptação ao “ambiente” analisado.

Em seguida serão selecionados os indivíduos que irão passar pelo processo de cruzamento e mutação. A tendência é sempre selecionar os melhores indivíduos da população que irão passar os seus “genes” para as gerações futuras, enquanto os piores indivíduos tenderão a ser descartados da população. Deve-se tomar providências para evitar que “super-indivíduos” dominem toda a população, gerando uma forte pressão seletiva e consequentemente uma perda de diversidade rápida. Na seção 2.4.3, foram abordados algumas formas de seleção que tendem a minimizar este problema.

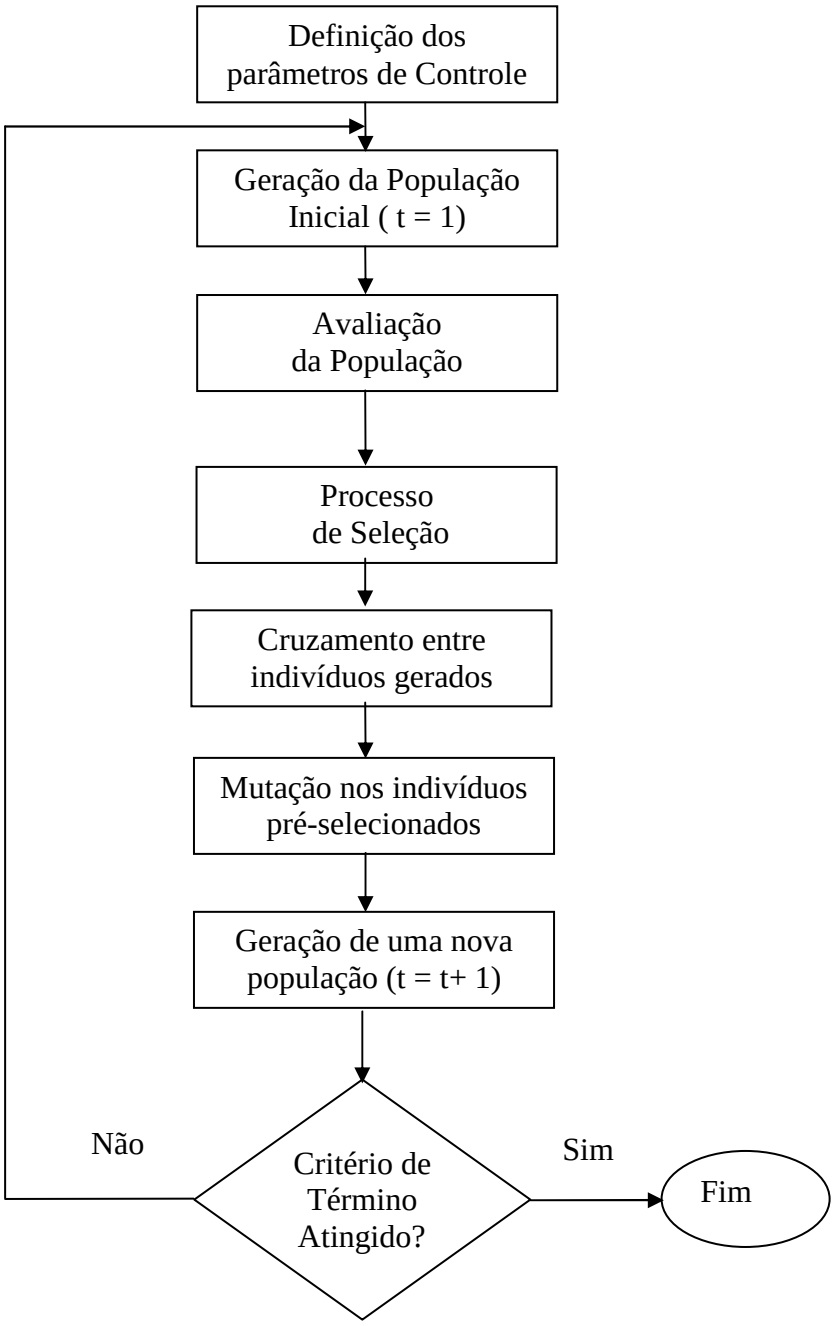

Figura 12- Fluxograma do AG

Depois de selecionados os indivíduos, serão aplicados os operadores de cruzamento e mutação. Primeiramente, o cruzamento, que divide os indivíduos selecionados em pares aleatórios, gerando novos indivíduos através da troca de material genético entre eles. Uma determinada percentagem destes novos indivíduos podem sofrer uma pequena mudança em seu genoma, chamando-se este processo de mutação. Após o processo de cruzamento e mutação, será gerada uma nova população em substituição a anterior, que geralmente tem o

mesmo número de indivíduos da população antiga, porém estes indivíduos podem ser totalmente diferentes da geração anterior. Os indivíduos da nova geração são novamente avaliados pela função de adaptação, começando assim um novo ciclo.

O ciclo de gerações de novas populações continuará até um critério de parada ser verdadeiro. Existem vários critérios de parada, que podem ser a quantidade parametrizada de gerações máximas; ou a extrapolação de um tempo máximo de execução; ou a estabilização da população em um determinado patamar médio de valores, que pode ser medido pelo desvio padrão entre as distâncias relativas de cada indivíduo. Por exemplo, pode-se determinar que se após “n” gerações não houver mudanças significativas na pontuação média dos indivíduos, o processo pare a execução.

Após o término da execução, pode ser gerada uma listagem da última população com suas respectivas pontuações, ou dependendo do problema a ser resolvido, listar outras informações que podem ser relevantes para uma análise complementar.

3 Metodologia

3.2 Extração da Base de Dados

Para desenvolver o algoritmo GA_FIND_RR, com o objetivo de predizer RR em organismos bacterianos (procariontes), partiu-se de três características importantes das regiões regulatórias: (i) a super-representatividade, (ii) a separação destas regiões em dois oligômeros distintos, separados por uma quantidade variável de bases, (iii) e o fato destas regiões situarem-se antes do início do TSS (ROSENBLUETH et al., 1996 e Li et al., 2002).

O primeiro passo foi a escolha do organismo a ser usado no processo de criação e validação do algoritmo. Após algumas analises com *Mycoplasma synoviae* (VASCONCELOS et al., 2005), *Escherichia coli* (BLATTNER et al., 1997) e *Bacillus subtilis* (KUNST et al., 1997), decidiu-se usar o *Bacillus subtilis*, por ser um organismo amplamente estudado e com documentação detalhada sobre técnicas e resultados obtidos na busca por regiões regulatórias. Como exemplo, pode-se destacar os trabalhos desenvolvidos por Helmann (1995), Li et al. (2002), Helden (2003), Mwangi e Siggia (2003), Makita et al. (2004) e Jacques et al. (2006).

Após a escolha do organismo, foram extraídas as bases "*upstream*", utilizando-se a ferramenta "RSATools" (HELDEN, 2003), de uso público, através da internet, pelo site: http://RSATools.ulb.ac.be/RSATools/, cuja interface com o usuário pode ser vista na figura 13. A primeira base "*upstream*" considerada, neste trabalho, é uma base antes do códon de iniciação.

Foram extraídas inicialmente as 300 primeiras bases "*upstream"* (0 a -300) para a execução do algoritmo, sendo criado uma matriz de 3567 X 300, onde as linhas representam os genes que possuem pelo menos uma base intergênica. O *Bacillus subtillis* tem aproximadamente 4100 genes, mas em alguns casos, não existem regiões intergênicas. Devido a esta característica, o número de linhas geradas é menor que o número de genes totais. Nem todas as linhas possuíam necessariamente 300 colunas, pois existem situações que a região intergênica é menor que 300 bases.

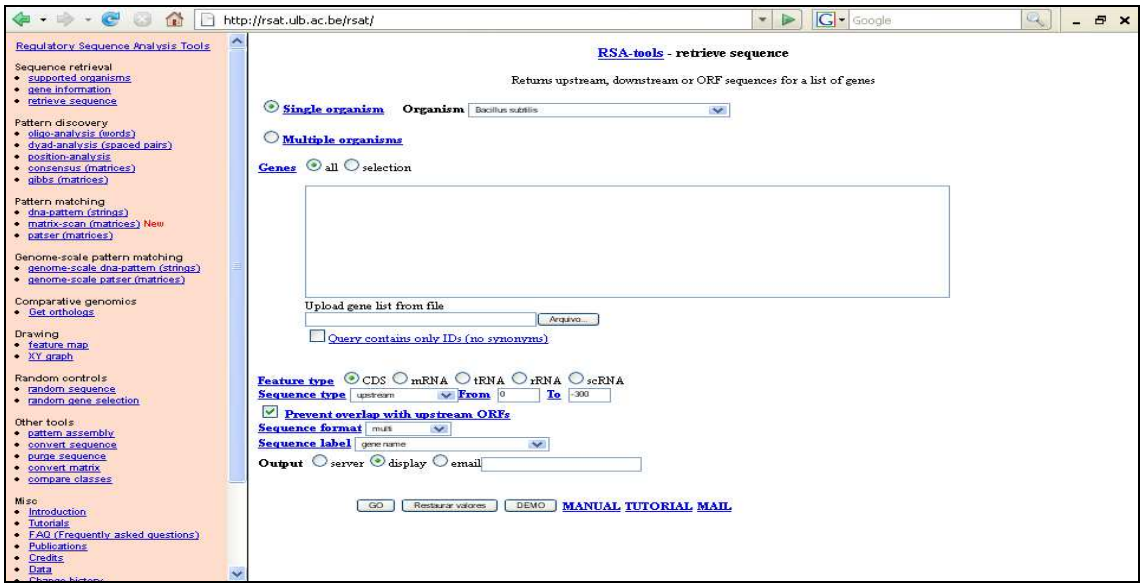

Figura 13 - Tela do site RSAT (2006).

Os principais parâmetros selecionados foram: seleção de todos os genes (*all*), forma de extração (*Feature type*, usou-se CDS: "*coding sequences*"), tipo da seqüência (*Sequence type*, usou-se a região "*upstream*" de 0 até a quantidade upstream desejada), o formato de apresentação (*sequence format*, usou-se o "multi" que traz apenas as bases sem informações adicionais) e o tipo de saída (*Output*, em tela, facilitando copiar o resultado para um editor de texto).

A escolha desta quantidade de bases baseou-se no artigo escrito por Mwangi e Siggia (2003). Estudos anteriores indicavam que existem altas concentrações de RR nas 50 primeiras bases "*upstream*" (HELMANN, 1995). Porém, os autores (Mwangi e Siggia (2003)) usaram uma quantidade maior de bases "*upstream*" com o objetivo de inferir a possibilidade de existirem RR mais "afastadas" do início do gene.

Após uma análise em artigos mais recentes a Mwangi e Siggia (2003), como, por exemplo, Jacques et al. (2006), reforçam as conclusões de Helmann (1995), decidiu-se usar 100 bases "*upstream*", pois este número já prevê uma "margem de segurança" no caso de existirem RR mais afastadas. Sendo assim, foi decidido usar 100 bases "*upstream*" como padrão para os testes executados, sendo criada uma matriz com 3567 x 100.

O trabalho de Helmann (1995) conclui que o oligômero mais freqüente é TTGACAN$_{\sim 16}$TATAAT. Já o trabalho de Mwangi e Siggia (2003) lista como o oligômero mais significativo TTGAN$_{20}$ATAAT e o trabalho de Jacques et al. (2006) conclui que uma das regiões regulatórias mais abundantes para o *Bacillus subtilis* é CCTTGACAAGN$_{16}$ATAATA. Estes oligômeros listados ficam em regiões compreendidas, em sua maior parte, nas 50 primeiras bases *"upstream"*, comprovando que a quantidade de 100 bases é suficiente para testar o algoritmo, para o *Bacillus subtilis*.

3.3 Descrição do algoritmo

Após a criação da matriz de 3567 X 100, esta foi incluída dentro do código fonte do programa em formato de matriz com o objetivo de usar as ferramentas de busca desenvolvidas pelo MatLab da Mathworks, objetivando um melhor desempenho em sua execução em relação a necessidade de ler um arquivo texto separadamente. Algumas linhas contêm menos de 100 bases, pois algumas regiões *"upstream"* são menores que a quantidade base de colunas (100). A contagem inicia-se em 0 (base mais próxima do gene) e termina em -100 (base mais afastada do gene), conforme o exemplo apresentado na figura 14.

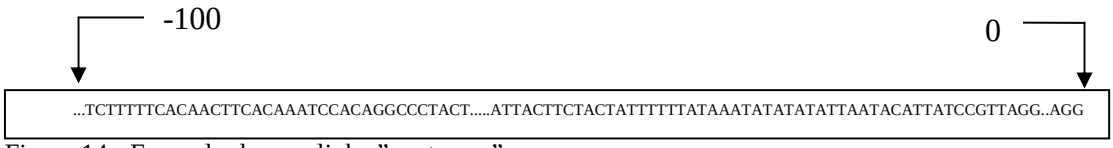

Figura 14 - Exemplo de uma linha "*upstream"*.
A primeira base à direita é a posição 0 e a última base a esquerda é a posição -100.

Após a criação da matriz, foram aplicadas as técnicas abordadas na seção 2.4 referente ao AG, com as características mencionadas nas próximas subseções.

3.3.1 População Inicial

A população inicial foi criada usando números binários para representar os oligômeros, pois as funções padrões da ferramenta usada para desenvolver o programa computacional em MatLab trabalha apenas com números reais ou números binários, sendo a numeração binária usada para a representação do problema proposto. Cada base foi representada por um conjunto de dois bits, sendo convencionado o seguinte critério:

- 00 → A;
- 01 → C;
- 10 → G;
- 11 → T.

Para representar um oligômero, foi usado a seguinte equação: $Q = B * 2$, sendo:

Q → Quantidade de Bits necessários para representar um oligômero;

B → Comprimento total do oligômero;

Ou seja, para representar um oligômero de 8 bases, foram usados 16 bits, como por exemplo: GTTACTAT , tem-se: 1011110001110011.

Após a geração da população inicial e antes de começar o processo de avaliação da adaptação do indivíduo, o oligômero foi convertido para caracteres, de acordo com a convenção estipulada. Após a conversão, o oligômero foi dividido em duas partes de mesmo tamanho (número de bases iguais). Como por exemplo, para um indivíduo gerado com os bits 1011110001110011, tem-se, após o processo de conversão, duas partes, sendo a primeira **GTTA** e a segunda **CTAT**. Este processo foi feito para todos os indivíduos da população.

A decisão de usar duas partes de mesmo tamanho, baseia-se no trabalho executado por Helmann (1995), onde foi observado que as RR tendem, por padrão, ter oligômeros de tamanhos iguais.

3.3.2 Função de adaptação

Foram desenvolvidas cinco versão distintas do GA_FIND_RR, com variações da função de adaptação e da estratégia de busca na região *upstream* para avaliação do *fitness* dos indivíduos. As cinco versões foram baseadas na super-representatividade dos oligômeros candidatos a serem RR e na separação destes oligômeros em duas partes, podendo variar entre 1 e 30 bases a distância entre elas. Para todos os indivíduos gerados foram necessários dois processos iniciais:

1) Conversão dos indivíduos gerados pelo AG de *bit string* para letras (Bases A, C, G, T) ;

2) Separação destes indivíduos em duas partes de mesmo tamanho.

Na figura 15 tem-se um fluxograma da função de adaptação.

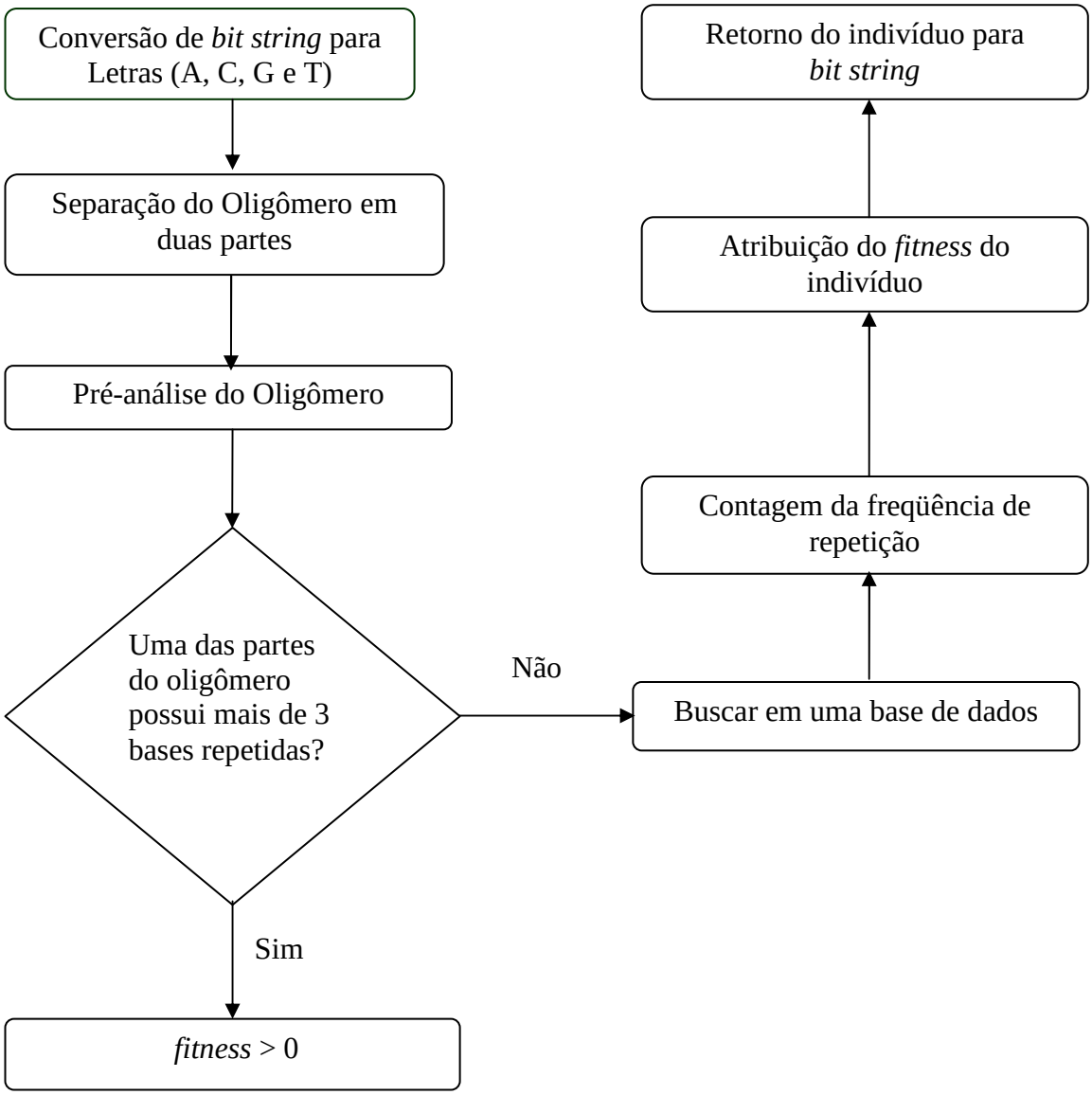

Figura 15 - Fluxograma da Função de adaptação.

Após a conversão de *Bit string* para as letras que representam as bases nitrogenadas, o algoritmo procedeu aos seguintes passos para a avaliação de todos os indivíduos:

1. Para cada parte do oligômero foi atribuida uma variável do tipo vetor. Como por exemplo: *dimer_1*(1,:) = "**TTGA**" e *dimer_2*(1,:) = "**ATAA**" ; *dimer_1* e *dimer_2* são vetores de uma linha por quatro colunas, o símbolo ":" é uma sintaxe do MatLab de representação de todas as colunas ou linhas de uma determinada matriz.

2. Conforme Robison et al. (1998), se ambas as partes do oligômero são iguais, como por exemplo, "**ATAT**Nx**ATAT**" ("Nx" é uma seqüência de bases podendo variar de 1 a 30), este oligômero é considerado um forte candidato a ser uma

região regulatória. As partes foram complementares reversas, como por exemplo: "**CTGA**Nx**TCAG**", também pode ser considerado um potencial candidato. Ambas as características (oligômeros iguais ou complementares reversos) também são considerados como "fortes candidatos" no algoritmo desenvolvido por Mwangi e Siggia (2003). No GA_FIND_RR, também foram previstas estas duas situações. Para aumentar o *fitness* do indivíduo que se encontre nas duas situações citadas, usou-se a seguinte equação:

$Ft = fa * Oi * Ocr * (-1)$, onde:

- Ft = *fitness* do Indivíduo;
- fa = Função de adaptação;
- $Oi = Oi > 1$, atribuído quando as duas partes do oligômero são iguais, se as duas partes do oligômero são diferentes $Oi = 1$;
- Ocr = Onde, $Ocr > 1$, atribuído quando o oligômero é complementar reverso, se não for complementar reverso $Ocr = 1$;
- Como o *Toolbox* do Matlab foi configurado para minimizar uma equação, a função de adaptação foi multiplicada por -1.

3. Outra característica abordada no artigo de Mwangi e Siggia (2003) e também considerada no GA_FIND_RR refere-se a repetição de bases nos oligômeros, como por exemplo, "**AAAA**Nx**TTCT**", "**CTAG**Nx**GGGG**" ou **CCCC**Nx**CCCC.** Esta característica não é desejada, por se tratar de um indivíduo com pouca chance de ser considerado uma região regulatória. Para o AG "desprezar" oligômeros com bases repetidas, a função de adaptação (fa), para estes casos, foi definida como: $fa = n$, com $n > 0$. Como as funções desenvolvidas para o *Tollbox* do AG no Matlab visam minimizar uma equação, quanto menor o valor de n, melhor será o *fitness* do indivíduo, consequentemente, quando $n > 0$, o indivíduo é considerado "pouco adaptado", sendo rapidamente eliminado nas gerações seguintes. Como pode ser observado na figura 16, é muito comum encontrar seqüências de bases repetidas, podendo comprometer o resultado de algoritmos que buscam por RR que não descartem

estes oligômeros. Estas regiões podem gerar muitos “ruídos” nos algoritmos devido a sua abundancia. Existem alguns casos de RR com várias bases repetidas, como é o caso do oligômero AAAA$\mathbf{N_5}$TTTT que é uma região ativadora do gene *ComK* do *Bacillus subtillis* (SINDEREN e VENEMA, 1994). Mas como o índice de freqüência de RR com várias bases repetidas não é estatisticamente relevante, segundo Mwangi e Siggia (2003), resolveu-se ignorar estes oligômeros evitando uma forte convergência do GA_FIND_RR para seqüências com oligômeros com bases repetidas. Se o GA_FIND_RR não encontrar nenhuma ocorrência do oligômero na seqüência “*upstream*” analisada, a função de adaptação será expressa como $fa = 0$. Se o GA_FIND_RR encontrar mais de uma ocorrência do oligômero na seqüência *upstream* analisada, a função de adaptação será definida conforme descrito no item 6 desta seção.

```
ATAACAGAGAAAGACGCCATTTTCTAAGAAAAGGAGGGACGTGCCGGAAGA
TATTTTTTATAAATATATATTAATACATTATCCGTTAGGAGGATAAAAA
TTTTTTTTAGTACAATTAGATATTAGTGATATTTGAAAGAGGTCGATATAA
                                        AGCGGGTGACACTGAT
                                      AGAAATGAGGTGAGCAAT
TTTTTATCACGAATATATCGTTTAGAAAAGTGTAGGTGAATGACGTGGCTA
GGACAATCTACTCCCACATATTTCATGTGATACTTCAGGGAGGTTTTTTAA
GCGGGAAAGAGTTGAAATATTTAGATAACGGAAAGGATTAAGAAATATACA
TAGTTGATAATCTACATATAATATTTTGCCGAAAAGAGGGGGGATTTACTAA
CTTAACGGCTTAATTATAGATGAAGAAAATGAAATACGGAGGTCGTACGAT
AATAAGGATTAGAAATCATATAACTATACCTTGATTAGGGGGGACCAAGAAA
                                    GAACATAGGAGCGCTGCTGACA
TGAGGGCTCTTTTTATTTTCGATAAATCAATAAAAAAAGGAGTGTTTCGCA
                                                        A
TATGAAGGTCGGTAACTGACGCACGTTTTTCAGATATAAGGAGGATTCCGA
TTCATACATTGATAGCGATATGAAAGGAGGCGTTTTTCATTCAAATTTATG
CTAAAAAGGCTACATATTAACTATAACTGAAACGGAAAGGAGACTGTCGAT
ATGTAGCCTTTTTAGGCAATGAAAAAACTTTGAAAAGAGAGCTTATCCTTA
ACAAGGTTCATGTATAATGGGAATGATGAATAACGGAGGAGGGCAAACCCG
                                  CAAAATGAAAGAGAGTGAATGCTA
                                          GGGGATAAAAGAACA
                                      GGAGGAAAAAGCGATCCA
GATCTTCTCATAAGCTTGTACTAGAACAAGCGAAGGAGATGAGAAGATTCA
GTATACAATATCCGTTTTAAGGGGAGGCTAACTGTACGGAGGTGGAGAAGA
AGTAGATAATAATAATAAAACTGAGTATAGACACAGGAGTCGATTATCTCA
                                        GGGAAGAGGGTAAGAGCGA
GAGGCACGATATAATAAGGTGTAAGAAGACACATTCAAAGGATTGTTTTCA
```

Figura 16 - Parte do resultado extraído da ferramenta RSATools.
São mostradas as 50 primeiras bases *upstream* do *Bacillus subtilis*, onde pode-se observar a alta incidência de bases repetidas (destacadas em amarelo). Considera-se bases repetidas para o GA_FIND_RR mais de 3 ou 4 bases.

4. Após as análises iniciais dos oligômeros (análise de similaridade, complementar reverso e bases repetidas), as duas partes distintas do oligômero foram

submetidas a uma pesquisa na matriz que contém as regiões *upstream* do organismo analisado, para verificar se existem combinações entre o indivíduo gerado e região *upstream*. Encontrando combinações exatas, o algoritmo gravou em uma matriz [*L* x *C*], sendo "*L*" o número de linhas totais da região *upstream* analisada e "*C*" é a quantidade de ocorrências encontradas na linha analisada. Cada coluna corresponde a posição inicial da parte do oligômero pesquisada. Se em uma determinada linha não foi encontrado nenhuma combinação, o algoritmo atribuiu à primeira coluna o valor 0 (zero). Este processo foi executado para ambas as partes do oligômero e para todos os indivíduos da população, gerando duas matrizes com a mesma quantidade de linhas, podendo ter quantidades variadas de colunas entre as linhas, pois o mesmo oligômero pode se repetir em uma mesma linha em posições distintas. A matriz da primeira parte do oligômero é chamada de *dimer_1* e da segunda parte *dimer_2*.

5. Após a geração das duas matrizes, estas foram comparadas para verificar se as distâncias entre as partes distintas do oligômero estão entre 1 a 30 bases. Como os dois vetores têm exatamente o mesmo número de linhas, pode-se "párea-los" e fazer a subtração das distâncias. O algoritmo fez um laço de repetição comparando as duas matrizes, linha a linha, usando como equação para comparar as distâncias (*Dist*) entre as partes do oligômero analisado: (*Dist = dimer_1[L,C] - dimer_2[L,C] - N),* onde: "*L*" é a linha da matriz e "*C*" é a coluna da matriz e "*N*" é o tamanho da metade do oligômero. Para cada resultado obtido, o algoritmo analisou se o resultado está compreendido entre 1 e 30. Se estivesse, foi somado o valor 1 (um) em um vetor chamado "*super*" (*super[1...30]*) na posição relativa da distância encontrada (*super[1,Dist] += 1*).

6. Para determinar qual foi o *fitness* dentro da geração, foram usados dois critérios. Cada critério gerou versões diferenciadas do GA_FIND_RR, sendo:

   - Um dos critérios usados foi o de atribuir o maior valor gravado na matriz "*super*". Ou seja, a função de adaptação é descrita como:
     fa = super[1, Mv] * (-1), onde: *Mv* é a posição da matriz que contém o maior número inteiro pertencente à matriz *super*. O *Mv = Nx*, ou seja, é a distância entre as bases do oligômero. Como o MatLab tem por objetivo minimizar uma equação, o valor obtido foi multiplicado por -1.

- Uma segunda versão para a função de adaptação foi desenvolvida, tendo com expressão: $\boldsymbol{fa = (\Sigma\ (super[N_{x1}..N_{x30}])\ /\ cont)\ *\ (-1)}$, sendo: *cont = Quantidade de* $N_{xn} > 0$, se *cont = 0* , então $fa = 0$. Esta função calcula a média aritmética de todos os valores maiores que zero armazenados na matriz *super*. A elaboração desta função foi inspirada na idéia de *cluster*, cujo significado é "grupo de coisas semelhantes que estão próximas umas das outras" (CAMBRIDGE UNIVERSITY, 2003). Ou seja, como os oligômeros são iguais e podem se repetir várias vezes em uma mesma seqüência *upstream*, foi considerado que as distâncias variando entre 1 e 30 estão próximas entre si, formando um *cluster*. A decisão de usar a média aritmética e não um somatório simples deve-se ao fato da "super-representatividade". Se fosse usado o critério de soma simples, e um indivíduo "X", tivesse o valor total de sua soma igual a 50, com valores maiores que 1 em todas as colunas da matriz *super*, e um outro indivíduo "Y", tivesse um valor igual a 49, em apenas uma determinada posição da matriz *super*, este último oligômero teria seu *fitness* menor que o primeiro. Porém, biologicamente, o oligômero "Y" é mais "super-representado" que o oligômero "X". Usando o critério da média, o valor de "X" seria: $fa(x) = 50 / 30 = 1,67$. Enquanto para o oligômero Y a função seria: $fa(x) = 49 /1 = 49$. Desta forma "Y" seria considerado mais "super-representado" que o "X".

Um exemplo do funcionamento da função de adaptação pode ser observado nas seqüências da figura 17.

1) Base de Dados Analisada

GATCTATTTCGGTAATTGTGACAACCATTGCAAGCTCTCGTTTATTTTGG TATTATATTTGTGTTTT
CAGGACCGGGGATCAATCGGGGGATCTGGGCGGTAAAAAGTGTGAATA ACTTTTCGGAAGTCATACAGATCTAYYYCGGTAAC
TCCGATACACTGCTGCCGACCCGTCGGCAGCTTTTCTATTCGGTATCTGCT CCGACAAGTTTTCCCTTTCCCTA
AGCGGGTGACACTGATGATCTAAAACGGTAA

2) População de uma geração

POPULAÇÃO
AACCTTAT
TTATAATA
AAATAATA
GATCGGTA
TGTGGGAG

3) Divisão dos Oligômeros

4) Busca na Base de Dados

dimer_1	dimer_2
AACC	TTAT
TTAT	AATA
AAAT	AATA
GATC	GGTA
TGTG	GGAC

5) Tabela com a freqüência por distância

Dimer_1	Dimer_2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
AACC	TTAT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
TTAT	AATA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AAAT	AATA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GATC	GGTA	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1
TGTG	GGAC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6) "*Fitness*" de cada indivíduo calculado pelas duas funções de adaptação

Dimer_1	Dimer_2	$fa = super[1, Mv]) * (-1)$	$fa = (\Sigma\ (super[N_{x1}..N_{x30}]) / cont) * (-1)$
AACC	TTAT	-1	-1
TTAT	AATA	0	0
AAAT	AATA	0	0
GATC	GGTA	-2	-1,5
TGTG	GGAC	0	0

Figura 17: Exemplo do processo para calcular o *fitness* de cada indivíduo.
Para cada versão do GA_FIND_RR é usada apenas uma função de adaptação. Neste exemplo foi demostrado ambas as versões por questões ilustrativas. Para este exemplo, o processo de busca da base de dados baseou-se em procurar a primeira ocorrência do *dimer_1* e localizar o primeiro *dimer_2* que estiver com uma distância entre 1 e 30 do *dimer_1*. O indivíduo AAATAATA, é descartado do processo de busca, pois têm 3 bases "A" em sua primeira parte (*dimer_1*).

3.3.3 Tamanho da população

A escolha para o tamanho da população baseou-se, inicialmente, nos trabalhos de Fogel et al. (2002) e Fogel et al. (2004). No primeiro trabalho, foram usados valores variados para a

população e no segundo foram usados 100 indivíduos. Ambos os trabalhos foram desenvolvidos para organismos eucariontes.

As execuções do GA_FIND_RR, concentraram-se, em sua maior parte, entre 100 e 200 indivíduos.

3.3.4 Seleção

De acordo com Whitley (1994), o processo de seleção é aplicado para criar uma população intermediária antes de ser aplicado os operadores de cruzamento e mutação. Para Fogel et al. (2004), o processo de seleção irá determinar quais indivíduos serão eliminados.

Baseando-se nos trabalhos de Fogel et al. (2002) e Fogel et al. (2004), foi usado a técnica de torneio (variando entre 6 a 10 indivíduos) para o processo de seleção. A taxa de elitismo foi variável, porém, sempre com valores pequenos, geralmente em torno de 1 a 10 indivíduos, dependendo do tamanho da população.

3.3.5 Cruzamento

Foram usados dois processos de cruzamento, o cruzamento uniforme (SYSWERDA, 1989) e o cruzamento de uma partição.

Com o cruzamento uniforme, poderia ocorrer uma certa probabilidade de ocorrer uma mutação em conjunto. Como o indivíduo é composto por “pares de bits” (cada base nitrogenada é composta por 2 bits, conforme explicado no tópico 3.3.1) e o cruzamento uniforme troca material genético de forma aleatória dos pais para gerar o filho, sendo assim, poderia ocorrer a possibilidade de um “par de bit” ser modificado, caracterizando uma mutação.

Para evitar este problema, usou-se o cruzamento de uma partição, onde cada pai “contribuiu” com a metade de seu gene. Para exemplificar este processo, dado dois pais i) TATATTAG e ii) TTGAAGGT, o filho resultante deste cruzamento seria: TATAAGGT.

3.3.6 Mutação

Para a mutação foi usado o processo de mutação uniforme (MICHALEWICZ, 1997). A taxa de mutação usada foi variável, sendo usada desde taxas pequenas (1%) até taxas maiores, em torno de 12,5% (usaram-se taxas maiores, com o objetivo de análise do comportamento do algoritmo).

3.3.7 Critérios de encerramento

No algoritmo desenvolvido por Fogel et al. (2004) o critério de encerramento era o número de gerações (500 gerações).

Neste trabalho foram usados as opções disponíveis no MatLab (2005), onde são disponibilizados duas opções por tempo, *Timelimit* e *Stalltimelimit*, sendo a primeira opção o tempo máximo em que o algoritmo será executado e a segunda opção é o tempo máximo que o algoritmo irá continuar a ser executado se não houver uma melhora do *fitness* do melhor indivíduo. Ambos os tempos são dados em segundos. Para ambas as opções, foram usados tempos suficientes para que ao menos 100 gerações fossem processadas.

Existem duas opções baseando-se no número de gerações: a *Generation* e *Stallgenlimit*. A primeira opção permite configurar qual é a quantidade máxima de gerações que a população irá ter. A segunda opção é a quantidade máxima de gerações a ser executada se não houver uma melhora do *fitness* do melhor indivíduo. Na maioria das execuções, usou-se pelo menos 100 gerações, sem limite para encerramento das gerações caso o melhor indivíduo não tivesse melhora no decorrer do processo.

3.3.8 Recursos utilizados

Para executar o GA_FIND_RR, foi utilizado o MatLab versão 7 (R14), sendo executado em um computador Pentium IV 2,8 Ghz com 756 Mb de memória RAM, e em um *Notebook* Acer, modelo Aspire 3523, com processador Intel Celeron 1,5 GHz com 1Gb de memória RAM. Ambos os computadores utilizavam o sistema operacional Windows XP SP2.

4 Experimentos e Discussão

Inicialmente foi desenvolvido uma versão "*beta-teste*" do GA_FIND_RR utilizando-se o organismo *Bacillus subtilis.* Foi gerada uma matriz de 3567 linhas por 300 colunas (para a criação da matriz, usou-se o RSATools para a extração dos dados). Nem todas as colunas continham 300 bases, pois existem regiões intergênicas no *Bacillus subtilis* com menos de 300 bases *upstream*. O total de bases *upstream* extraídas foram 485.657, sendo aproximadamente 33% de bases "A", 31% de bases "T", 20% de bases "G" e 16% de bases "C".

Após a preparação da base de dados, foram realizados uma série de execuções do algoritmo GA_FIND_RR, com o objetivo de analisar os resultados obtidos com a documentação disponível. Os parâmetros para a execução desta versão foram:

Parâmetros	**Configuração adotada**
População	10 – 200
Gerações	50 – 300
Taxa de Seleção	80%
Tipo de Seleção	Torneio
Cruzamento	Uniforme
Taxa de Mutação	2% - 20%
Tipo de Mutação	Uniforme
Tamanho do Indivíduo	24 Bits
Execuções Compiladas	11

Após as execuções, percebeu-se que a as seqüências preditas como sendo prováveis RR não estavam de acordo com as referências usadas como padrão de validação citadas na seção 5.2. Um exemplo de resultado obtido nestes testes pode ser analisado na tabela 5, na figura 18, observa-se a evolução do algoritmo durante as gerações.

Tabela 10 - Exemplo de resultado obtido pelo GA_FIND_RR, para a *Bacillus subtilis.*
Foram usados os parâmetros: PP=100; GR=100; TX=80%; TS=Uniforme; CZ=Uniforme; TM=10% ; TI=24 bits; EL=00**.** O melhor indivíduo repetiu-se 81 vezes no resultado final. Não foram encontradas referências para estas RR. O **Nx**, é a distância entre as bases, podendo variar entre 0 a 30. SI = Sem Informação e SR = Sem referência. A função de Fitness usada foi a contagem de freqüência de repetições.

Indivíduos	*Fitness*	Região Regulatória Completa do oligômero	Referências
AAAATA**Nx**AAAAAG	5	SI	SR
AAAACA**Nx**AAAAGG	4	SI	SR
AACATA**Nx**GAAAGG	3	SI	SR
AAAATA**Nx**GAAGGG	2	SI	SR
GAAGTG**Nx**AAAAGG	2	SI	SR
ACAATA**Nx**AAGAGA	2	SI	SR
ACAGTA**Nx**AAAAGG	1	SI	SR
AGAATA**Nx**CAAAAA	1	SI	SR
ACAGTA**Nx**AAAAGG	1	SI	SR
AAACTA**Nx**CAAGAT	1	SI	SR

Para facilitar a análise dos resultados, foi solicitado, via e-mail, ao Sr. Yuko Makita, um dos desenvolvedores da base disponível do site DBTBS (2006), um arquivo com os dados compilados das RR conhecidas do *Bacillus subtillis*. O Sr. Makita forneceu dois arquivos. Um em padrão .XML e outro em padrão .XLS.

Baseando-se nesses arquivos, conforme exemplo da tabela 6, foram usados os recursos do editor de texto Microsoft Word para localizar e realçar os oligômeros encontrados, sendo executadas as seguintes etapas para comparação dos resultados obtidos com o GA_FIND_RR:

- Execução do GA_FIND_RR, salvando o resultado da última geração em arquivo .TXT, juntamente com os parâmetros utilizados. Os gráficos foram salvos em formato .JPG;

- Selecionados os 10 melhores indivíduos da última geração;

- Para cada indivíduo selecionado, foram localizadas e realçadas em cores distintas as partes do oligômero na tabela fornecida;

- Quando as duas partes do oligômero ficavam entre as bases destacadas como RR, foi feita uma busca no site DBTBS (2006), através do dado contido na coluna *the first gene*, para buscar as referências bibliográficas da RR encontrada;

- Todos os resultados positivos foram tabulados, contendo as informações da região *upstream* analisada.

Para todas as versões desenvolvidas para o *Bacillus subtilis*, o processo de comparação dos resultados foi igual ao exposto acima. As outras referências citadas na seção 5.2 também foram usadas para confirmação dos resultados.

Tabela 11 - Amostra dos dados compilados do site DBTBS (2006).
A coluna "Binding sequence", contêm as informações das RR conhecidas para o *Bacillus subtilis*, as bases que estão entre "{}" são as RR propriamente ditas. A Coluna *Transcription factor* é o tipo de transcrição da RR e a coluna *the first gene* é o gene regulado pela RR referenciada.

Binding sequence	***Transcription factor***	***the first gene***
CG{GGAAACTT}TTTCAAAGTTTCATT{CGTCTA}CGATA/TA/TT/G/A	SigW	*abh*
CG{GGAAACTT}TTTCAAAGTTTCATT{CGTCTA}CGATA/TA/TT/G/A	SigX	*abh*
CTATTT{TTTTGTCTGTACAAAT}TACAGCA	AraR	*abnA*
CTATTTTTTTGTC{TGTACA}AATTACAGCATAGTGAC{TACAAT}AAAGGG/G/ATACCG	SigA	*abnA*
CAAAATGATTGACGATTATTGGAAACCTTG	AbrB	*abrB*
TCTTACAATCAA{TAGTAA}ACAAAATGATTGACGATTATTGGAAACCTT/G/TTATGCT	SigA	*abrB*
TAGTAAACAAAATGA{TTGACG}ATTATTGGAAACCT{TG}T{TATGCT}ATGAAG/G/TAAGGAT	SigA	*abrB*
ATTT{TGTCGAA}TAA{TGACGAA}GAAAAAT	Spo0A	*abrB*
{TGTAAGCGTTCATC}A	CcpA	*ackA*
GACTTCTTAT{TGTAAGCGTTATCA}ATACGCAAGT	CcpA	*ackA*

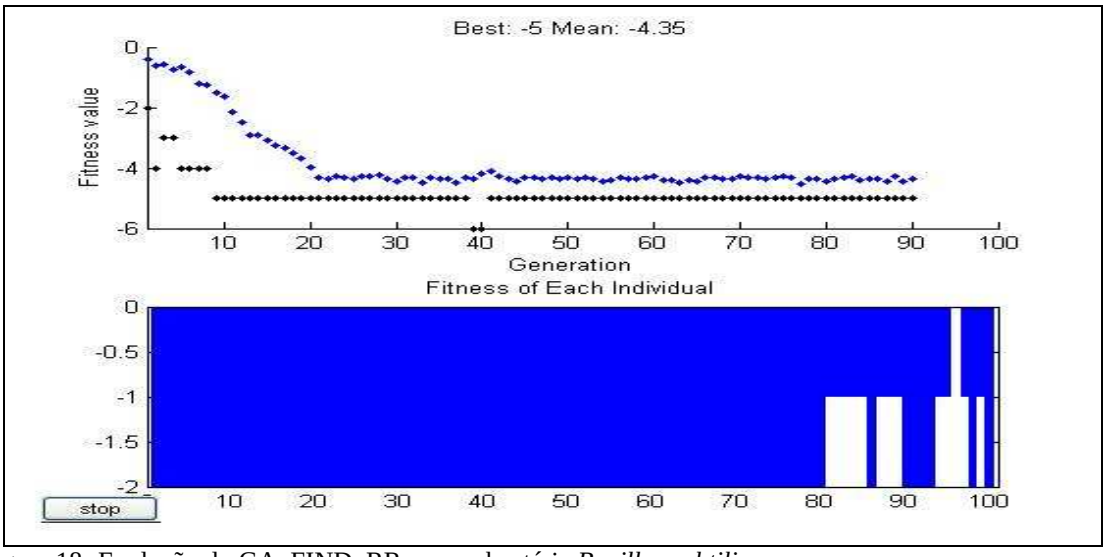

Figura 18- Evolução do GA_FIND_RR, para a bactéria *Bacillus subtilis.*
O primeiro gráfico representa a evolução da população, sendo o eixo X referente ao número da população e o eixo Y o "*score*" dos indivíduos, as bolas azuis representam a tendência de evolução da população e as bolas pretas o melhor indivíduo. O gráfico na parte de baixo representa o "*score*" de cada indivíduo da última geração, percebe-se que existe uma perda rápida de diversidade, onde a tendência de todos os indivíduos é ficarem com o mesmo "*score*".

Baseando-se nos resultados negativos que estavam ocorrendo até aquele momento, decidiu-se criar uma base de dados de teste. Esta continha uma pequena quantidade de bases e foi criado um oligômero artificial com as mesmas características de uma região regulatória,

ou seja, alta taxa de repetição e um distanciamento entre as duas partes do oligômero variando entre 0 e 30 bases.

A intenção de criar uma base de testes foi com intuito de validar se o GA_FIND_RR seria capaz de predizer a região artificialmente implantada na base. Consequentemente, poder-se-ia validar sua eficiência, e em caso de necessidade, seriam feitos ajustes no algoritmo para melhorar o seu grau de acerto.

6.1 Base de Testes

Foi criada uma base de testes com 744 bases, tendo 5 linhas com 315, 219, 146, 31 e 33 colunas respectivamente, conforme figura 19. Em cada linha foi colocado um oligômero contendo as bases "GATCTAYYYCGGTAA", onde o "YYY" foi propositalmente incluído para garantir a distância de 3 bases entre as duas partes do oligômero, ou seja, a sua representação pode se descrita como "GATCTA$\mathbf{N_3}$CGGTAA". A distribuição percentual das bases ficou com: 33% de bases "T", 30% de bases "A", 20% de bases "G" e 17% de bases "C".

GATCTAYYYCGGTAATTGTGACAACCATTGCAAGCTCTCGTTTATTTTGGTATTATATTTGTGT
TTTAACTCTTGATTACTAATCCTACCTTTCCTCTTTATCCACAAAGTGTGGATAAGTTGTGGAT
TGATTTCACACAGCTTGTGTAGAAGGTTGTCCACAAGTTGTGAAATTTGTCGAAAAGCTATTTA
TCTACTATATTATATGTTTTCAACATTTAATGTGTACGAATGGTAAGCGCCATTTGCTCTTTTTT
TGTGTTCTATAACAGAGAAAGACGCCTTTTCTAAGAAAAGGAGGGACGTGCCGGAAGA
CAGGACCGGGGATCAATCGGGGGATCTAYYYCGGTAAAAAGTGTGAATAACTTTTCGGAAGT
CATACAGATCTAYYYCAGTAACAGTCTGTCCACATGTGGATAGGCTGTGTTTCCTGTCTTTTTC
ACAACTTATCCACAAATCCACAGGCCCTACTATTACTTCTACTATTTTTTATAAATATATATATT
AATACATTATCCGTTAGGAGGATAAAAA
TCCGATACACTGCTGCCGACCCGTCGGCAGCTTTTCTATTCGGTATCTGCTCCGACAAGTTTT
CCCTTTCCCTAATTCGTTTTGATCTAYYYCGGTAATTTTTAGTACAATTAGATATTAGTGATAT
TTGAAAGAGGTCGATATAA
AGCGGGTGACACTGATGATCTAYYYCGGTAA
GATCTAYYYCGGTAAAGAAATGAGGTGAGCAAT

Figura 19 – Base de testes utilizada com o GA_FIND_RR.
Cada linha (região *upstream*) está representada por uma cor diferente, a seqüência, **GATCTAYYYCGGTAA**, em destaque, são as "regiões regulatórias" artificialmente implantadas.

Os primeiros testes foram executados fazendo com que o GA_FIND_RR, procurasse oligômeros com 12 bases (24 bits). Os resultados foram negativos, ou seja, todas as execuções do algoritmo falharam na tentativa de localizar as RR implantadas artificialmente. Os parâmetros utilizados para esta versão foram:

Parâmetros	Configuração adotada
População	10 – 1000
Gerações	10 – 1000
Taxa de Cruzamento	50% - 80%
Tipo de Seleção	Torneio
Cruzamento	1-partição e Uniforme
Taxa de Mutação	2% - 20%
Tipo de Mutação	Uniforme
Tamanho do Indivíduo	24 Bits

Analisando os resultados obtidos por Mwangi e Siggia (2003), percebeu-se que a tabela dos oligômeros mais representativos (chamado pelos autores de "TOP 10"), continha oligômeros com tamanho variando entre 4 e 5 bases, como $TTGAN_{19}TATA$. O trabalho realizado por Hu et al. (2005) concluiu que um dos fatores que diminuem a eficiência de predições das prováveis regiões regulatórias é o tamanho do oligômero usado como padrão de busca. Quanto maiores são os oligômeros analisados, menor é a eficiência dos algoritmos. Baseando-se nestes dados, resolveu-se executar o GA_FIND_RR utilizando-se indivíduos com 8 bases (16 bits). Os parâmetros utilizados foram:

Parâmetros	Configuração adotada
População	10 – 200
Gerações	10 – 1000
Taxa de Cruzamento	40% - 80%
Tipo de Seleção	Roleta e Torneio
Cruzamento	1-partição e Uniforme
Taxa de Mutação	2% - 40%
Tipo de Mutação	Uniforme
Tamanho do Indivíduo	16 Bits

Nas várias execuções do GA_FIND_RR, o algoritmo conseguiu localizar as regiões artificialmente implantadas na base. O resultado do algoritmo melhorava significativamente quando maior o número da população e maior o número de gerações. Com população de 100 indivíduos e 100 gerações, O GA_FIND_RR conseguia localizar, em média, 70% das execuções as RR artificialmente implantadas. Com populações e gerações maiores a taxa de acerto é incrementada, mas o tempo de execução do algoritmo também aumentava significativamente.

As RR eram computadas como corretas quando o GA_FIND_RR conseguia localizar variações dos oligômeros "GATCTA$\mathbf{N_3}$CGGTAA", como por exemplo: "TCTA$\mathbf{N_5}$GTAA" ou "GATC$\mathbf{N_6}$GGTA". Um exemplo dos testes executados pode ser observado na tabela 7 e a evolução dos indivíduos na figura 20.

Tabela 12 – Exemplo de um resultado obtido pelo GA_FIND_RR, para a *base de testes.*
Foram usados os parâmetros: PP=300; GR=150; TX=80%; TS=Uniforme; CZ=Uniforme; TM=25% ; TI=16 bits; EL=01. O melhor indivíduo repetiu-se 240 vezes no resultado final, sendo este o oligômero artificialmente implantado. O **Nx**, é a distância entre as bases, podendo variar entre 0 a 30. O resultado -5, são indivíduos considerados como não adaptados ao ambiente. Neste caso, são indivíduos cuja repetição de uma mesma base é igual ou superior a 3, ou que não foram encontradas referências na base analisada.

Indivíduos	***Fitness***
GATC**Nx**CGGT	5
AATT**Nx**TAGT	1
ATTC**Nx**CGGT	1
CACA**Nx**AGTG	1
TAGA**Nx**AGGT	1
TCAC**Nx**AAGT	1
AAAT**Nx**CGAG	-5
AACC**Nx**CTGG	-5
AACT**Nx**CAGT	-5
AAGT**Nx**CGCT	-5

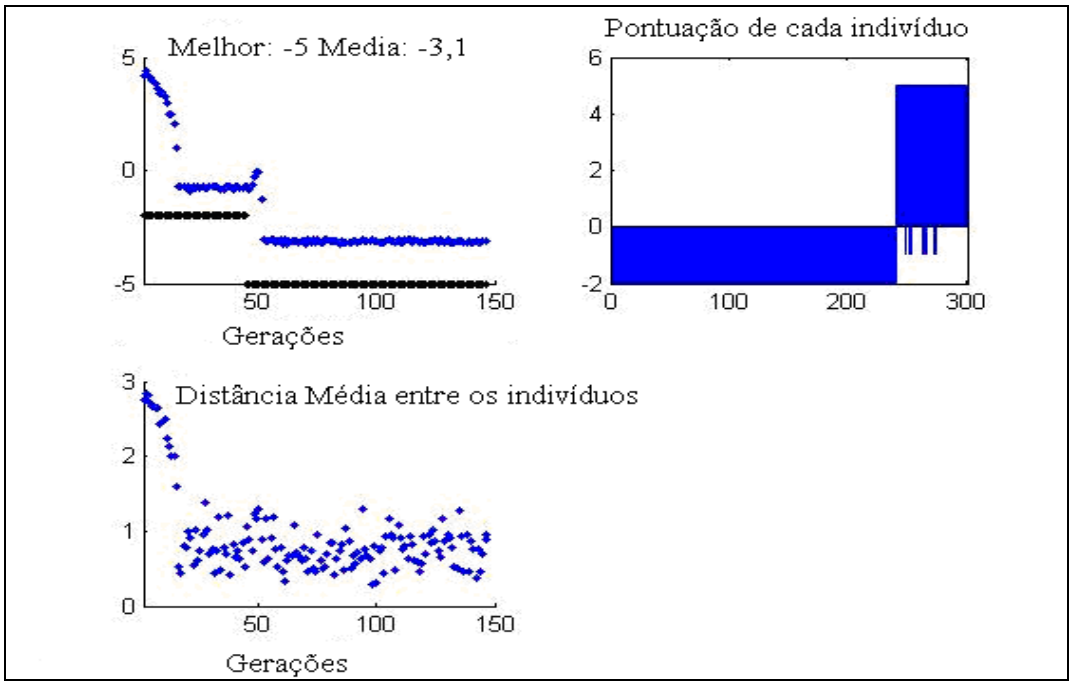

Figura 20 – Evolução do GA_FIND_RR, para base de testes.
O primeiro gráfico representa a evolução da população (a), tendo no eixo X o número da população e o eixo Y o *"score"* dos indivíduos. As bolas azuis representam à tendência de evolução da população e as bolas pretas o melhor indivíduo. O gráfico a direita (b) representa o *fitness* de cada indivíduo da última geração. O gráfico de baixo (c) representa a distância média entre os indivíduos.

Com os ensaios realizados na base de testes, pode-se concluir que o algoritmo GA_FIND_RR era capaz de predizer oligômeros com as características inerentes às regiões regulatórias, que são:

- Dois conjuntos de bases separados entre 1 a 30 bases;

- Repetem-se uma ou várias vezes dentro de um determinado organismo (super-representatividade);

As limitações encontradas referem-se principalmente ao tamanho do oligômero. Nos testes, só obteve-se sucesso com indivíduos de 16 bits e com populações superiores a 50 indivíduos. Com indivíduos de 24 bits o algoritmo não foi capaz de encontrar as RR mesmo com populações grandes (1000 indivíduos).

6.2 Tamanho das Bases de dados

Inicialmente foram extraídas as primeiras 300 bases *upstream*, em Mwangi e Siggia (2003) e posteriormente foram extraídos apenas as 100 primeiras bases *upstream* do organismo.

A decisão de usar as 100 primeiras bases *upstream* foi baseada nos trabalhos desenvolvidos por Helmann (1995) e por Jacques et al. (2006) que usaram 100 bases em seus experimentos e que comprovaram que a maior concentração de regiões regulatórias do *Bacillus subtilis* estava entre as posições -10 e -45.

Hu et al. (2005) fizeram uma análise dos algoritmos: MEME (BAILEY e ELKAN, 1995), AlignACE (ROTH et al., 1998), BioProspector (LIU et al., 2001), MDSCAN (LIU et al., 2002) e MotifSampler (THIJS et al. 2002), utilizados para localizar regiões regulatórias em procariontes. Esta análise demonstrou que quanto maior é a quantidade de bases *upstream*, menor é a eficiência dos algoritmos. Os autores também relatam que os melhores resultados obtidos concentram-se entre as bases -20 a -100.

Após diversas execuções do GA_FIND_RR, comprovou-se que trabalhar com 100 bases *upstream* resultaram nos melhores resultados do algoritmo.

6.3 Resultados do GA_FIND_RR para Base de dados com 100 colunas

Após os experimentos bem sucedidos usando a base de testes, decidiu-se usar os mesmos princípios em uma base de dados real. Sendo assim, foram extraídos as 100 primeiras bases *upstream* do *Bacillus subtilis*, usando a ferramenta RSATools, gerando uma matriz com 3567 linhas por 100 colunas. As linhas representam os genes existentes no organismo e as

colunas as bases *upstream.* O tamanho mínimo considerado de colunas foi 1 e o tamanho máximo 101, as médias dos tamanhos das colunas são de 75 bases. Os totais de bases extraídas foram: 270.961, sendo aproximadamente 35% de bases “A”, 30% de bases “T”, 21% de bases “G” e 14% de bases “C”.

Foi realizado um levantamento dos oligômeros de tamanho 4, na base analisada com a intenção de listar em ordem decrescente os oligômeros mais representativos, vistos na tabela 8. Foram encontrados 231 oligômeros, sendo desconsiderados na contagem os oligômeros com 3 ou 4 bases repetidas e sobreposições entre as bases. A intenção deste levantamento foi a de fazer uma analise com os resultados obtidos do GA_FIND_RR e verificar se os oligômeros de tamanho 4, listados na tabela 8, estavam presentes com freqüência nas RR encontradas.

Tabela 13 – Os 231 oligômeros, de tamanho 4, mais representativos do *Bacillus subtilis*.
A coluna cujo titulo é “**Olig**.” Representa os oligômeros de tamanho 4, encontrados no critério estabelecido e o título “**Ocor**.” É o número de ocorrências presentes na base analisada. Informações extraídas no RSATools.

Olig.	**Ocor.**	**Olig.**	**Ocor.**	**Olig.**	**Ocor.**	**Olig.**	**Ocor.**	**Olig.**	**Ocor.**
ATAA	2843	AATC	1176	CTGT	719	CTGC	463	GTCC	259
AATA	2630	TAAG	1165	ACTA	709	ACGG	461	CGTC	257
AGGA	2505	ATAC	1162	GCAT	705	ACGT	458	CTAC	244
AAGG	2448	GTAT	1152	GCTG	704	AGCC	455	CACG	242
TGAA	2400	GGAT	1097	AGCT	697	CACT	453	GGCC	235
GGAG	2367	GATG	1060	CAGG	689	GTGC	453	GCCC	234
TTAT	2364	TGGA	1043	AGGC	687	GGAC	450	TCGC	234
AGAA	2315	GACA	1013	ATGC	684	TGCC	447	GACC	232
TATA	2187	GTTA	1008	CTTG	683	GTGG	442	CGCC	228
ATAT	2179	AAGC	1006	CGTT	675	GTAC	441	CTCG	214
TATT	2134	AAGT	1006	TAGT	673	TCGA	441	GTCG	189
GAGG	2128	TTAC	997	TCCT	673	CGTA	433	CGAC	177
ATGA	2105	CTAT	992	TCTG	673	TACG	432	CCAC	169
AAGA	2092	ATGG	986	GCAG	664	AGTC	419	CGCG	156
AATT	2017	GCTT	969	TGGT	656	GGCG	406		
GGAA	1871	GGTG	967	AGAC	654	CGCT	404		
AATG	1857	TATC	953	GTTG	653	GCCT	401		
TTAA	1815	GGTT	939	GGCA	647	GCGA	397		
AACA	1754	TGAG	937	ACGA	642	CTGG	394		
GAAT	1720	ACAG	928	TTCG	638	TGGC	392		
TAAT	1685	ACTT	928	CACA	632	CCTA	390		
TTCA	1653	AGTT	911	CTTC	626	CCTG	390		
ACAA	1650	TTAG	901	CATC	624	CCCT	389		
TGAT	1647	CTTA	882	TGTC	624	TACC	388		
TTGA	1628	ATCT	868	AACC	621	TAGC	379		
TGTT	1577	AACT	860	GTCA	616	CGGT	378		
GATA	1568	CAGA	859	GATC	602	TCGG	378		

Tabela 8 - Continuação

ATTA	1532	TAAC	850	AGCG	598	GCCA	377
TTCT	1471	CCTT	834	GTTC	598	GCCG	377
CATA	1469	TCTA	834	TCTC	595	GACG	369
TCAT	1461	AGCA	830	CAGC	593	TGCG	369
GAGA	1457	AGTA	828	ACTG	592	CGGC	368
GAAG	1454	TGCT	812	CCAT	585	GTAG	365
TATG	1450	TAGG	809	CGGA	583	TCCG	365
ATAG	1435	TTGG	805	GGCT	581	CTAG	350
TTGT	1429	GAAC	804	TCAC	563	CTCC	345
ATCA	1428	CTGA	803	GTCT	559	GCGT	345
ATTG	1416	GCAA	800	GAGC	557	CCGG	340
TCTT	1378	AACG	799	CCAA	555	CCGT	338
ACAT	1377	TGAC	798	CGAT	553	GCTC	334
GATT	1347	AGTG	790	ATCG	546	CCTC	333
CATT	1326	CTAA	782	ATCC	545	ACTC	326
ATGT	1322	TGTG	767	ACAC	544	CGCA	326
TACA	1321	GGTA	765	GCGG	541	GCAC	322
TCAA	1301	TCAG	757	CTCT	538	CGTG	321
AGGT	1281	TGCA	751	TCGT	531	CCGA	318
TGTA	1272	CATG	750	CTCA	519	ACCG	316
GTAA	1265	GAGT	750	CAGT	512	CCGC	314
AGAT	1264	TTGC	743	GCTA	506	CACC	307
AGAG	1255	TACT	736	ACCA	505	GGTC	304
GTGA	1255	CAAG	732	TCCA	505	CCAG	299
TAGA	1213	CGAA	732	CAAC	494	CGAG	286
CAAT	1196	TTCC	730	GACT	482	ACGC	281
ATTC	1185	GTGT	722	ACCT	465	GCGC	265

Foram usadas três formas distintas de busca e duas formas distintas para calcular a função de adaptação, conforme mencionado na seção 5.3.2.

6.4 Versão 1

Após o oligômero já estar separado em duas partes distintas, chamando estas partes, respectivamente de *dimer_1* e *dimer_2, a* versão 1 baseou-se no seguinte critério de busca e função de adaptação:

- Buscada a primeira ocorrência encontrada, tanto para o *dimer_1* como para o *dimer_2*;

- Comparado se o *dimer_1* está antes do *dimer_2* e se as distâncias entre eles estão entre 0 e 30 bases;

- A cada ocorrência verdadeira em relação aos critérios do tópico descrito acima, foi somado o valor de 1 na matriz "s*uper"*, na posição relativa à distância entre as bases;

- O maior valor armazenado na matriz *super* era o *fitness* do indivíduo. A função de adaptação foi representada como: ***fa = super[1, Mv]) * (-1).***

Após as execuções, observou-se que o GA_FIND_RR, foi capaz de localizar seqüências similares as apresentadas nos trabalhos usados como bases de comparação.

A seqüência mais freqüentemente localizada pelo GA_FIND_RR, nesta versão, foi TTGA$\mathbf{N_x}$AATA, com distâncias médias ($\mathbf{N_x}$) de 16 a 24 bases, sendo muito similar a seqüência de consenso (***TTGA***CA$\mathbf{N_{\sim 16}}$TA***AAT***) observada por Helmann (1995). Esta seqüência também tem bastante similaridade com alguns oligômeros encontrados no trabalho de Mwangi e Siggia (2003) que estão relacionados na tabela "*top 10*" dos oligômeros mais freqüentes encontrados pelo algoritmo desenvolvido por estes autores.

6.5 Versão 2

Após o oligômero já estar separado em duas partes distintas, chamando estas partes, respectivamente em *dimer_1* e *dimer_2,* a versão 2 baseou-se no seguinte critério de busca e função de adaptação:

- Buscada a primeira ocorrência encontrada para o *dimer_1*;

- Buscada uma ocorrência para o *dimer_2,* em uma posição maior que o *dimer_1*, cuja distância fique entre 0 e 30 bases;

- A cada ocorrência verdadeira em relação aos critérios do tópico descrito acima, foi somado o valor de 1 na matriz "*super"*, na posição relativa à distância entre as bases;

- O maior valor armazenado na matriz *super* era o *fitness* do indivíduo. A função de adaptação foi representada como: ***fa = super[1, Mv]) * (-1).***

Após as primeiras execuções observou-se que o GA_FIND_RR não era capaz de localizar seqüências similares às apresentadas nos trabalhos usados como base de comparação.

O oligômero ATAAN$_x$GGAG, encontrado com bastante freqüência nas primeiras execuções do GA_FIND_RR não é considerado uma RR (JACQUES et al., 2006). Este oligômero possui as principais características de uma RR (seqüência super-representativa e distância entre 0 e 30 bases). A seqüência ATAA é considerada uma região de consenso na posição -10 da RR TTGAN$_x$ATAA, e a seqüência GGAG é uma parte do oligômero conhecido como *Ribossome Binding Site* (RBS), sendo esta seqüência considerada como consenso, quando está próxima do início de um gene (JACQUES et al., 2006).

A seqüência completa de consenso da RBS é AGGAGC, também conhecida como seqüência de Shine-Delgarno. “Esta região é um dos elementos mais importantes para o processo de tradução genética” (SHINE e DELGARNO, 1974).

O algoritmo proposto por Makita et al. (2007) utiliza-se da localização desta região para predição dos locais de início do processo de tradução em procariontes. Portanto, mesmo que a seqüência ATAAN$_x$GGAG possua as principais características de uma RR, biologicamente ela deve ser descartada dos possíveis candidatos. No artigo escrito por Terai et al. (2001), conclui-se que *clusters* com alto *score* são gerados devido à região Shine-Delgarno, mesmo não sendo considerado o maior obstáculo nas predições de RR, esta área acaba por gerar resultados falsos que devem ser ignorados.

Uma solução para descartar esta região pelo GA_FIND_RR é simplesmente desconsiderar as posições 0 a -20 e extrair as regiões *upstream* com a utilização do RSATools a partir da posição -21 até -121. Porém, podem ocorrer “perdas” de indícios de RR cuja seqüência tem bases entre as posições -20 e 0.

Para solucionar o problema, foram descartadas as bases na segunda parte do oligômero (*dimer_2*) que continham combinações da seqüência AGGAGC (AGGA, GGAG e GAGC), eliminando a rápida convergência do GA_FIND_RR para esta região. Este procedimento de descarte também foi usado para as demais versões desenvolvidas para o GA_FIND_RR.

6.6 Versão 3

Após o oligômero já estar separado em duas partes distintas, chamando estas partes, respectivamente de *dimer_1* e *dimer_2*, a versão 3 baseou-se no seguinte critério de busca e função de adaptação:

- Buscada a primeira ocorrência encontrada para o *dimer_1*;

- Buscada uma ocorrência para o *dimer_2*, em uma posição maior que o *dimer_1*, cuja distância fique entre 0 e 30 bases;

- A cada ocorrência verdadeira em relação aos critérios do tópico descrito acima, foi somado o valor de 1 na matriz *super*, na posição relativa à distância entre as bases;

- Calculado a média aritmética de todos os valores da matriz super, cujo valor seja maior que zero. A função de adaptação foi representada como: ***fa = (Σ (super[Nx1..Nx30]) / cont) * (-1).***

Após as execuções, observou-se que o GA_FIND_RR foi capaz de localizar seqüências similares as apresentadas nos trabalhos usados como base de comparação.

6.7 Versão 4

Após o oligômero já estar separado em duas partes distintas, chamando estas partes, respectivamente de *dimer_1* e *dimer_2, a* versão 4 baseou-se no seguinte critério de busca e função de adaptação:

- Buscada a primeira ocorrência encontrada para o *dimer_2*;

- Buscada uma ocorrência para o *dimer_1*, em uma posição menor que o *dimer_2*, cuja distância fique entre 0 e 30 bases;

- A cada ocorrência verdadeira em relação aos critérios do tópico descrito acima, foi somado 1 na matriz *super*, na posição relativa à distância entre as bases;

- O maior valor armazenado na matriz *super* será o *fitness* do indivíduo. A função de adaptação é representada como: ***fa = super[1, Mv]) * (-1)***.

Após as primeiras execuções e antes de descartar as seqüências do *dimer_2* com combinações pertencentes ao RBS (AGGAGC) , observou-se que o GA_FIND_RR obteve resultados muito similares aos da versão 2. Este fato comprovou que a seqüência RBS (AGGAGC) é extremamente freqüente nas posições próximas ao início de um gene, devendo ser descartada em algoritmos de busca por RR minimizando resultados falsos. Nas proximidades desta região (menos de 30 bases), existe a presença muito constante do oligômero “ATAA”, como pode ser observado nos trabalhos realizados por Helmann (1995) e

Jacques et al. (2006), levando desta maneira o GA_FIND_RR a uma rápida convergência para este ponto de busca, o que na realidade é indesejado.

O objetivo de criar esta versão, invertendo as partes do oligômero na busca, ou seja, buscando-se primeiramente o "*dimer_2*" e depois "aproximando" o "*dimer_1*" do "*dimer_2*", foi com a intenção de comprovar que era necessário o descarte de bases próximas ao início do TSS, cujas seqüências do oligômero sejam combinações da seqüência RBS.

Após o descarte das seqüências do RBS, o GA_FIND_RR foi capaz de localizar RR utilizando-se desta versão.

6.8 Versão 5

Após o oligômero já estar separado em duas partes distintas, chamando estas partes, respectivamente de *dimer_1* e *dimer_2,* a versão 5 baseou-se no seguinte critério de busca e função de adaptação:

- Buscada a primeira ocorrência encontrada para o *dimer_2*;
- Buscada uma ocorrência para o *dimer_1,* em uma posição menor que o *dimer_2*, cuja distância fique entre 0 e 30 bases;
- A cada ocorrência verdadeira em relação aos critérios do tópico descrito acima, foi somado o valor de 1 na matriz *"super"*, na posição relativa à distância entre as bases;
- Calculada a média aritmética de todos os valores da matriz *super*, cujo valor seja maior que zero. A função de adaptação foi representada como: ***ƒa = (Σ (super[Nx1..Nx30]) / cont) * (-1)***.

Após as execuções, observou-se que o GA_FIND_RR foi capaz de localizar seqüências similares as apresentadas nos trabalhos usados como base de comparação.

6.9 Compilação dos Resultados

A tabela 9 é um resumo dos principais parâmetros utilizados nas execuções do GA_FIND_RR. A tabela 10 contém a compilação dos principais oligômeros considerados "super-representados" pelo GA_FIND_RR e que estão documentados nas referências

utilizadas como padrão de comparação. Nesta tabela estão sendo considerados todos os oligômeros independentes da versão utilizada.

Tabela 14 - Principais parâmetros usados para a execução do GA_FIND_RR

Parâmetros	**Configuração adotada**
População	50 – 300
Gerações	50 – 300
Taxa de Seleção	50% - 80%
Tipo de Seleção	Torneio
Cruzamento	1-partição e uniforme
Taxa de Mutação	1% - 25%
Tipo de Mutação	Uniforme
Tamanho do Indivíduo	16 Bits
Execuções Compiladas	90

Tabela 15 - Compilação dos *motifs* com referência na literatura.

Nesta tabela estão sendo considerados os oligômeros mais "super-representados", independente da versão utilizada. As bases em vermelho são as seqüências completas da RR. O **Nx**, é a distância entre as bases, podendo variar entre 1 a 30. **FT** é o Fator de Transcrição. **Gene** é o primeiro gene da RR. **Reg** é o tipo de regulação (A = Ativador, P = Promotor e R = Repressor). **MF** é o maior *fitness* encontrado na tabela "super" (super[1..30], sendo representado como XX (FF), onde XX é a posição na tabela "super" e FF é a freqüência encontrada. **PO** é o *ranking* de 1º a 30º do oligômero na tabela "super" referenciado como RR, quanto menor é o número, maior é o seu *ranking*, sendo representado como RA (FF), onde RA é o *ranking* e o FF é a freqüência encontrada. **VR** foi a versão utilizada do GA_FIND_RR.

Indivíduos	**FT**	**Gene**	**Reg**	**Região Regulatória Completa do oligômero**	**MF**	**PO**	**VR**
AACA**N**$_{17}$ATAA	SigE	ypqA	P	TTTATAACAACATCTGGCATAGACGCATAATCTG GTTAAAAAAGGCGGT[1]	10 (17)	21º (09)	4
AACA**N**$_{17}$ATAA	SigK	ypqA	P	TTTATAACAACATCTGGCATAGACGCATAATCTG GTTAAAAAAGGCGGT[2]	10 (17)	21º (09)	4
AATA**N**$_{04}$ACAA	YdiH	ldh	R	ATATAAAT**G**TGAAATACTTCACAAACAAAAAGAC ATCAAAGAGAAACATACCCT[3]	10 (17)	13º (12)	5
AATA**N**$_{19}$ACAA	SigF	yuiC	P	TTTTGAATAATGCTCTCTCCACTTGGGAACAATG ATTCGGAGGAGAGGTGAATG[4]	10 (17)	15º (11)	5
AATA**N**$_{19}$ATAA	SigG	yvaB	P	GACAGAATAATCATTATGCATCTGTATGATAATA ATTGATGTGTGATTTTTAAAAACGAAAGGGCTGG TAAAAATG[5]	03 (27)	30º (07)	5
AATA**N**$_{09}$ATAG	YrzC	cysK	R	GATTATATACATAATACCAATACAAATAGTCGGA AATTGAGGTGTCGAGA[6]	09 (18)	01º (18)	4
AATA**N**$_{19}$ATGA	SigE	glgB	P	CTTCGAATAAATACTATAAATGAAAACTATGATG TCAGAAAGG[7]	10 (22)	25º (11)	5
AATA**N**$_{19}$ATGA	SigA	menE	P	GCTGTTTTCTTTTCAATACAGACATTTTACCTCG GAGATGATGACATGCTGACAGAACAGCCCAAC[8]	10 (22)	25º (11)	5

[1] Eichenberger et al. (2004)
[2] Eichenberger et al. (2004)
[3] Larsson et al. (2005)
[4] Wang et al. (2006)
[5] Wang et al. (2006)
[6] Even et al. (2006)
[7] Hay et al. (1986)
[8] Driscoll e Taber (1992)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
AATAN_{19}ATGA	SigE	ykvU	P	AATAAAATAATTTTTGAACTTGTCTCATATGATGT TGGTAGTACAAG[9]	10 (22)	25º (11)	5
AATAN_{04}TTAG	YrzC	ydbM	R	AAAAATGGCAGGAAATCTATAATACATATTAAAT TTATCGGAATTAAAACTGGGGGGCTGCCGG[10]	16 (14)	14º (07)	4
AATG$_{03}$ATAA	PerR	hemA	R	TTCTATGTTAGAATGATTATAAATTAAGATTGGG TGTTGGGG[11]	02 (22)	30º (04)	5
ACAAN_{16}ATAA	SigK	ydgB	P	CAAGGAACAATTGGGTGCAGCGGCGCATAATGT ACTGTACAGAAAGATGGAA[12]	15 (18)	06º (14)	5
ACAAN_{07}ATAA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[13]	15 (18)	05º (15)	5
ACTAN_{02}AATA	YrzC	yxeK	R	ATTCATATTAAACGACTAGGAATATAGGAGTTTA TTTTTCGCATT[14]	01 (18)	03º (08)	2
AGAAN_{05}ATAA	PerR	hemA	R	TTCTATGTTAGAATGATTATAAATTAAGATTGGG TGTTGGGG[15]	04 (18)	08º (15)	2
AGAAN_{19}ATAA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[16]	04 (18)	14º (13)	2
ATAAN_{04}AATG	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[17]	19 (19)	09º (14)	5
ATAAN_{15}AATT	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[18]	19 (28)	10º (16)	5
ATAAN_{06}AATT	Fur	feuA	R	CTATAATTCCAATTGATAATAGTTATCAATTGAAC AGGAGGCTCTATAGA[19]	19 (28)	27º (06)	5
ATAAN_{21}ACAA	SigG	gerBA	P	TTTTCCTCGATAAGAATAATTCTCCTTTTTTGATA CAAATTAATAAAAACCGTC[20]	05 (22)	13º (12)	2
ATAAN_{05}ACAA	PerR	PerR	R	TTATAAACATTACAATGTAAGAA[21]	05 (22)	01º (22)	2
ATAAN_{13}AGAA	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[22]	01 (26)	24º (14)	5
ATAAN_{13}ATAA	SigE	dacB	P	TTATTCATAACTGATGGACATGCGCATAAACTTG TACAAACCA[23]	01 (29)	02º (27)	5
ATAAN_{20}ATAA	SigF	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[24]	01 (29)	07º (21)	5
ATAAN_{20}ATAA	SigG	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[25]	01 (29)	07º (21)	5
ATAAN_{17}ATAA	SigG	sspC	P	GCGTGTATAAATTAAAATAATCTCTCCATAATAT GATTCAAACAAG[26]	01 (29)	27º (14)	5
ATAAN_{21}ATAA	SigD	tlpC	P	CTAAAATAAAACTTTAAACCCAAAAACCCGATAA GTAATATGACCTGC[27]	01 (29)	07º (21)	5
ATAAN_{02}ATAA	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[28]	01 (29)	10º (20)	5
ATAAN_{16}ATAG	SigE	yteV	P	TCTATCATAACGCTGTTCCAAACGGAATAGATTG ATAGAGAAAG[29]	09 (14)	07º (11)	2

[9] Eichenberger et al. (2003)
[10] Even et al. (2006)
[11] Herbig e Helmann (2001)
[12] Reischl et al. (2001)
[13] Yoshida et al. (2004)
[14] Even et al. (2006)
[15] Herbig e Helmann (2001)
[16] Yoshida et al. (2004)
[17] Baichoo et al. (2002) e Ollinger et al. (2006)
[18] Herbig e Helmann (2001)
[19] Baichoo et al. (2002) e Ollinger et al. (2006)
[20] Corfe et al. (1994)
[21] Fuangthong et al. (2002)
[22] Herbig e Helmann (2001)
[23] Simpson et al. (1994)
[24] Schuch e Piggot (1994)
[25] Schuch e Piggot (1994)
[26] Nicholson et al. (1989)
[27] Hanlon et al. (1994)
[28] Baichoo et al. (2002) e Ollinger et al. (2006)
[29] Henriques et al. (1997)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
ATAA**N$_{16}$**ATGA	SigE	ykvU	P	AATAAAATAATTTTTGAACTTGTCTCATATGATGT TGGTAGTACAAG[30]	09 (14)	07º (11)	2
ATAA**N$_{05}$**ATGA	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[31]	09 (14)	15º (09)	2
ATAA**N$_{20}$**ATTA	SigE	spoIVB	P	CAGTTATAAATAAGCCGTCAGAAGGCAAAATTAA ATGATGTA[32]	02 (22)	13º (11)	3
ATAA**N$_{20}$**ATTA	SigF	spoIVB	P	CAGTTATAAATAAGCCGTCAGAAGGCAAAATTAA ATGATGTA[33]	02 (22)	13º (11)	3
ATAA**N$_{27}$**GTAA	PurR	ytiP	R	CAAATAAAACGAATAATATTAATGGTGTTTTGTTA AAACGTTCGTAATTGGAGG[34]	03 (25)	03º (14)	5
ATAA**N$_{03}$**TTAC	perR	perR	R	TTATAAACATTACAATGTAAGAA[35]	09 (16)	03º (11)	4
ATAA**N$_{03}$**TTAT	Fur	feuA	R	CTATAATTCCAATTGATAATAGTTATCAATTGAAC AGGAGGCTCTATAGA[36]	08 (25)	03º (11)	2
ATAA**N$_{03}$**TTAT	PerR	katA	R	CTATTTTATAATAATTATAAAATAATATTGACTTTT TACTTAGAGATGATATTATGTT[37]	08 (25)	03º (21)	2
ATAA**N$_{07}$**TTAT	CcpC	ccpC	R	GGGAGATAAGAAAAACTTATTGATA[38]	08 (25)	18º (11)	2
ATAA**N$_{03}$**TTCT	Fur	fhuD	R	GTGGTATAATCACAGATGATAATGATTCTCTTTT TCATCTATCTTTTAGA[39]	08 (12)	25º (04)	2
ATAA**N$_{01}$**TTCT	LexA	yqjW	R	AAAAGCGAACATAAGTTCTTTTTA[40]	08 (12)	03º (01)	2
ATAG**N$_{16}$**ATAA	SigE	ycgF	P	TTGTGCATAGCTTGGCCCGTTCCCGAATAAATT GTACAAGTTACAT[41]	01 (22)	20º (09)	2
ATAG**N$_{20}$**ATAA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[42]	01 (22)	17º (10)	2
ATAG**N$_{07}$**ATAG	YrzC	yrrT	R	GTTCCATCATTAATCCATAGTATACTTATAGGAA TTATTAAATATGGAGT[43]	09 (13)	23º (03)	3
ATAG**N$_{08}$**ATCA	YdiH	alsS	R	AAAGAGTGTATAGTGAAACTTATCACAAGATATT TA[44]	08 (12)	01º (12)	2
ATAG**N$_{02}$**ATCA	Fur	feuA	R	CTATAATTCCAATTGATAATAGTTATCAATTGAAC AGGAGGCTCTATAGA[45]	08 (12)	13º (05)	2
ATAG**N$_{02}$**ATCA	CcpA	hutP	R	GTTAATAGTTATCA[46]	08 (12)	13º (05)	2
ATAG**N$_{22}$**TAAT	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[47]	22 (14)	01º (14)	4
ATAG**N$_{06}$**TTAT	YdiH	alsS	R	AAAGAGTGTATAGTGAAACTTATCACAAGATATT TA[48]	10 (10)	12º (08)	4
ATAG**N$_{05}$**TTAT	YrzC	yrrT	R	GTTCCATCATTAATCCATAGTATACTTATAGGAA TTATTAAATATGGAGT[49]	10 (10)	24º (05)	4
ATAG**N$_{19}$**TTCT	SigG	sspA	P	TTCTGAATGAAGCCATGTGTTTTGACACATTCTA TACTCACAAGGAGGTGA[50]	08 (12)	08º (05)	4
ATAG**N$_{06}$**TTCT	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[51]	08 (12)	26º (02)	4

[30] Eichenberger et al. (2003)
[31] Baichoo et al. (2002) e Ollinger et al. (2006)
[32] Gomez e Cutting (1996)
[33] Gomez e Cutting (1996)
[34] Saxild et al. (2001)
[35] Fuangthong et al. (2002)
[36] Baichoo et al. (2002) e Ollinger et al. (2006)
[37] Herbig et al. (2001)
[38] Kim et al. (2002)
[39] Baichoo et al. (2002)
[40] Au et al. (2005)
[41] Eichenberger et al. (2003)
[42] Yoshida et al. (2004)
[43] Even et al. (2006)
[44] Reents et al. (2006)
[45] Baichoo et al. (2002) e Ollinger et al. (2006)
[46] Wray e Fischer (1994)
[47] Yoshida et al. (2004)
[48] Reents et al. (2006)
[49] Even et al. (2006)
[50] Nicholson et al. (1989)
[51] Baichoo et al. (2002) e Ollinger et al. (2006)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
ATAT**N**$_{18}$AGAT	SigF	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[52]	05 (14)	20º (05)	3
ATAT**N**$_{18}$AGAT	SigG	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[53]	05 (14)	20º (05)	3
ATAT**N**$_{20}$ATAA	SigF	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[54]	01 (24)	02º (23)	4
ATAT**N**$_{20}$ATAA	SigG	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[55]	01 (24)	02º (23)	4
ATAT**N**$_{16}$ATAT	SigE	spoIVFA	P	TTCTTGACTAAACCGAATATTTGCCATGGACAAG ACATATGATGTACAAACC[56]	01 (24)	10º (13)	3
ATAT**N**$_{16}$ATAT	SigE	ydcC	P	GTCTGCATATTAGGGAAACCCCACTCATATATTT GATAGTGCATTAAGG[57]	01 (24)	10º (13)	3
ATAT**N**$_{26}$ATAT	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[58]	01 (24)	10º (13)	3
ATAT**N**$_{11}$ATCA	Fur	ykuN	R	AAAGTGATACATATGATATTGAAAATCATTATCA ACTAATGG[59]	17 (16)	15º (09)	4
ATAT**N**$_{19}$GATA	SigF	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[60]	06 (14)	07º (10)	2
ATAT**N**$_{19}$GATA	SigG	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[61]	06 (14)	07º (10)	2
ATAT**N**$_{16}$GATA	SigF	sigG	P	GCAGTGCATATTTTTCCCACCCAAGGAGATACTT AACGTTGTACAGCAGCTCC[62]	06 (14)	07º (10)	2
ATAT**N**$_{16}$GATA	SigG	sigG	P	GCAGTGCATATTTTTCCCACCCAAGGAGATACTT AACGTTGTACAGCAGCTCC[63]	06 (14)	07º (10)	2
ATAT**N**$_{22}$GATA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[64]	06 (14)	06º (11)	2
ATAT**N**$_{17}$TATA	SigE	ydcC	P	GTCTGCATATTAGGGAAACCCCACTCATATATTT GATAGTGCATTAAGG[65]	02 (32)	17º (12)	2
ATAT**N**$_{16}$TATA	SigE	yuzC	P	TTTGTCATATTCGGCAATTAGGGATCTATACATA TAGAAACATCCTTTTT[66]	02 (32)	04º (17)	2
ATAT**N**$_{16}$CATA	SigE	nucB	P	TTAAAAATATTCTTCATCAAGCGCCCATACATTG AAATGAACAAA[67]	03 (16)	19º (06)	2
ATAT**N**$_{15}$CATA	SigE	cwlD	P	GTAATCATATTTCCCGACCCTGTCCCATAGTTAT GTAATAACGGACAAG[68]	03 (16)	05º (12)	2
ATAT**N**$_{15}$CATA	SigE	ybaN	P	TCGGTTATATTCAATTGTCCATGCTCATAAGATG TAAAACAAGA[69]	03 (16)	05º (12)	2
ATAT**N**$_{15}$CATA	SigE	ydcC	P	GTCTGCATATTAGGGAAACCCCACTCATATATTT GATAGTGCATTAAGG[70]	03 (16)	05º (12)	2
ATAT**N**$_{19}$CATA	SigE	ytvI	P	GTATTCATATTCAGCCGCAGCGTGAATACATATA AAAAATAGGACAT[71]	03 (16)	11º (08)	2
ATGA**N**$_{20}$AATA	SigF	gpr	P	TTTAGCATGATTTATTCAGCAAATGGCAACAATA TAGGTACT[72]	03 (17)	13º (12)	5

[52] Feavers et al. (1990)
[53] Feavers et al. (1990)
[54] Feavers et al. (1990)
[55] Feavers et al. (1990)
[56] Cutting et al. (1991)
[57] Eichenberger et al. (2003)
[58] Yoshida et al. (2004)
[59] Baichoo et al. (2002)
[60] Feavers et al. (1990)
[61] Feavers et al. (1990)
[62] Sun et al. (1991)
[63] Sun et al. (1991)
[64] Yoshida et al. (2004)
[65] Eichenberger et al. (2003)
[66] Eichenberger et al. (2003)
[67] Sinderen e Venema (1995)
[68] Sinderen e Venema (1995)
[69] Eichenberger et al. (2003)
[70] Eichenberger et al. (2003)
[71] Eichenberger et al. (2003)
[72] Sussman e Setlow (1991)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
ATGA**N$_{20}$**AATA	SigG	gpr	P	TAGCATGATTTATTCAGCAAATGGCAACAATAT[73]	03 (17)	13º (12)	5
ATGA**N$_{16}$**AATA	SigE	yngJ	P	CAGGGAATGATTATAGAACTCGCCTAATAGGAT GTTACAAAGATGTGAA[74]	03 (17)	03º (14)	5
ATGA**N$_{02}$**ATAA	PerR	hemA	R	TTCTATGTTAGAATGATTATAAATTAAGATTGGG TGTTGGGG[75]	03 (17)	21º (10)	2
ATGA**N$_{10}$**TGAA	CssR	cssR	A	TGGGATAAAAATGAAAAGAATATGTGAAATTATG AAAA[76]	14 (22)	11º (14)	1
ATGT**N$_{18}$**ATAG	SigE	yyaD	P	TATGGCATGTTTGCTTTCCTTTATTTATATAGTAA CAATAACGGG[77]	04 (09)	08º (07)	3
CATA**N$_{18}$**ATAA	SigE	ycgF	P	TTGTGCATAGCTTGGCCCGTTCCCGAATAAATT GTACAAGTTACAT	04 (19)	04º (12)	4
CTAA**N$_{16}$**ATAA	SigE	bofA	P	AGTGGTCTAAACTCCTGGATCTTCTCATAAGCTT GTACTAG[78]	02 (15)	09º (09)	2
CTAA**N$_{02}$**ATAA	perR	perR	R	TTACACTAATTATAAACATTACAATG[79]	02 (15)	01º (15)	2
CTAT**N$_{18}$**ATAC	SigE	ytxC	P	TTACGTCTATTTTAAAAACATCCCCCATATACTT GTAACAGATGCCG[80]	01 (10)	21º (3)	3
CTAT**N$_{16}$**ATAC	SigE	yunB	P	CTATTACTATGTCCCCTCTTACAAGCATACATTG TGATATGTAAGGGGG[81]	01 (10)	10º (5)	3
GACA**N$_{17}$**AATA	SigA	ahpC	P	TATGGCTTGACAAAAAATATATATTAATTAATAAT TCATATATAATT[82]	01 (14)	21º (05)	2
GACA**N$_{19}$**AATA	SigA	htpG	P	ATCTAATTGACAATTGTCATCTTATGTGATAAATA GATGCTGAAAA[83]	01 (14)	07º (10)	2
GAGA**N$_{18}$**ATAA	SigA	acsA	P	GGTTTATATTTTAAAAATTGAGAAGAATATGAATA TATACTATAATAATTGTGACAACTTCAGCAAAGG G[84]	03 (14)	25º (03)	4
GATA**N$_{22}$**AATC	Fur	ykuN	R	AAAGTGATACATATGATATTGAAAATCATTATCA ACTAATGG[85]	01 (12)	12º (05)	2
GATA**N$_{22}$**TTAT	CssR	cssR	A	TGGGATAAAATGAAAAGAATATGTGAAATTATG AAAA[86]	11 (19)	27º (4)	5
GCAT**N$_{19}$**ATAA	SigA	arsR	P	TGCTTGCATTATTTTAAAAATCATGAGTATAATAA ATACATCAA[87]	05 (13)	05º (09)	2
GCAT**N$_{18}$**ATAA	SigA	dppA	P	TTCCCAGTTATATTGCATTTTTCCTCTTTTTTTAA TATAATTTGTTAGAATATTCATAATTTAGT[88]	05 (13)	17º (06)	2
GGAA**N$_{18}$**AATA	SigH	yvyD	P	CAGCAGGAATTGTAAAGGGTAAAAGAGAAATAG ATACATATCCT[89]	02 (19)	02º (15)	2
GTAT**N$_{22}$**ATAA	SigE	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[90]	01 (20)	04º (12)	4
GTAT**N$_{22}$**ATAA	SigF	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[91]	01 (20)	04º (12)	4
GTAT**N$_{22}$**ATAA	SigE	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[92]	01 (20)	04º (12)	4
GTAT**N$_{22}$**ATAA	SigF	gerAA	P	CACAGTATATCATTTTTTTAACAGGAAAAGATAA CCTCTAC[93]	01 (20)	04º (12)	4

[73] Sussman e Setlow (1991)
[74] Eichenberger et al. (2003)
[75] Herbig e Helmann (2001)
[76] Darmon et al. (2002)
[77] Eichenberger et al. (2003)
[78] Ricca et al. (1992)
[79] Fuangthong et al. (2002)
[80] Eichenberger et al. (2003)
[81] Eichenberger et al. (2003)
[82] Antelmann et al. (1996)
[83] Schulz et al. (1997)
[84] Grundy et al. (1994)
[85] Baichoo et al. (2002)
[86] Darmon et al. (2002)
[87] Sato e Kobayashi (1998)
[88] Slack et al. (1991)
[89] Drzewiecki et al. (1998)
[90] Wu et al. (1992), Schuch e Piggot (1994)
[91] Wu et al. (1992), Schuch e Piggot (1994)
[92] Feavers et al. (1990)
[93] Feavers et al. (1990)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
GTAT**N$_{18}$**ATAA	SigA	lytR	P	AGAGTTGTATTTATTGGAAATTTAACTCATAATG AAAGTAATTT[94]	01 (20)	04º (12)	4
GTAT**N$_{19}$**ATAA	SigG	sspC	P	GCGTGTATAAATTAAAATAATCTCTCCATAATAT GATTCAAACAAG[95]	01 (20)	30º (02)	4
GTTA**N$_{17}$**AATA	purR	purR	R	GATTAAATCCGTATGTTAAGTTATATTGATCTTAA AATATTCGGATTTTGGGG[96]	01 (29)	2º (13)	4
GTTA**N$_{22}$**AATA	SigG	sspI	P	ACATGATGTTATTATATCGCAAGAACAGCACATA ATAAACCAGGTGC[97]	01 (29)	29º (3)	4
TAAC**N$_{04}$**ACAA	sigX	sigX	P	AATGTAACTTTTCAAGCTATTCATACGACAAAAA AGTGAACG[98]	19 (10)	02º (08)	2
TAAT**N$_{04}$**ATAA	PerR	hemA	R	AGAAACTATGTTATAATTATTATAAATAA[99]	06 (28)	10º (15)	3
TAAT**N$_{04}$**ATAA	PerR	mrgA	R	CTAAATTATAATTATTATAATTTAGTATTGATTTTT ATTTAGTATATGATATAA[100]	06 (28)	10º (15)	3
TAAT**N$_{21}$**ATAA	SigA	nasB	P	TTGTGACACGTTTAATGCGTTAACAATGCATTGT GACATAATTTTTAATAGGAGAAAACTTACGAG[101]	06 (28)	22º (11)	3
TAAT**N$_{19}$**ATAA	SigD	ybdO	P	TTTGAGGTTAATATATACATTATATTCGCCGATA AAAAAGAATAAGAGAGAATAC[102]	06 (28)	17º (12)	3
TAAT**N$_{01}$**ATAA	Fur	dhbA	R	TTATTTTTATAATTGATAATGATAATCATTATCAA TAGATTGCGTTTTTC[103]	06 (28)	4º (17)	3
TAAT**N$_{01}$**ATAA	Fur	yclN	R	GGTAATATGTAAATGATAATGATAATCAATTACT ATATGGCCATATTGTT[104]	06 (28)	4º (17)	3
TAAT**N$_{01}$**ATAA	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[105]	06 (28)	4º (17)	3
TAAT**N$_{01}$**ATAA	Fur	yxeB	R	CTATATTATTAATTGATAATGATAATCATTACTAA TCTATTGAGATACAT[106]	06 (28)	4º (17)	3
TAAT**N$_{23}$**TTAT	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[107]	09 (21)	7º (11)	3
TAAT**N$_{02}$**TTAT	Fur	feuA	R	CTATAATTCCAATTGATAATAGTTATCAATTGAAC AGGAGGCTCTATAGA[108]	09 (21)	4º (14)	3
TAAT**N$_{02}$**TTAT	CcpA	hutP	R	GTTAATAGTTATCA[109]	09 (21)	4º (14)	3
TAAT**N$_{11}$**TTAT	YrzC	ydbM	R	AAAAATGGCAGGAAATCTATAATACATATTAAAT TTATCGGAATTAAAACTGGGGGGCTGCCGG[110]	09 (21)	6º (12)	3
TAAT**N$_{08}$**TTAT	Fur	ydhU	R	AAGCGGTTAAACATGCTAATCCTCATCATTATAT TATTGGAGCGCAAAGT[111]	09 (21)	4º (14)	3
TAAT**N$_{11}$**TTAT	YrzC	yrrT	R	GTTCCATCATTAATCCATAGTATACTTATAGGAA TTATTAAATATGGAGT[112]	09 (21)	6º (12)	3
TACA**N$_{19}$**ATAA	SigA	gltR	P	TCTAAGCTTTAGTTTACATGAAGCTCTGCTATCA TATATAATTCAAAATTAAGATGGAA[113]	01 (17)	3º (15)	2
TACA**N$_{06}$**ATAA	PerR	PerR	R	TTACACTAATTATAAACATTACAATG[114]	01 (17)	4º (14)	2
TACA**N$_{18}$**ATAA	SigA	spo0E	P	AATGAAAATATGTTTACAAATAAAGTATAATCTGT AATAATGCACAATAACCCAATCAAACTTGT[115]	01 (17)	5º (13)	2
TACA**N$_{18}$**ATAA	Fur	ywbL	R	CTATGATTATGTTATACAATGATAATCATTTTCAA TTATAGGAGGAACAT[116]	01 (17)	5º (13)	2

[94] Lazarevic et al. (1992)
[95] Nicholson et al. (1989)
[96] Weng et al. (1995), Shin et al. (1997) e Saxild et al.(2001)
[97] Cabrera-Hernandez e Setlow (2000)
[98] Huang et al. (1997)
[99] Herbig e Helmann (2001)
[100] Weng et al. (1995), Shin et al. (1997) e Saxild et al.(2001)
[101] Nakano et al. (1995)
[102] Serizawa et al. (2004)
[103] Baichoo et al. (2002) e Ollinger et al. (2006)
[104] Herbig e Helmann (2001)
[105] Herbig e Helmann (2001)
[106] Herbig e Helmann (2001)
[107] Herbig e Helmann (2001)
[108] Baichoo et al. (2002) e Ollinger et al. (2006)
[109] Wray et al. (1994b)
[110] Even et al. (2006)
[111] Baichoo et al. (2002)
[112] Even et al. (2006)
[113] Belitsky e Onenshein (1997)
[114] Fuangthong et al. (2002)
[115] Peregot e Hoch (1991)
[116] Ollinger et al. (2006)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
TACA**N**$_{30}$ATAA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[117]	01 (17)	29º (4)	2
TACA**N**$_{06}$ATCA	Fur	ywbL	R	CTATGATTATGTTATACAATGATAATCATTTTCAA TTATAGGAGGAACAT[118]	13 (14)	17º (4)	2
TAGA**N**$_{02}$ATAT	TnrA	tnrA	A	TGTTAGAAAATATGACA[119]	02 (12)	1º (12)	2
TAGA**N**$_{16}$ATAT	SigD	yfmT	P	TGAACCGATAGAAAAAATAGATTCGCCCATATTT TGATTTGCGGTTATAAAGGAG[120]	02 (12)	21º (3)	2
TAGA**N**$_{23}$ATAT	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[121]	02 (12)	19º (4)	2
TATA**N**$_{05}$AATA	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[122]	03 (31)	14º (14)	2
TATA**N**$_{20}$AATA	SigF	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[123]	03 (31)	26º (10)	2
TATA**N**$_{20}$AATA	SigG	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[124]	03 (31)	26º (10)	2
TATA**N**$_{20}$AATA	SigG	sleB	P	AAAGAGTGTATAAAAATAACCTCGTTACAGAAAA TACGATTACACTT[125]	03 (31)	26º (10)	2
TATA**N**$_{18}$AATA	SigG	sspC	P	GCGTGTATAAATTAAAATAATCTCTCCATAATAT GATTCAAACAAG[126]	03 (31)	30º (6)	2
TATA**N**$_{07}$AATA	Fur	ywjA	R	CAGCCCGTGTATAGTATAATTGAGAAATATTATC AGTTATTTATACATTG[127]	03 (31)	8º (19)	2
TATA**N**$_{24}$AATA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[128]	03 (31)	29º (7)	2
TATA**N**$_{06}$ATAA	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[129]	16 (26)	20º (13)	5
TATA**N**$_{21}$ATAA	SigF	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[130]	16 (26)	13º (16)	5
TATA**N**$_{21}$ATAA	SigG	dacF	P	GGCGTATAAAACCATCACGCTTGGAAAAAATAA AAAGGAT[131]	16 (26)	13º (16)	5
TATA**N**$_{05}$AATG	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[132]	01 (21)	18º (09)	2
TATA**N**$_{01}$ACAT	PerR	perR	R	TTATAAACATTACAATGTAAGAA[133]	15 (14)	9º (9)	2
TATA**N**$_{24}$AGAA	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[134]	04 (25)	19º (11)	3
TATA**N**$_{06}$ATAA	PerR	mrgA	R	CTAAATTATAATTATTATAATTTAGTATTGATTTTT ATTTAGTATATGATATAA[135]	02 (36)	05º (21)	3
TATA**N**$_{16}$ATAA	SigE	ybaN	P	TCGGTTATATTCAATTGTCCATGCTCATAAGATG TAAAACAAGA[136]	02 (36)	03º (24)	3
TATA**N**$_{18}$ATAT	SigG	gerD	P	GTTATGTATAATTCCAAACAGATGAATCATATTA AAGGTAAGACAAGTATGTGAAAGGA[137]	04 (23)	22º (09)	2

[117] Yoshida et al. (2004)
[118] Baichoo et al. (2002) e Ollinger et al. (2006)
[119] Robichon et al. (2000)
[120] Serizawa et al. (2004)
[121] Yoshida et al. (2004)
[122] Herbig e Helmann (2001)
[123] Wu et al. (1992), Schuch e Piggot (1994)
[124] Schuch e Piggot (1994)
[125] Moriyama et al. (1999)
[126] Nicholson et al. (1989)
[127] Baichoo et al. (2002)
[128] Yoshida et al. (2004)
[129] Cabrera-Hernandez e Setlow (2000)
[130] Wu et al. (1992), Schuch e Piggot (1994)
[131] Wu et al. (1992), Schuch e Piggot (1994)
[132] Baichoo et al. (2002) e Ollinger et al. (2006)
[133] Fuangthong et al. (2002)
[134] Herbig e Helmann (2001)
[135] Herbig e Helmann (2001)
[136] Eichenberger et al. (2003)
[137] Kemp et al. (1991)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
TATA**N$_{27}$**ATAT	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[138]	04 (23)	20º (10)	2
TATA**N$_{6}$**ATGA	Fur	yoaJ	R	GATGGATTGAGTCTTATAATGATAATGATTCTCA TTTGAAGTCTGGTTTG[139]	2 (25)	2º (22)	2
TATA**N$_{22}$**TTAA	SigF	spoIVB	P	CAGTTATAAATAAGCCGTCAGAAGGCAAAATTAA ATGATGTA[140]	6 (19)	22º (8)	2
TATA**N$_{22}$**TTAA	SigG	spoIVB	P	CAGTTATAAATAAGCCGTCAGAAGGCAAAATTAA ATGATGTA[141]	6(19)	22º (8)	2
TATT**N$_{22}$**AGAA	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[142]	08 (22)	06º (13)	3
TATT**N$_{15}$**ATAA	SigE	yjmC	A	TATAATATAAAGAATATTTAAAATAATTTGTAAAT AAAATGTGTTTGTA[143]	15 (26)	01º (16)	5
TCTA**N$_{12}$**ATAA	LmrA	yxaG	R	TCCTACAATTATATAGAACGGTCTAGACAAATGA ATGATAATATATAGACTGGTCTAAATTGGAGGAA GC[144]	12 (17)	01º (17)	2
TCAT**N$_{06}$**TTAG	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[145]	08 (9)	04º (7)	3
TCAT**N$_{02}$**TTAT	Fur	ydhU	R	AAGCGGTTAAACATGCTAATCCTCATCATTATAT TATTGGAGCGCAAAGT[146]	01 (15)	02º (14)	3
TCTA**N$_{12}$**TTAT	PerR	ahpC	R	CTTGACAAAAAATATATATTAATTAATAATTCATA TATAATTAGAATTATTATTGAAAGCGA[147]	01 (15)	03º (13)	3
TGAA**N$_{19}$**ATAA	SigA	lepA	P	CTTTTCTCTTTCTTGTTTTACATTGAATCTTTACA ATCCTATTGATATAATCTAAGCTAGTGTATTTTG[148]	10 (29)	11º (15)	2
TGAA**N$_{07}$**TCAT	Fur	yfhC	R	TTCCGTTATCATTATGAAATGATAATCATTTTCAA TTGCATAGGAAGGTG[149]	07 (14)	01º (14)	4
TGAA**N$_{02}$**TCAT	Fur	ykuN	R	TTTATTTATCTGTTGACAATGAAAATCATTATCAT TTAAAGT[150]	07 (14)	28º (04)	4
TGAA**N$_{02}$**TTAT	CcpA	acuA	R	TGAAAACGCTTTAT[151]	27 (18)	28º (06)	4
TGAA**N$_{02}$**TTAT	YdiH	alsS	R	AAAGAGTGTATAGTGAAACTTATCACAAGATATT TA[152]	27 (18)	28º (06)	4
TGAA**N$_{05}$**TTAT	Fur	ybbB	R	TATTTGGTACAATTTTTATTGAAAATGATTATCAA TTGAAAGCTTCTGAA[153]	27 (18)	10º (11)	4
TGAA**N$_{05}$**TTAT	Fur	ykuN	R	TTTATTTATCTGTTGACAATGAAAATCATTATCAT TTAAAG[154]	27 (18)	10º (11)	4
TGAT**N$_{01}$**ATAA	PerR	hemA	R	TTCTATGTTAGAATGATTATAAATTAAGATTGGG TGTTGGGG[155]	03 (21)	5º (17)	3
TGAT**N$_{20}$**ATAT	SigA	iolR	P	CTATTGATTAACTTTTGGTTTTTATTATATTTTAT GTTACGTA[156]	03 (20)	22º (7)	3
TGAT**N$_{15}$**ATAT	SigE	yjbX	P	TTTCTTCTGATTTTCAGCTTTCTGTCATATAGATA GAATATGACACAAT[157]	03 (20)	16º (8)	3
TGTA**N$_{06}$**ATCA	CcpA	malA	R	TGGAATTGTAAACGTTATCAAGGAGGT[158]	02 (13)	07º (6)	2

[138] Yoshida et al. (2004)
[139] Baichoo et al. (2002) e Ollinger et al. (2006)
[140] Gomez e Cutting (1996)
[141] Gomez e Cutting (1996)
[142] Herbig e Helmann (2001)
[143] Mekjian et al. (1999)
[144] Yoshida et al. (2004)
[145] Herbig e Helman (2001)
[146] Baichoo et al. (2002)
[147] Herbig e Helman (2001)
[148] Hippler et al. (1997)
[149] Baichoo et al. (2002)
[150] Baichoo et al. (2002)
[151] Grundy et al. (1994)
[152] Reents et al. (2006)
[153] Baichoo et al. (2002)
[154] Baichoo et al. (2002)
[155] Cabrera-Hernandez e Setlow (2000)
[156] Yoshida et al. (1997)
[157] Feucht et al. (2003)
[158] Yamamoto (2001)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
TGTA**N$_{21}$**ATAA	SigA	lonA	P	TGCCGTTTATTGTACAAGGATGAAAAAGTTGTAT TATAATGGTTCATACTAAAGTCACGGAGGT[159]	02 (22)	18º (08)	2
TGTA**N$_{20}$**ATAA	SigA	lytR	P	AGAGTTGTATTTATTGGAAATTTAACTCATAATG AAAGTAATTT[160]	02 (22)	29º (03)	2
TTAA**N$_{09}$**AACA	PhoP	glpQ	A	GAAAGACACATAAAAGATTAATAGTTTTCCAACA CGCCGTTTACATCCGT[161]	3 (13)	3º (11)	1
TTAA**N$_{04}$**AACA	ExuR	uxaC	R	AAAACAAATCAAAATGTTAACGTTAACATTTTGA AATAGAATGA[162]	3 (13)	18º (8)	1
TTAA**N$_{10}$**ATAT	YrzC	yxeK	R	ATTCATATTAAACGACTAGGAATATAGGAGTTTA TTTTTCGCATT[163]	20 (18)	03º (15)	4
TTAC**N$_{20}$**ATAA	SigA	gltR	P	TCTAAGCTTTAGTTTACATGAAGCTCTGCTATCA TATATAATTCAAAATTAAGATGGAA[164]	01 (14)	17º (06)	4
TTAC**N$_{07}$**ATAA	PerR	PerR	R	TTACACTAATTATAAACATTACAATG	01 (14)	09º (08)	4
TTAC**N$_{19}$**ATAA	SigA	yfmP	P	TTAACGTTTACGTTAAGGTTCAAAAGGTGTATAA TGGTAACAGAAA[165]	01 (14)	09º (08)	4
TTAG**N$_{02}$**ATAA	SpoIII D	cotJA	A	AAGTCGTGTTTTAGTCATAATCATGCCTCC[166]	03 (15)	08º (08)	2
TTAG**N$_{07}$**ATAA	PerR	hemA	R	TTCTATGTTAGAATGATTATAAATTAAGATTGGG TGTTGGGG[167]	03 (15)	02º (02)	2
TTAT**N$_{01}$**ACAA	YdiH	alsS	R	AAAGAGTGTATAGTGAAACTTATCACAAGATATT TA[168]	23 (15)	21º (08)	4
TTAT**N$_{02}$**AGAA	LmrA	yxaG	R	CTACAATTATATAGAACGGTCTAGACAAATGAAT GATAATATATAGACTGGTCTAAATTGGAGGAC[169]	03 (21)	02º (19)	3
TTAT**N$_{01}$**ATAA	PerR	katA	R	TTATTTATCAGTTTATAATAATTATAGTTGGAA[170]	01 (40)	01º (40)	3
TTAT**N$_{07}$**ATAA	PerR	hemA	R	AGAAACTATGTTATAATTATTATAAATAA[171]	01 (40)	07º (22)	3
TTAT**N$_{18}$**ATAA	SigF	spoIIR	P	CACGTTTATCCCAGGCTCTCCTTGTCCATAATAG GGCTAGA[172]	01 (40)	11º (19)	3
TTAT**N$_{19}$**ATAA	SigG	sspI	P	ACATGATGTTATTATATCGCAAGAACAGCACATA ATAAACCAGGTGC[173]	01 (40)	9º (20)	3
TTAT**N$_{18}$**ATAA	SigE	ybaN	P	TCGGTTATATTCAATTGTCCATGCTCATAAGATG TAAAACAAGA[174]	01 (40)	11º (19)	3
TTAT**N$_{20}$**ATGA	SigA	gabR	P	TCGGTATTTTCTTATCATTCTGACTTCTCTTTGGT ATGATGAAAAGTACCA[175]	30 (19)	12º (12)	5
TTAT**N$_{05}$**ATGA	Fur	ybbB	R	TATTTGGTACAATTTTTATTGAAAATGATTATCAA TTGAAAGCTTCTGAA[176]	30 (19)	14º (13)	5
TTAT**N$_{20}$**ATGA	SigG	sspK	P	TAACGCTTTATTACGTGGTGTTCTCCATATACTA ACCTTACGTCTTC[177]	01 (20)	04º (15)	2
TTAT**N$_{16}$**GTAA	SigB	katE	P	TAGCAGTTTATATGAAGAACGCCACGGGTAAAT GTGCTGTAGAA[178]	11 (17)	09º (11)	4
TTAT**N$_{15}$**GTAA	SigB	ytxG	P	AGTACACATGTTTATGATTGAAGAAAACGGGTAA ACAGCAGTATAT[179]	11 (17)	09º (11)	4
TTAT**N$_{05}$**TTAC	perR	perR	R	TTATAAACATTACAATGTAAGAA[180]	01 (14)	02º (9)	3

[159] Riethdorf et al. (1994)
[160] Lazarevic et al. (1992)
[161] Allenby et al. (2005)
[162] Mekjian et al. (1999)
[163] Even et al. (2006)
[164] Belitsky e Sonenshein (1997)
[165] Gaballa et al. (2003)
[166] Henriques et al. (1997)
[167] Herbig e Helmann (2001)
[168] Reents et al. (2006)
[169] Yoshida et al. (2004)
[170] Fuangthong et al. (2002)
[171] Herbig e Helmann (2001)
[172] Karow et al. (1995)
[173] Cabrera-Hernandez e Setlow (2000)
[174] Eichenberger et al. (2003)
[175] Belitsky e Sonenshein (2002)
[176] Baichoo et al. (2002)
[177] Cabrera-Hernandez e Setlow (2000)
[178] Engelmann et al. (1995)
[179] Varón et al. (1996)
[180] Fuangthong et al. (2002)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
TTAT**N$_{02}$**TTAT	PerR	hemA	R	AGAAACTATGTTATAATTATTATAAATAA[181]	02 (26)	01º (26)	3
TTAT**N$_{02}$**TTAT	PerR	mrgA	R	CTAAATTATAATTATTATAATTTAGTATTGATTTTTATTTAGTATATGATATAA[182]	02 (26)	01º (26)	3
TTAT**N$_{05}$**TTAT	PerR	katA	R	CTATTTTATAATAATTATAAAATAATATTGACTTTTTACTTAGAGATGATATTATGTT[183]	02 (26)	06º (19)	3
TTAT**N$_{09}$**TTAT	Fur	ybbB	R	TATTTGGTACAATTTTTATTGAAAATGATTATCAATTGAAAGCTTCTGAA[184]	02 (26)	23º (10)	3
TTCT**N$_{20}$**ACAA	SigE	yabP	P	CTTGTTCTAAAAAAACCCCCCACCTCATACAATGCAGTAATAATG[185]	20 (11)	01º (11)	2
TTGA**N$_{20}$**AAGA	SigA	sboA	P	AATATATGTATTGAATTAGTAATTTGATAGTTTTAAGATAAAAGTACAAC[186]	11 (17)	15º (9)	1
TTGA**N$_{21}$**AAGA	SigA	yfkJ	P	GACATCAGTTGAAAAGAAAATGAACATCCTACTAAGATATTCATGAAGGTTT[187]	11 (17)	24º (6)	1
TTGA**N$_{19}$**AATA	SigA	pyrR	P	ACGGTTGACAGAGGGTTTCTTTTCTGAAATAATAAACGAAG[188]	22 (30)	24º (7)	1
TTGA**N$_{19}$**AATA	SigA	ahpC	P	TATGGCTTGACAAAAATATATATTAATTAATAATTCATATATAATT[189]	22 (30)	24º (7)	1
TTGA**N$_{21}$**AATA	SigA	htpG	P	ATCTAATTGACAATTGTCATCTTATGTGATAAATAGATGCTGAAAA[190]	22 (30)	10º (11)	1
TTGA**N$_{20}$**AATA	SigA	hutP	P	AAAAAACCTTTTGACTTCTGCTGCTGAACCAATTAATATAATACTCAGTTAATAGTTATCAGA[191]	22 (30)	03º (23)	1
TTGA**N$_{20}$**AATA	SigA	nadB	P	ATAAAAAACTCTTGAGTTTATTTTATCCTTGTGTAAATATAGGTGTCAAGACAGGTGTAAACA[192]	22 (30)	03º (23)	1
TTGA**N$_{20}$**AATA	YrxA	nadB	R	CATCCGGTCTTCCTCCATCCGTTCTCCATAAAAAACTCTTGAGTTTATTTTATCCTTGTGTAAATATAGGTGTCAAGACAGGTGTAAACAACAGGAGGATGGCATATG[193]	22 (30)	03º (23)	1
TTGA**N$_{02}$**AATA	Fur	ywjA	R	CAGCCCGTGTATAGTATAATTGAGAAATATTATCAGTTATTTATACATTG[194]	22 (30)	10º (11)	1
TTGA**N$_{20}$**ATAA	SigA	lepA	P	ACTTTTCTCTTTCTTGTTTTACATTGAATCTTTACAATCCTATTGATATAATCTAAGCTAGTGTATTTTG[195]	20 (26)	01º (26)	1
TTGA**N$_{20}$**ATAA	SigA	acsA	P	GGTTTATATTTTAAAAATTGAGAAGAATATGAATATATACTATAATAATTGTGACAACTTCAGCAAAGGG[196]	20 (26)	01º (26)	1
TTGA**N$_{20}$**ATAA	SigA	ahpC	P	TATGGCTTGACAAAAATATATATTAATTAATAATTCATATATAATT[197]	20 (26)	01º (26)	1
TTGA**N$_{19}$**ATAA	SigA	argC	P	AATATACGATTGAATTAATTTTTATTCATGTTATAATGTTAAATAATTTCACAAAGACCAA[198]	20 (26)	13º (13)	1
TTGA**N$_{20}$**ATAA	SigA	lmrA	P	CAATGAATTTTTCTTGACAATTGATGATTGAATCAAGATAATAGACCAGTCACTAT[199]	20 (26)	01º (26)	1
TTGA**N$_{19}$**ATAA	SigA	mrgA	P	CTAAATTATAATTATTATAATTTAGTATTGATTTTTATTTAGTATATGATATAATTAAGTCAAC[200]	20 (26)	13º (13)	1
TTGA**N$_{21}$**ATAA	SigA	rpmH	P	ACTAGTGAAGTTGACAATGAATAGGTAACGCAAATATAATAAGTAAGACTGTCTTTAACAGCTATTCCTCGA[201]	20 (26)	06º (15)	1

[181] Herbig e Helmann (2001)
[182] Herbig e Helmann (2001)
[183] Herbig e Helmann (2001)
[184] Baichoo et al. (2002)
[185] Asai et al. (2001)
[186] Zheng et al. (2000)
[187] Price et al. (2001)
[188] Quinn et al. (1991)
[189] Antelmann et al. (1996)
[190] Schulz et al. (1997)
[191] Wray et al. (1994)
[192] Sun e Setlow (1993)
[193] Sun e Setlow (1993) e Rossolillo et al. (2005)
[194] Baichoo et al. (2002)
[195] Hippler et al. (1997)
[196] Grundy et al. (1994)
[197] Antelmann et al. (1996)
[198] O'Reilly et al. (1994) e Smith et al. (1986)
[199] Kumano et al. (2003)
[200] Fuangthong e Helmann (2003)

Indivíduos	FT	Gene	Reg	Região Regulatória Completa do oligômero	MF	PO	VR
TTGA**N$_{24}$**ATAA	SigA	spoIIE	P	TTACCTTCTTTTGACAAAATCCTATCTGTGCTTTC GCTATAATGACAGGCAACGAATATAACAGGTG[202]	20 (26)	16º (11)	1
TTGA**N$_{16}$**ATAA	SigE	yybI	P	TATGTCTTTGATGTGCCATTCCTGCATATAATGA TTCATTCAGCTG[203]	20 (26)	26º (7)	1
TTGA**N$_{17}$**GATA	SigA	hrcA	P	TGGGTGAGTTATAATTGACATTTTTCTTGTGGTT TGATACTTTTGTTATAGAATTAGCACTCGCTTA[204]	17 (19)	01º (19)	1
TTGA**N$_{19}$**GATA	SigA	lmrA	P	CAATGAATTTTTCTTGACAATTGATGATTGAATC AAGATAATAGACCAGTCACTAT[205]	17 (19)	11º (8)	1
TTGC**N$_{22}$**TTAT	SigA	ycdH	P	ATTTATTCTTGCAAAACGTAATGACTTCGGTTTA TTATGATATAGGATTACAAAATCGTTATCATTTTG ATTTAAAG[206]	02 (12)	05º (7)	3
TTGT**N$_{20}$**ATAA	SigA	csbA	P	ACGATTGTCCGATTCTTCATTTTTTACTATAATCA GATCAGATG[207]	03 (24)	03º (17)	3
TTGT**N$_{20}$**ATAA	SigA	lrpC	P	GAACTGTACTTGTCATTTACAAAAATACCCGAGA TAATGTGTACAAAATCAAAAAAGAAGGATGT[208]	03 (24)	03º (17)	3
TTGT**N$_{20}$**ATAA	SigA	lytR	P	AGAGTTGTATTTATTGGAAATTTAACTCATAATG AAAGTAATTT[209]	03 (24)	03º (17)	3
TTGT**N$_{20}$**ATAA	SigA	rsbR	P	AGAGCAACTTTTTTTGTTTTCAAAAAACATAAAC GATATAATAGTGAAATAACGAAAAAATATGTT[210]	03 (24)	03º (17)	3
TTGT**N$_{05}$**ATAA	SacT	sacB	A	TCGCGCGGGTTTGTTACTGATAAAGCAGGCAAG ACCTAAAATG[211]	03 (24)	06º (15)	3
TTGT**N$_{09}$**ATCA	YdiH	cydA	R	TTTCGTCTATTTTGTGAATTACTGATCAAAGTCT CGGTTCTA[212]	03 (11)	27º (02)	4
TTGT**N$_{08}$**ATGA	TnrA	gltA	R	AGAGTTGTTAGATTTTATGACCGGTA[213]	29 (14)	03º (12)	4
TTGT**N$_{20}$**ATGA	SigA	ptsG	P	TGCTTGTCAGATGACAAGTACGGTTGTATGATAT AATATTGTGAAG[214]	29 (14)	04º (11)	4

Os 216 *motifs* encontrados pelo GA_FIND_RR, que possuem referências bibliográficas, estão distribuídos da seguinte forma:

- 113 *motifs* são promotores;
- 96 *motifs* são repressores;
- 7 *motifs* são ativadores.

GA_FIND_RR localizou *motifs* diferentes de uma mesma RR. Excluindo as repetições, as RR encontradas que estão listadas em referências bibliográficas, totalizam 124 RR. Sendo distribuídos da seguinte forma:

---

[201] Ogasawara et al. (1985)
[202] York et al. (1992) e Guzman et al. (1988)
[203] Wetzstein et al. (1992)
[204] Wetzstein et al.. (1992) e Homuth et al. (1997)
[205] Kumano et al. (2003)
[206] Gaballa et al. (2002)
[207] Boylan et al. (1991)
[208] Beloin et al. (2000)
[209] Lazarevic et al. (1992) , Huang e Helmann (1998)
[210] Wise et al. (1995)
[211] Steinmetz et al. (1989)
[212] Reents et al. (2006)
[213] Belitsky et al. (2000)
[214] Stülke et al. (1997)

- 83 *motifs* são promotores;
- 35 *motifs* são repressores;
- 6 *motifs* são ativadores.

Os dois oligômeros de quatro bases mais representativos, presentes na base de dados analisada são: ATAA com 2843 repetições e AATA com 2630 repetições, conforme demonstrado na tabela 8. Observa-se que estes dois oligômeros são comuns nas RR encontradas, o que comprova que a "super-representatividade" realmente é um fator determinante na localização de RR.

Neste trabalho, não é possível a comprovação de que todas as RR estão entre as distâncias de 0 a 30 bases, pois os testes concentraram-se entre estas distâncias, não sendo executado nenhum teste com distâncias maiores. Porém, pode-se observar que as distâncias encontradas entre os oligômeros estão abaixo de 25 bases, o que pode significar um forte indício que distâncias até 30 bases são suficientes nos testes em busca de RR, pelo menos para o *Bacillus subtilis.*

Na tabela 11, encontram-se cinqüenta oligômeros preditos como possíveis RR pelo GA_FIND_RR e que não estão documentados como RR. Foram selecionados os dez melhores oligômeros de cada versão que não estão documentados como regiões regulatórias.

Tabela 16- *Motifis* sem referência na literatura.
Nesta tabela estão sendo considerados os oligômeros mais "super-representados", independente da versão utilizada. A coluna "**Indivíduo**" é o oligômero encontrado pelo GA_FIND_RR. As colunas "**F01**" até "**F10**" são os 10 melhores *fitness* de cada indivíduo, sendo representado como DD (FF), onde DD é a distância entre as duas partes do oligômero e FF é o *fitness* do indivíduo. "V" é a versão que foi encontrado o oligômero

Indivíduos	F01	F02	F03	F04	F05	F06	F07	F08	F09	F10	V
AAGG**Nx**TGAA	03 (29)	02 (21)	06 (20)	04 (19)	05 (17)	01 (16)	07 (15)	11 (13)	13 (12)	14 (11)	1
AATA**Nx**AGAA	02 (26)	01 (24)	05 (23)	18 (21)	06 (20)	03(19)	07 (16)	04 (15)	10 (15)	12 (15)	1
AATA**Nx**AGGA	02 (25)	03 (23)	20 (23)	05 (21)	12 (21)	16 (19)	01 (18)	07 (18)	08 (18)	06 (17)	1
AATA**Nx**AGGT	23 (15)	03 (14)	28 (14)	02 (11)	09 (11)	14 (11)	15 (11)	04 (10)	08 (10)	13 (10)	1
AATA**Nx**GATA	07 (17)	06 (15)	02 (14)	13 (12)	01 (11)	08 (11)	12 (11)	04 (10)	15 (10)	05 (09)	1
GAGG**Nx**TGAA	03 (26)	05 (21)	01 (20)	04 (19)	02 (16)	06 (08)	09 (06)	12 (06)	18 (06)	25 (06)	1
GGAG**Nx**ATCA	03 (32)	04 (22)	05 (19)	02 (16)	01 (11)	06 (10)	07 (10)	08 (07)	09 (07)	29 (05)	1
TCAT**Nx**ATAA	04 (23)	05 (21)	03 (20)	08 (17)	14 (16)	22 (16)	02 (15)	12 (15)	01 (13)	06 (12)	1

Tabela 16 - Continuação...

Indivíduos	F01	F02	F03	F04	F05	F06	F07	F08	F09	F10	V
TTCT**Nx**ATAA	03 (18)	17 (18)	19 (18)	01 (17)	04 (17)	16 (17)	02 (16)	20 (16)	09 (15)	12 (14)	1
TTGA**Nx**ACAA	20 (18)	04 (11)	06 (11)	14 (10)	21 (10)	13 (09)	01 (08)	02 (08)	27 (08)	05 (07)	1
ATAA**Nx**ATAC	09 (16)	05 (13)	08 (13)	12 (13)	03 (12)	07 (12)	19 (11)	18 (10)	25 (10)	10 (09)	2
ATAA**Nx**ATCA	01 (16)	09 (14)	04 (13)	05 (13)	07 (12)	11 (12)	12 (12)	15 (12)	16 (12)	21 (12)	2
GAGA**Nx**AATA	02 (19)	01 (13)	09 (11)	10 (11)	03 (09)	19 (09)	04 (08)	13 (08)	08 (07)	12 (07)	2
GATA**Nx**AATA	01 (47)	03 (16)	10 (15)	02 (14)	05 (14)	07 (14)	16 (12)	04 (11)	06 (10)	09 (10)	2
GATA**Nx**ATAA	02 (31)	03 (21)	08 (16)	12 (14)	15 (14)	04 (13)	01 (12)	10 (12)	18 (12)	07 (11)	2
GGTA**Nx**AATA	01 (27)	04 (08)	07 (07)	12 (07)	03 (06)	05 (06)	08 (06)	13 (06)	06 (05)	21 (05)	2
GTAA**Nx**ATAA	01 (27)	11 (15)	10 (13)	12 (13)	05 (12)	09 (12)	02 (11)	04 (11)	14 (11)	17 (10)	2
TATA**Nx**ATCA	09 (17)	08 (14)	02 (13)	12 (13)	18 (12)	05 (11)	06 (11)	10 (11)	24 (11)	04 (10)	2
TATA**Nx**GGAA	23 (17)	09 (16)	24 (16)	06 (15)	04 (13)	14 (13)	17 (13)	03 (12)	05 (12)	27 (12)	2
TCAT**Nx**ATAA	04 (23)	05 (21)	03 (20)	08 (17)	14 (16)	22 (16)	02 (15)	12 (15)	01 (13)	06 (12)	2
CTAT**Nx**ATGA	01 (14)	11 (10)	19 (10)	02 (08)	04 (08)	12 (08)	26 (08)	27 (08)	13 (07)	14 (07)	3
GACA**Nx**TATA	15 (12)	10 (10)	22 (10)	05 (08)	18 (08)	20 (08)	30 (08)	02 (07)	17 (07)	26 (07)	3
GATA**Nx**AAGA	01 (16)	07 (13)	14 (13)	05 (12)	02 (11)	08 (11)	10 (11)	18 (11)	24 (11)	11 (10)	3
GATA**Nx**AATA	01 (47)	03 (16)	10 (15)	02 (14)	05 (14)	07 (14)	16 (12)	04 (11)	06 (10)	09 (10)	3
GATA**Nx**CATA	09 (12)	02 (09)	05 (09)	17 (09)	08 (08)	14 (08)	01 (07)	10 (07)	11 (07)	13 (07)	3
GATG**Nx**AATA	11 (11)	06 (10)	10 (10)	13 (09)	04 (08)	09 (07)	15 (07)	19 (07)	01 (06)	03 (06)	3
GATT**Nx**AATA	08 (18)	20 (18)	05 (16)	06 (13)	10 (13)	14 (12)	17 (11)	02 (10)	03 (10)	07 (10)	3
TATA**Nx**AGAA	04 (25)	01 (23)	05 (22)	02 (21)	07 (19)	08 (18)	19 (18)	25 (16)	16 (15)	21 (15)	3
TATG**Nx**ATAA	02 (26)	05 (22)	01 (21)	06 (17)	07 (17)	03 (15)	17 (14)	09 (13)	10 (13)	11 (13)	3
TTAT**Nx**CTAA	01 (11)	03 (10)	21 (10)	12 (09)	15 (09)	24 (09)	06 (07)	29 (07)	04 (06)	16 (06)	3
AATA**Nx**AGAA	02 (25)	01 (23)	18 (21)	09 (20)	16 (20)	28 (18)	05 (17)	10 (17)	19 (17)	24 (17)	4
AATA**Nx**TTAA	18 (21)	21 (20)	15 (17)	20 (17)	04 (16)	07 (16)	06 (15)	08 (15)	10 (15)	19 (15)	4
ATAA**Nx**ATAT	10 (23)	07 (22)	05 (21)	17 (20)	22 (20)	01 (19)	04 (18)	27 (18)	12 (16)	15 (16)	4
ATAA**Nx**TCAT	05 (20)	13 (16)	02 (14)	09 (14)	14 (14)	19 (14)	24 (13)	10 (12)	12 (12)	18 (12)	4
ATAA**Nx**TTAT	08 (25)	14 (21)	10 (20)	25 (19)	18 (18)	28 (18)	30 (18)	12 (16)	21 (16)	06 (15)	4
ATAA**Nx**TTAC	09 (16)	10 (12)	03 (11)	04 (11)	25 (10)	14 (09)	15 (09)	22 (09)	16 (08)	18 (08)	4
GATA**Nx**ATAA	02 (24)	12 (16)	16 (15)	03 (14)	08 (14)	17 (12)	10 (11)	04 (10)	11 (10)	18 (10)	4
TATA**Nx**TGAA	01 (31)	03 (20)	27 (19)	11 (18)	15 (18)	22 (18)	29 (18)	02 (16)	07 (15)	08 (15)	4
TTAA**Nx**ATAA	18 (22)	22 (20)	07 (18)	03 (16)	01 (15)	10 (15)	14 (15)	08 (14)	12 (14)	27 (14)	4
TTAT**Nx**ATAG	01 (21)	07 (13)	11 (13)	14 (13)	18 (13)	21 (13)	22 (13)	26 (13)	03 (12)	06 (12)	4
AAGA**Nx**ATAA	20 (20)	14 (17)	13 (16)	11 (15)	27 (13)	29 (13)	09 (12)	18 (12)	30 (12)	01 (11)	5
AATT**Nx**ATAA	28 (23)	25 (22)	22 (21)	05 (20)	12 (17)	14 (17)	19 (17)	21 (17)	04 (15)	10 (15)	5
AGAA**Nx**ATGA	10 (19)	07 (16)	01 (15)	06 (15)	11 (15)	03 (14)	04 (14)	15 (14)	22 (14)	23 (14)	5
AGAA**Nx**GGAA	01 (33)	05 (21)	03 (17)	06 (17)	04 (16)	10 (16)	08 (15)	07 (14)	09 (13)	20 (13)	5
ATAA**Nx**AAGA	08 (21)	01 (20)	09 (19)	26 (19)	16 (18)	05 (17)	14 (17)	19 (17)	28 (17)	17 (16)	5
ATAA**Nx**AATA	19 (28)	09 (26)	20 (26)	27 (25)	02 (24)	10 (22)	11 (21)	08 (20)	15 (20)	21 (20)	5
ATAA**Nx**ATAT	10 (23)	07 (22)	05 (21)	17 (20)	22 (20)	01 (19)	04 (18)	27 (18)	12 (16)	15 (16)	5
ATAA**Nx**ATCA	09 (19)	01 (17)	13 (17)	16 (17)	05 (15)	21 (14)	04 (12)	07 (12)	11 (12)	19 (12)	5
ATAA**Nx**ATTG	22 (16)	05 (15)	14 (15)	24 (15)	13 (14)	07 (13)	09 (13)	15 (13)	16 (13)	30 (13)	5
ATAA**Nx**GGAA	29 (25)	21 (24)	18 (20)	25 (20)	23 (19)	03 (18)	04 (18)	06 (18)	24 (18)	17 (17)	5

6.10 Limitações e Potencialidades

Após noventa execuções usando o GA_FIND_RR, percebeu-se que o mesmo não é capaz de localizar todas as RR do *Bacillus subtilis*. O algoritmo foi capaz de localizar em torno de 20% das 635 regiões conhecidas do *Bacillus subtilis.*

As dificuldades e limitações para o desenvolvimento do GA_FIND_RR são semelhantes às descritas na seção 1.3.5, podendo ser destacadas como principais dificuldades os itens a seguir:

- O melhor indivíduo de cada execução do GA_FIND_RR "dominava" a última geração, restando poucos indivíduos com *fitness* representativos. Porém, geralmente o melhor indivíduo era uma RR catalogada nas referências bibliográficas analisadas;

- Ocorreram resultados não documentados na literatura com as principais características conhecidas como sendo uma RR, ou seja, repetem-se várias vezes e têm espaçamentos entre eles de 0 e 30 bases. Estes "ruídos" dificultaram a criação de um algoritmo mais preciso;

- A função de adaptação baseou-se, principalmente, em dois fatores: a "super-representatividade" e a separação dos oligômeros em distâncias variando de 0 a 30 bases. Pode ter sido deixado de fora alguma característica importante que ainda não é conhecida pelos cientistas que trabalham com Biologia Molecular, que poderia auxiliar na criação de uma função de adaptação mais precisa;

- Outro fator que deve ser levado em consideração é a heurística utilizada na escolha das configurações dos parâmetros usados no GA_FIND_RR, pois como ainda não existe um perfeito conhecimento de todas as características das RR, as escolhas dos parâmetros ficaram na dependência dos resultados obtidos em execuções passadas. Ou seja, baseou-se no esquema "tentativa → erro → tentativa → acerto" e assim sucessivamente;

- O tamanho da base de dados analisada e o tamanho do oligômero foram fatores decisivos na execução do GA_FIND_RR. O algoritmo só obteve resultados positivos com tamanhos de bases *upstream* com 100 bases e oligômeros de 16 bits (ou 8 bases, sendo 4 bases para cada parte do oligômero).

Em relação às potencialidades do GA_FIND_RR, podem-se destacar as seguintes:

- Como nem todas as RR para o *Bacillus subtilis* são conhecidas, podem existir oligômeros encontrados pelo GA_FIND_RR que possam ter funções reguladoras. Sendo assim, sugere-se uma análise detalhada da tabela 11, que lista os dez oligômeros de melhor *fitness* para cada versão desenvolvida do GA_FIND_RR;

- Algoritmo com conceito relativamente simples e funcional. À medida que novas descobertas referentes às características relacionadas às RR de procariontes forem sendo reveladas, estas podem ser introduzidas na função de adaptação, o que, provavelmente, melhorará a eficiência do mesmo;

- O GA_FIND_RR pode ser convertido, futuramente, para uma linguagem de programação de baixo nível, podendo melhorar consideravelmente o seu desempenho. Como o MatLab é um ambiente computacional com uma linguagem interpretada, o tempo de execução ficou elevado. Para uma população de 100 indivíduos e 100 gerações, o tempo médio de execução ficou em torno 4 a 6 horas, dependendo do computador onde o programa foi executado;

- Na maioria das execuções do GA_FIND_RR, percebeu-se que o melhor indivíduo da última geração, geralmente, era uma RR documentada, o que comprova que pelo menos as RR mais representativas são localizadas. Esta característica pode vir a auxiliar pesquisadores como ponto de partida na pesquisa de prováveis RR de organismos de procariontes recém seqüenciados;

- O GA_FIND_RR pode ser usado em conjunto com outros algoritmos de predição de RR, sendo mais uma ferramenta para comparação dos resultados obtidos. Caso um determinado oligômero receba um *fitness* elevado em todos os algoritmos usados, este oligômero pode ser considerado como sendo um provável candidato a ser uma RR.

5 Conclusões e Propostas de Melhorias

Os resultados obtidos com o algoritmo GA_FIND_RR comprovaram que o uso do AG pode ser uma solução complementar a outras soluções já desenvolvidas para a predição de RR.

Porém, como ainda não há um perfeito conhecimento biológico de todas as características das RR em organismos procariontes, existe um grande trabalho a ser desenvolvido em todos os algoritmos de predição de RR, incluído o GA_FIND_RR. Os algoritmos estudados que apresentaram os melhores resultados são os desenvolvidos para predição de RR objetivando apenas um organismo específico, como é o caso dos algoritmos desenvolvidos por Li et al. (2002) e Mwangi e Siggia (2003).

Foram criadas cinco versões distintas do GA_FIND_RR, conforme explicado no capítulo 3. A soma dos resultados obtidos com estas versões possibilitaram a predição de aproximadamente 20% das 635 RR conhecidas para o *Bacillus subtilis*. Sugere-se, em uma versão futura, do GA_FIND_RR, desenvolver uma função de adaptação com as cinco funções implementadas no mesmo código fonte, deixando como parâmetro para o usuário selecionar qual versão deverá ser usada.

Mudanças nos operadores genéticos, também alteravam os resultados obtidos, para uma mesma versão, principalmente quando alterava-se o valor percentual da taxa de mutação e cruzamento. Taxas de mutação, em torno de 2% a 10%, com taxas de cruzamento variando entre 60% a 80%, apresentaram bons resultados. Para garantir que o melhor indivíduo não fosse perdido, em praticamente todas as execuções do GA_FIND_RR, foi usado o conceito de elitismo, variando entre 1 a 10 indivíduos.

O uso da técnica de torneio para o processo de seleção, variando entre 6 a 10 indivíduos, diminuiu a perda de diversidade. Em versão futura, sugere-se usar a técnica de *Rank*, com a intenção de minimizar a perda de diversidade devido a forte pressão seletiva ocasionada por indivíduos com alto valor de *fitness*.

Outra proposta de melhoria para o GA_FIND_RR, para uma versão futura, com o objetivo de minimizar uma rápida convergência para um determinado ponto no espaço de busca, é a técnica conhecida como: “Adaptação das probabilidades de cruzamento e mutação” (SRINIVAS e PATNAIK, 1994a), onde propõe-se uma nova versão do AG, chamado

"Algoritmo Genético Adaptativo", que visa manter uma maior diversidade da população, trabalhando dinamicamente com os operadores de cruzamento e mutação de acordo com o *fitness* da população. Uma segunda alternativa é usar uma técnica conhecida como *Population-Based Incremental Learning* (PBIL), proposta por Baluja e Caruana (1995), onde as regras para a criação da nova população podem ser modificadas durante a execução do algoritmo, objetivando gerar um conjunto de resultados mais satisfatórios que o proposto tradicionalmente pelo método canônico.

Sugere-se para próxima versão do GA_FIND_RR, o desenvolvimento de uma interface gráfica amigável, onde o usuário possa parametrizar todas as variáveis disponíveis no algoritmo, incluindo, ente elas, a possibilidade de escolher o tamanho do oligômero.

Foi demonstrado que a utilização de métodos alternativos, como o AG, em relação aos métodos mais tradicionalmente usados (Matrizes de peso, *Hiden Markov Model, Gibbs Sampling* e busca exaustiva) também podem auxiliar na predição de RR de procariontes, podendo ser usados em conjunto com os demais métodos, com o objetivo de melhorar a eficiência global da busca por estas regiões.

REFERÊNCIAS

AERTS, S., LOO, P.V., MOREAU, Y., MOOR, D.B. **A genetic algorithm for the detection of new cis-regulatory modules in sets of coregulated genes**. Bioinformatics, v. 20, n. 12, p. 1974 - 1976 , 2004.

ALLENBY, N.E.E., O'CONNOR, N., PRAGAI, Z., WARD, A.C., WIPAT, A., HARWOOD, C.R. **Genome-wide transcriptional analysis of the phosphate starvation stimulon of *Bacillus subtilis***. Journal of Bacteriology, v. 187, n. 23, p. 8063–8080, 2005.

ALTSCHUL, S.F., MADDEN, T.L., SCHÄFFER, A.A., ZHANG, J., ZHANG, Z., MILLER, W., LIPMAN, D. J. **Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.** Nucleic Acids Research, v. 25, n. 17, p. 3389–3402, 1997.

ANTELMANN, H., SCHARF, C., HECKER, M. **Phosphate starvation-inducible proteins of *Bacillus subtilis*: proteomics and transcriptional analysis.** Journal of Bacteriology, v. 182, n. 16, p. 4478–4490, 2000.

ANTELMANN, H., ENGELMANN, S., SCHMID, R. HECKER, M. **General and oxidative stress responses in *Bacillus subtilis*: cloning, expression, and mutation of the alkyl hydroperoxide reductase operon.** Journal of Bacteriology, v. 178, n. 22 , p. 6571–6578, 1996.

ASAI K., TAKAMATSU H., IWANO M., KODAMA T., WATABE K., OGASAWARA N. **The *Bacillus subtilis* yabQ gene is essential for formation of the spore cortex.** Microbiology, v. 147, p. 919 - 946, 2001.

AU N, KUESTER-SCHOECK E, MANDAVA V, BOTHWELL LE, CANNY SP, CHACHU K, COLAVITO SA, FULLER SN, GROBAN ES, HENSLEY LA, O'BRIEN TC, SHAH A, TIERNEY JT, TOMM LL, O'GARA TM, GORANOV AI, GROSSMAN AD, LOVETT CM. **Genetic composition of the *Bacillus subtilis* SOS system.** Jornal of Bacteriology, v. 187, n. 22, p. 7655 - 7721, 2005.

AZEVEDO, F.M. **Algoritmos genéticos em redes neurais artificiais**. V Escola de Redes Neurais: Conselho Nacional de Redes Neurais – ITA, São José dos Campos, SP, p. 91-121, 1999.

BAICHOO, T. WANG, R. YE, HELMANN, J.D. **Global analysis of the *Bacillus subtilis* Fur regulon and the iron starvation stimulon.** Molecular Microbiology, v. 45, n. 6, p. 1613–1629, 2002.

BAILEY,T.L., ELKAN, C. **Unsupervised learning of multiple motifs in biopolymers using expectation maximization**. Machine Learning, v. 21, p. 51–80. 1995.

BALUJA, S., CARUANA, A. **Removing the genetics from the standard genetic algorithm. Machine learning.** Proceedings of the Twelfth International Conference, Pittsburgh-Pennsylvania, p. 38 – 46, 1995.

BARNES, M.R., GRAY, I.C. **Bioinformatics for geneticists.** John Wiley & Sons Ltd, Chichester, England, 2003.

BARRETO J. M. **Conexionismo e a Resolução de Problemas**. Dissertação do Departamento de Informatica e Estatística da UFSC, Florianópolis-SC, 1996.

BELITSKY, B.R., SONENSHEIN, A.L. **Altered Transcription activation specificity of a mutant form of *bacillus subtilis* gltr, a lysr family member.** Journal of Bacteriology, v. 179, n. 4, p. 1035 - 1043, 1997.

BELITSKY, B.R., WRAY, L.V., FISHER, S.H., BOHANNON, D.E., SONENSHEIN, A.L. **Role of TnrA in Nitrogen Source-Dependent Repression of *Bacillus subtilis* Glutamate Synthase Gene Expression.** Journal of Bacteriology, v. 182, n. 21, p. 5939–5947, 2000.

BELITSKY, B.R., SONENSHEIN, A.L. **GabR, a member of a novel protein family, regulates the utilization of g-aminobutyrate in *Bacillus subtilis***. Molecular Microbiology, v. 45, n. 2, p. 569–583, 2002.

BELOIN, C., EXLEY, R., MAHE, A., ZOUINE, M., CUBASCH, S., GARAT, F. **Characterization of LrpC DNA-binding properties and regulation of *Bacillus subtilis* lrpC gene expression.** Journal of Bacteriology, v. 182, n. 16, p. 4414–4424, 2000.

BENSON, D.A., MIZRACHI, I. K., LIPMAN, D.J., OSTELL, J., WHEELER, D. L. **GenBank: update.** Nucleic Acids Research, v. 32, p. 23-26, 2004.

BESEMER, J., LOMSADZE, A., BORODOVSKY, M. **GeneMarkS: a self-training method for prediction of gene starts in microbial genomes. Implications for finding sequence motifs in regulatory regions.** Nucleic Acids Research, v. 29, n. 12, p. 2607-2618, 2001.

BLATTNER F.R., PLUNKETT G., BLOCH C.A., PERNA N.T., BURLAND V., RILEY M., COLLADO-VIDES J., GLASNER J.D., RODE C.K., MAYHEW G.F., GREGOR J., DAVIS N.W., KIRKPATRICK H.A., GOEDEN M.A., ROSE D.J., MAU B., SHAO Y. **The complete genome sequence of Escherichia coli K-12**. Science, v. 277, p. 1453-1527, 1997.

BLICKLE, T. **Tournament selection**. Handbook of evolutionary computation release 97/1, IOP Publishing Ltd and Oxford University, c. 2.3, 1997.

BOLSHOY, A., NEVO, E. **Ecologic genomics of DNA: upstream bending in prokaryotic promoters**. Genome Res. v. 10, p. 1185-1193, 2000.

BOYLAN, S.A., THOMAS, M.D., PRICE, C.W. **Genetic method to identify regulons controlled by nonessential elements: isolation of a gene dependent on alternate transcription factor sB of *Bacillus subtilis*.** J. Bacteriol. v. 173, p.7856–7866, 1991.

BROWN, T.A. **Genomes.** 2º ed. Bios Scientific Publishers, Ltd, 2002.

BURLEY, S.K., ALMO, S.C., BONANNO, J.B., CAPEL, M., CHANCE, M.R., GAASTERLAND, T., LIN, D., SALI, A., STUDIER, F.W., SWAMINATHAN, S. **Structural genomics:beyond the Human Genome Project. N**ature genetics, v. 23, 1999.

HERNANDEZ, C., SETLOW, P. **Analysis of the regulation and function of five genes encoding small, acid-soluble spore proteins of *Bacillus subtilis* A**. Gene, v. 248, n. 1-2, p. 169-181, 2000.

CAMBRIDGE UNIVERSITY. **Advanced learner´s dictionary.** Cambridge University Press, UK, p. 225, 2003.

COELHO, L.S., COELHO, A.A.R. **Algoritmos evolutivos em identificação e controle de processos: uma visão integrada e perspectivas**. Revista SBA Controle & Automação, v. 10, n. 1, p. 13-30, 1999.

CORFE B.M., MOIR A., POPHAM D., SETLOW P. **Analysis of the expression and regulation of the gerB spore germination operon of *Bacillus subtilis* 168.** Microbiology, v. 140, p. 3079 - 3162, 1994.

CRICK, F. **Central dogma of molecular bioloy**. Nature. v. 227, p. 561-563, 1970.

CUTTING S., ROELS S., LOSICK R. **Sporulation operon spoIVF and the characterization of mutations that uncouple mother-cell from forespore gene expression in *Bacillus subtilis*.** J Mol Biol, v. 221, n. 4, p. 1237-1293 , 1991

DARMON, E., NOONE, D., MASSON, A., BRON, S., KUIPERS, O.P., DEVINE, K.M., DIJL, J.M.V. **A novel class of heat and secretion stress-responsive genes is controlled by the autoregulated cssrs two-component system of *Bacillus subtilis***. Journal of Bacteriology, v. 184, n. 20, p. 5661-5671, 2002.

DARWIN, C. **A origem das espécies**. Rio de Janeiro - RJ: Ediouro, 2004.

DEB, K. **Introduction**. Handbook of evolutionary computation release 97/1, IOP Publishing Ltd and Oxford University, c. 2.1, 1997.

DIAS, J.S. e BARRETO, J.M. **Algoritmo genético: inspiração biológica na solução de problemas - uma introdução**. Revista Marítima Brasileira - Suplemento Especial, Pesquisa Naval, n. 11, p. 105-128, 1998.

DRISCOLL J.R., TABER H.W. **Sequence organization and regulation of the *Bacillus subtilis* menBE operon.** Jornal of Bacteriology, v. 174, n. 15, p. 5063- 5134, 1992.

DRZEWIECKI, K., EYMANN, C., MITTENHUBER, G., HECKER, M. **The yvyD Gene of *Bacillus subtilis* Is under Dual Control of sB and sH.** Jornal of Bacteriology, v. 180, n. 24, p. 6674–6680, 1998.

EDDY, S.R. **What is a hidden Markov model?** Nature Biotechnology, v. 22, n. 10, p. 1315-1316, 2004.

EICHENBERGER, P., JENSEN, S.T., CONLON, E.M., OOIJ1, C.V., SILVAGGI, J., PASTOR, J.E.G., FUJITA, M., YEHUDA, S.B., STRAGIER, P., LIU, J.S., LOSICK, R. **The σE regulon and the identification of additional sporulation genes in *Bacillus subtilis*.** Journal of Molecular Biology, v. 327, n. 5, p. 945-972, 2003.

EICHENBERGER P, FUJITA M, JENSEN ST, CONLON EM, RUDNER DZ, (2004) **The program of gene transcription for a single differentiating cell type during sporulation in *Bacillus subtilis*.** PLoS Biol, n. 2, v.10, p. 1664 – 1683, 2004.

ENGELMANN, S., LINDNER, C., HECKER, M. **Cloning, Nucleotide Sequence, and Regulation of katE Encoding a sB-Dependent Catalase in *Bacillus subtilis*.** Journal of Molecular Biology, v. 177, n. 19, p. 5598–5605, 1995.

EVEN, S., BURGUIE`RE, P., AUGER, S. SOUTOURINA, O., DANCHIN, A., MARTIN-VERSTRAETE, I. **Global control of cysteine metabolism by CymR in *Bacillus subtilis***. Journal of Bacteriology, v. 188, n. 6, p. 2184–2197, 2006.

FEAVERS I.M., FOULKES J., SETLOW B., SUN D., NICHOLSON W., SETLOW P., MOIR A. **The regulation of transcription of the gerA spore germination operon of *Bacillus subtilis*.** Mol Microbiol. v.4, n.2, p. 275-82, 1990.

FEUCHT, A., EVANS, L., ERRINGTON, J. **Identification of sporulation genes by genome-wide analysis of the sE regulon of *Bacillus subtilis.*** Microbiology, n. 149, p. 3023–3034, 2003.

FINKELSTEIN, A., HETHERINGTON, J., LI, L., MARGONINSKI, O., SAFFREY, P., SEYMOUR, R., WARNER, A. **Computational challenges of systems biology.** IEEE Computer Society, n. 18, p. 26-33, 2004.

FOGEL, G.B., PORTO V.W., WEEKES, D.G.; FOGEL, D.B., GRIFFEY, R.,H., MCNEIL, J.A., LESNIK, E., ECKER, D.J., SAMPATH, R. **Discovery of RNA structural elements using evolutionary computation.** Nucleic Acids Research, v. 30, n. 23, p. 5310-5317, 2002.

FOGEL, G.B.; WEEKES, D.G.; VARGA, G.; DOW, E.R; HARLOW, H.B.; ONYIA, J.E.; SU, C. **Discovery of sequence motifs related to coexpression of genes using evolutionary computation.** Nucleic Acids Research, v. 32, n. 13, p. 3826-3835, 2004.

FOGEL, D.B. **Evolutionary Computation**. IEEE Press: Piscataway, NJ, 1995.

FRITH, M.C., STORMO, G.D. **Finding functional sequence elements by multiple local alignment**. Nucleic Acids Res. v. 32, p. 189-200, 2004.

FUANGTHONG, M., HERBIG, A., BSAT, N., HELMANN, J.D. **Regulation of the *Bacillus subtilis* fur and perr genes by perr: not all members of the perr regulon are peroxide inducible**. Journal of Bacteriology, v. 184, n. 12, p. 3276–3286, 2002.

FUANGTHONG, M., HELMANN, J.D. **Recognition of DNA by three ferric uptake regulator (fur) homologs in *Bacillus subtilis***. Journal of Bacteriology, v. 185, n. 21, p. 6348 – 6357, 2003.

FURTADO, J.C. **Algoritmo Genético Construtivo na otimização de problemas combinatórios de agrupamentos.** Tese de doutorado em Computação Aplicada - Instituto Nacional de Pesquisas Espaciais, São José dos Campos-SP, 1998.

FUTUYAMA, D.J. **Biologia evolutiva**. Ribeirão Preto: FUNPEC-RP, 2003.

GABALLA, A., WANG, T., YE, R.W., HELMANN, J.D. **Unctional Analysis of the *Bacillus subtilis* Zur Regulon.** Journal of Bacteriology, v. 184, n. 23, p. 6508–6514, 2002.

GABALLA, A., CAO, M. HELMANN, J.D. **Two MerR homologues that affect copper induction of the *Bacillus subtilis* copZA operon**. Microbiology, v. 149, p. 3413–3421, 2003.

GHAHRAMANI, Z. **An introduction to hidden Markov model and bayesian networks.** International Journal of Pattern Recognition and Artificial intelligence, v. 15, n. 1, p. 9-42, 2001.

GOLDBERG, D.E. **Genetic algorithms in search, optimization, and machine learning**. Addison-Wesley, MA, 1989.

GOLDBERG, D.E. **Genetic and evolutionary algorithms come of age**. Communications of the ACM, v. 37, n. 3, p. 113-119, 1994.

GOMEZ, M., CUTTING, S.M. **Expression of the Bacillus subtilis spoIVB gene is under dual sigma F/sigma G control.** Microbiology, v. 142, p. 3453-3457, 1996.

GREFENSTETTE, J. **Rank-based selection**. Handbook of Evolutionary Computation release 97/1, IOP Publishing Ltd and Oxford University, c. 2.4, 1997.

GRUNDY, F.J., TURINSKY, A.J., HENKIN, T.M. **Catabolite regulation of *Bacillus subtilis* acetate and acetoin utilization genes by ccpa.** Jornal of Bacteriology, v. 176, n. 15, p. 4527-4533, 1994.

GORDON, L., CHERVONENKIS, A.Y., SHAHMURADOV, A.J.G.I., SOLOVYEV,V.V. **Sequence alignment kernel for recognition of promoter regions**. Bioinformatics, v. 19, n. 15, p. 1964-1971, 2003.

GRIFFITHS, A. J. F**. Introdução a Genética.** 7º ed. Guanabara Koogan: Rio de Janeiro, 2002.

GUZMAN, P., WESTPHELING, J., YOUNGMAN, D.P. **Characterization of the promoter region of the *Bacillus subtilis* spoIIE operon.** American Society for Microbiology, v. 170, n. 41, p. 1598-1609, 1998.

HANLON D.W., ROSARIO M.M., ORDAL G.W., VENEMA G., VAN SINDEREN D. **Identification of TlpC, a novel 62 kDa MCP-like protein from *Bacillus subtilis*.** Microbiology, v. 140 n. 8, p. 1847-1901, 1994.

HARVEY, L., ARNOLD, B., PAUL, M., CHRIS A. K., MONTY, K., MATTHEW P, LAWRENCE, S.Z., JAMES, D. **Molecular cell biology**. W. H. Freeman; 5 ed., 2003.

HAUPTY, R.L., HAUPTY, S.E. **Practical genetic algorithm**. 2º ed. A John Wiley & Sons, Inc., Publication, 2004.

HAY R.E., TATTI K.M., VOLD B.S., GREEN C.J., MORAN C.P. **Promoter used by sigma-29 RNA polymerase from *Bacillus subtilis*.** Gene, v. 48, p. 301-307, 1986.

HELDEN, J.V., **Regulatory sequence analysis tools.** Nucleic Acids Research, v. 31, n. 13, p. 3593-3596, 2003.

HELMANN, J.D. **Compilation and analysis of *Bacillus subtillis* σ^A-dependent promoter sequences: evidence for extended contact between RNA polymerase and upstream promoter DNA.** Nucleic Acids Research, v. 23, n. 13, p. 2351-2360, 1995.

HENRIQUES A.O., BRYAN E.M., BEALL B.W., MORAN C.P. **Cse15, cse60, and csk22 are new members of mother-cell-specific sporulation regulons in *Bacillus subtilis*.** Journal of Bacteriology, v. 179, n. 2, p. 389 - 487, 1997.

HERBIG, A.F., HELMANN, J.D. **Roles of metal ions and hydrogen peroxide in modulating the interaction of the *Bacillus subtilis* PerR peroxide regulon repressor with operator DNA**. Molecular Microbiology, v. 41, n. 4, p. 849–859, 2001.

HERMSEN, R., TANS, S., WOLDE, P.R.T. **Transcriptional regulation by competing transcription factor modules.** PLOS Computational Biology, v. 2, n. 12, p. 1552 – 1560, 2006.

HERTZ, G.Z. STORMO, G.D. **Identifying DNA and protein patterns with statistically significant alignments of multiple sequences**. Bioinformatics v. 15, p. 563-577, 1999.

HIPPLER B., HOMUTH, G., HOFFMANN T., HUNGERER, C., SCHUMANN, W., JAHN, D. **Characterization of *Bacillus subtilis* hemN**, Journal of Bacteriology, v. 179, n. 22, p. 7181–7185, 1997.

HOLLAND, J.H. **Adaptation in natural and artificial systems**. University of Michigan Press, 1975.

HOMUTH, G., MASUDA, S., MOGK, A., KOBAYASHI, Y., SCHUMANN, W. **The dnaK operon of *Bacillus subtilis* Is heptacistronic**, Journal of Bacteriology, v. 179, n. 4, p. 1153–1164, 1997.

HU, J., LI, B. KIHARA, D. **Limitations and potentials of current motif discovery algorithms.** Nucleic Acids Reseach. v. 33, n. 15, p. 4899-4913, 2005.

HUANG, X., DECATUR, A., SOROKIN, A., HELMANN, J.D. **The *Bacillus subtilis* sx protein is an extracytoplasmic functions factor contributing to survival at high temperature.** Jornal of Bacteriology, v. 179, n. 9, p. 2915–2921, 1997.

HUANG, X., HELMANN, J.D. **Identification of target promoters for the Bacillus subtilis σ^X factor using a consensus-directed search.** Journal of Molecular Biology, v. 279, n. 1, p. 165-173, 1998.

HUERTA, A.M., COLLADO-VIDES, J. **Sigma70 promoters in Escherichia coli: specific transcription in dense regions of overlapping promoter-like signals. program of computational genomics, nitrogen fixation center**. J. Mol. Biol., v. 333, n. 2, p. 261-278, 2003.

HUNTER. L. **Molecular biology for computer scientists. Artificial intelligence and molecular biology**. L. Hunter AAAI Press, Menlo Park, CA, 1993.

JACQUES, P.E., RODRIGUE, S., GAUDREAU, L., GOULET, J. **Detection of prokaryotic promoters from the genomic distribution of hexanucleotide pairs**. BMC Bioinformatics, v. 7, 2006.

JONG, K.D., FOGEL L., SCHWEFEL, H.P. **Handbook of evolutionary computation release 97/1**. IOP Publishing Ltd and Oxford University, 1997.

KANHERE, A., BANSAL, M. **A novel method for prokaryotic promoter prediction based on DNA stability.** BMC Bioinformatics, v. 6, 2005.

KANZ, C., ALDEBERT P., ALTHORPE, N., BAKER, W., BALDWIN A., BATES, K., BROWNE, P., BROEK, A.V. D., CASTRO, M., COCHRANE, G., DUGGAN, K., EBERHARDT, R., FARUQUE, N., GAMBLE, J., DIEZ, F. G., HARTE, N., KULIKOVA, T., LIN, Q., LOMBARD, V., LOPEZ, R., MANCUSO, R., MCHALE, M., NARDONE, F., SILVENTOINEN, V., SOBHANY, S., STOEHR, P., TULI. M. A., TZOUVARA, K., VAUGHAN, R., WU, D., ZHU, W., APWEILER, R. **The EMBL nucleotide sequence database**. Nucleic Acids Research, v. 33, p. 29 – 33, 2005.

KAROW, M.L., GLASER, P., PIGGOT, P.J. **Identification of a gene, spoIIR, that links the activation of SrE to the transcriptional activity of OF during sporulation in *Bacillus subtilis***. Proc. Natl. Acad. Sci, v. 92, p. 2012-2016, 1995.

KAZAKOV, A.E., CIPRIANO, M.J., NOVICHKOV, P.S., MINOVITSKY, S., VINOGRADOV, D.V., ARKIN, A., MIRONOV, A.A., GELFAND, M.S., DUBCHAK, I. **RegTransBase—a database of regulatory sequences and interactions in a wide range of prokaryotic genomes.** Nucleic Acids Res. v. 35, p. 407 – 412, 2007.

KEMP, E.H., SAMMONS, R.L. MOIR, A., SUN, D., SETLOW, P. **Analysis of Transcriptional Control of the gerD Spore Germination Gene of *Bacillus subtilis* 168.** Journal of Bacteriology, v. 173, n. 15, p. 4646 – 4652, 1991.

KIBLER, D., HAMPSON, S. E. **Learning weight matrices for identifying regulatory elements**. METMBS-2001, p. 208-214, 2001a.

KIM H.J., JOURLIN-CASTELLI C., KIM S.I., SONENSHEIN A.L. **Regulation of the *Bacillus subtilis* ccpC gene by ccpA and ccpC.** Mol Microbiol, v. 43, n. 2, p. 399 - 410, 2002.

KIBLER, D., HAMPSON, S. E. **Characterizing the Shine-Dalgarno motif: probability matrices and weight matrices**. METMBS-2001, p. 208-214, 2001b.

KOZA, J.R. **Survey of genetic algorithms and genetic programming**. Wescon® 95: E2. Neural-Fuzzy Technologies and Its Applications, p. 589-594, 1995.

KUMANO, M., FUJITA, M., NAKAMURA, K., MURATA, M., OHKI, R., YAMANE, K.. **Lincomycin resistance mutations in two regions immediately downstream of the 10 region of lmr promoter cause overexpression of a putative multidrug efflux pump in *Bacillus subtilis* mutants.** Antimicrobial Agents and Chemotherapy, v. 47, n. 1, p. 432–435, 2003.

KUNST, F., OGASAWARA, N., MOSZER, I., ALBERTINI, A.M., ALLONI, G., AZEVEDO, V., BERTERO, M.G., BESSIERES, P., BOLOTIN, A., BORCHERT, S., BORRISS, R., BOURSIER, L., BRANS, A., BRAUN, M., BRIGNELL, S.C., BRON, S., BROUILLET, S., BRUSCHI, C.V., CALDWELL, B., CAPUANO, V., CARTER, N.M., CHOI, S.K., CODANI, J.J., CONNERTON, I.F., ET AL. (126 other authors), AND DANCHIN, A. **The complete genome sequence of the gram-positive bacterium *Bacillus subtilis*.** Nature, v. 390, p. 249-256. 1997.

LARSSON J.T., ROGSTAM A., VON WACHENFELDT C. **Coordinated patterns of cytochrome bd and lactate dehydrogenase expression in *Bacillus subtilis*.** Microbiology, v. 10, p. 3323 - 3358, 2005.

LAZAREVIC V., MARGOT P., SOLDO B., KARAMATA D.J. **Sequencing and analysis of the *Bacillus subtilis* lytRABC divergon: a regulatory unit encompassing the structural genes of the N-acetylmuramoyl-L-alanine amidase and its modifier.** Gen Microbiol. v. 138, n. 9, p. 1949-2010, 1992.

LESK, A. M. **Introduction to bioinformatics**. Oxford University Press Inc., New York, United States, 2002.

LEVINE, M., TJIAN, R. **Transcription regulation and animal diversity.** Nature. v. 424, n. 6945, p. 147-151, 2003.

LI, H, RHODIUS V, GROSS C, SIGGIA ED. **Identification of the binding sites of regulatory proteins in bacterial genomes**. Proc Natl Acad Sci, v. 99, n. 18, p. 11772-11777, 2002.

LIU, X., BRUTLAG, D.L. LIU, J.S. **BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes**. Pac. Symp. Biocomput., v. 6, p. 127–138, 2001.

LIU, X.S., BRUTLAG, D.L. LIU, J.S. **An algorithm for finding protein–DNA binding sites with applications to chromatinimmunoprecipitation microarray experiments.** Nat. Biotechnol. v. 20, p. 835–839, 2002

MAKITA, Y., NAKAO, M., OGASAWARA, N., NAKAI, K. **DBTBS: database of transcriptional regulation in *Bacillus subtilis* and its contribution to comparative genomics**. Nucleic Acids Research, v. 32, p. 75 – 77, 2004.

MAKITA, Y., HOON, M.J.L., DANCHIN, A. **Hon-yaku: a biology-driven Bayesian methodology for identifying translation initiation sites in prokaryontes.** BMC Bioinformatics, v. 8, n. 47, 2007.

MATLAB. **Genetic algorithm and direct search toolbox user's guide** © COPYRIGHT 2004–2005 by The MathWorks, Inc., 2005.

MEIDANIS, J. SETUBAL, J.C. **Introduction to computational molecular biology**. Psw Publishing Company, 1997.

MEKJIAN, K.R., BRYAN, E.M., BERNARD, D., W.B., DAGGER, MORAN JR. C.P. **Regulation of hexuronate utilization in *Bacillus subtilis***. Journal of Bacteriology, v. 181, n. 2, 1999.

MICHALEWICZ, Z. **Genetic algorithm + data structures = evolution programs.** 3º ed. Springer-Verlag, 1996.

MICHALEWICZ, Z. **Constraint-preserving operators.** Handbook of evolutionary computation release 97/1, IOP Publishing Ltd and Oxford University, c. 5.5, 1997.

MIYAZAKI, S., SUGAWARA, H., IKEO, K., GOJOBORI, T., TATENO, Y. **DDBJ in the stream of various biological data**. Nucleic Acids Res., v. 32, p. 31– 34, 2004.

MORIYAMA, R., FUKUOKA, H., MIYATA, S., KUDOH, S., HATTORI, A., KOZUKA, S., YASUDA, Y., TOCHIKUBO, K., MAKINO, S. **Expression of a germination-specific amidase, sleb, of bacilli in the forespore compartment of sporulating cells and its localization on the exterior side of the cortex in dormant spores**. Journal of Bacteriology, v. 181, n. 8, p. 2373–2378, 1999.

MWANGI, M.M., SIGGIA, E.D., **Genome wide identification of regulatory motifs in Bacillus subtilis.** BMC Bioinformatics, v. 4, 2003.

NAKANO, M.M., YANG, F., HARDIN, P., ZUBER, P. **Nitrogen regulation of nasA and the nasB operon, which encode genes required for nitrate assimilation in *Bacillus subtilis*.** J Bacteriol, v. 177, n. 3, p. 573–579, 1995.

NEUWALD, A.F., LIU, J.S., LAWRENCE, C.E. **Gibbs motif sampling: detection of bacterial outer membrane protein repeats**. Protein Science, v. 4, n. 8, p. 1618-1632, 1995.

NICHOLSON, W.L., SUN, D., SETLOW, B., SETLOW, P. **Promoter specificity of og-containing rna polymerase from sporulating cells of *Bacillus subtilis*: identification of a group of forespore-specific promoters**. Journal of Bacteriology, v. 171, n. 5, p. 2708-2718, 1989.

OGASAWARA, N., MORIYA, S., YOSHIKAWA, H. **Structure and hmction of the region of the replication origin of the Bacillus subdls chromosome Structure and hmction of the region of the replication origin of the *Bacillus subdls* chromosome IV. Transcription of the oniC region and expression of DNA gyrase genes and other open reading frames**. Nucleic Acids Research, v. 13 n. 7, 1985.

OLLINGER, J., SONG, K., ANTELMANN, H., HECKER, M., ELMANN, J.D. **Role of the Fur regulon in iron transport in *Bacillus subtilis***. Journal of Bacteriology, v. 188, n. 10, p. 3664-3673, 2006.

O'REILLY M., WOODSON K., DOWDS B.C., DEVINE K.M. **The citrulline biosynthetic operon, argC-F, and a ribose transport operon, rbs, from *Bacillus subtilis* are negatively regulated by Spo0A**. Mol Microbiol, v. 11, n. 1, p. 87-98, 1994.

PEARSON, W.R., LIPMAN, D.J. **Improved tools for biological sequence comparison**. Proc Natl Acad Sci , v.85, p. 2444–2448, 1988.

PEREGO, M. HOCH, J.A. **Negative Regulation of *Bacillus subtilis* Sporulation by the spoOE Gene Productt.** Journal of Bacteriology, v. 173, n. 8, p. 2514 – 2520, 1991.

PETERSON, J.D., UMAYAM, L.A., DICKINSON, T.M., HICKEY, E.K., WHITE, O. **The comprehensive microbial resource**. Nucleic Acids Research, v. 29 p. 123-125, 2001.

PRICE, C.W., FAWCETT, P., CÉRÉMONIE, H., SU, N., MURPHY, C.K., YOUNGMAN, P. **Genome-wide analysis of the general stress response in *Bacillus subtilis*.** Molecular Microbiology, v. 41, n. 4, p. 757-774, 2001.

QUINN C.L., STEPHENSON B.T., SWITZER R.L. **Functional organization and nucleotide sequencoef the *Bacillus subtilis* pyrimidine biosynthetic operon**. Journal of Biological Chemistry, v. 266, n. 14, p. 9113 – 9127, 1991.

REENTS, H., MUNCH, R., DAMMEYER, T., JAHN D., HARTIG, E. **The Fnr regulon of *Bacillus subtilis***. Journal of Bacteriology, v. 188, n. 3, p. 1103–1112, 2006.

RÉGNIER, M. DENISE, A. **Rare events and conditional events on randon strings.** Discrete Math. Theor. Comput. Sci., v. 6, p. 191-214, 2004.

RICCA E., CUTTING S., LOSICK R. **Characterization of bofA, a gene involved in intercompartmental regulation of pro-sigma K processing during sporulation in *Bacillus subtilis*.** Jornal of Bacteriology, v. 174, n. 10, p. 3177 - 3261, 1992.

RIETHDORF, S., VOLKER, U., GERTH, U., WINKLER, A., ENGELMANN, S., HECKER, M. **Cloning, Nucleotide Sequence, and Expression of the *Bacillus subtilis* ion Gene.** Journal of Bacteriology, v. 176, n. 21, p. 6518-6527, 1994.

ROBICHON, D., ARNAUD, M., GARDAN, R., PRAGAI, Z., O'REILLY, M., RAPOPORT, G., BARBOUILLE, M. **Expression of a New Operon from Bacillus subtilis, ykzB-ykoL, under the Control of the TnrA and PhoP-PhoR Global Regulators.** Journal of Bacteriology, v. 182, n. 5, p. 1226 – 1231, 2000.

ROBISON, K., MCGUIRE, A.M., CHURCH, G.M. **A Comprehensive library of dna-binding site matrices for 55 proteins applied to the complete *Escherichia coli k-12* genome.** J. Mol. Biol, n. 284, p. 241–254, 1998.

ROSENBLUETH, D.A., THIEFFRY, D., HUERTA, M., SALGADO, H., COLLADO-VIDES, J. **Syntactic recognition of regulatory regions in *Escherichia coli*.** Bioinformatics, v.12, n. 5, p. 415-422, 1996.

ROSSOLILLO, P., MARINONI I., GALLI, E., COLOSIMO A., ALBERTINI, A.M. **YrxA is the transcriptional regulator that represses de novo nad biosynthesis in *Bacillus subtilis***. Journal of Bacteriology, v. 187, n. 20, p. 7155–7160, 2005.

REISCHL S., THAKE S., HOMUTH G., SCHUMANN W. **Transcriptional analysis of three *Bacillus subtilis* genes coding for proteins with the alpha-crystallin domain characteristic of small heat shock proteins.** Mol Microbiol, v. 194, n. 1, p. 99-103, 2001.

ROSENBLUETH, D.A., THIEFFRY, D., HUERTA, M., SALGADO, H., COLLADO-VIDES, J. **Syntactic recognition of regulatory regions in *Escherichia coli*.** Bioinformatics, v.12, n. 5, p. 415-422, 1996.

ROTH, F.P., HUGHES,J.D., ESTEP,P.W. and CHURCH,G.M. **Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation**. Nat. Biotechnol., n. 16, p. 939–945, 1998.

SATO, T., KOBAYASHI, Y. **The ars Operon in the skin Element of *Bacillus subtilis* Confers Resistance to Arsenate and Arsenite.** Jornal of Bacteriology, v. 180, n. 7, p. 1655–1661, 1998.

SATROM, P., SNEVE, R., KRISTIANSEN, K.I., SNOVE, O., GRÜNFELD, T., ROGNES, T., SEEBERG, E. **Predicting non-coding RNA genes in *Escherichia Coli* with boosted genetic programming**. Nucleic Acids Research, v. 33, n. 10, p. 3263-3270, 2005.

SAXILD, H.H., BRUNSTEDT, K., NIELSEN, K.I., JARMER, H., NYGAARD2, P. **Definition of the *Bacillus subtilis* PurR operator using genetic and bioinformatic tools and expansion of the purr regulon with glya, guac, pbug, xpt-pbux, yqhz-fold, and pbuo**. Journal of Bacteriology, v. 183, n. 21, p. 6175–6183, 2001.

SCHUCH, R., PIGGOT, P.J. **The dacF-spoIL4 operon of *Bacillus subtilis*, encoding cF, is autoregulated**. Journal of Bacteriology, v. 176, n. 13, p. 4104-4110, 1994.

SCHULZ, A., SCHWAB, S., HOMUTH, G., VERSTEEG, S., SCHUMANN, W. **The htpG gene of *Bacillus subtilis* belongs to class iii heat shock genes and is under negative control**. Journal of Bacteriology, v. 179, n. 10, p. 3103–3109, 1997.

SEKIGUCHI, J., AKEO, K., YAMAMOTO, H.I., KHASANOV, F.K., ALONSO, J.C., KURODA, A. **Wall Hydrolase Gene, cwlD, Which Affects Germination in *Bacillus subtilis*.** Journal of Bacteriology, v. 177, n. 19, p. 5582–5589, 1995.

SERIZAWA, M., YAMAMOTO, H. YAMAGUCHI, H., KOBAYASHI, Y.F.K., OGASAWARA, N., SEKIGUCHI, J. **Systematic analysis of SigD-regulated genes in *Bacillus subtilis* by DNA microarray and Northern blotting analyses.** Gene, v. 329, n. 31, p. 125-136, 2004.

SIMPSON, E.B., HANCOCK, T.W., BUCHANAN, C.E. **Transcriptional Control of dacB, Which Encodes a Major Sporulation-Specific Penicillin-Binding Protein.** Journal of Bacteriology, v. 176, n. 24, p. 7767-7769, 1994.

SINDEREN D., KIEWIET R., VENEMA G. **Differential expression of two closely related deoxyribonuclease genes, nucA and nucB, in *Bacillus subtilis*.** Mol Microbiol, v.15, n.2, p. 213-23, 1995.

SHIN, B.S., STEIN, A., ZALKIN, H. **Interaction of *Bacillus subtilis* purine repressor with DNA**. Journal of Bacteriology, v. 179, n. 23, p. 7394–7402, 1997.

SHINE e DELGARNO, L. **The 3'-terminal sequence of *Escherichia coli* 16S ribosomal rna: complementarity to nonsense triplets and ribosome binding sites j.** Proc. Nat. Acad. Sci., v. 71, n. 4, p. 1342-1346, 1974.

SINDEREN, D., VENEMA, G. **comK Acts as an Autoregulatory Control Switch in the Signal Transduction Route to Competence in Bacillus subtilis**. Journal of Bacteriology, v. 176, n. 18, p. 5762-5770, 1994.

SIVARAMAN, K., SESHASAYEE, A.S.N., SWAMINATHAN, K. MUTHUKUMARAN, G., PENNATHUR, **G. Promoter addresses: revelations from oligonucleotide profiling applied to the *Escherichia coli* genome.** Theor Biol Med Model, v. 2, n. 20, p. 2-20, 2005.

SLACK F.J., MUELLER J.P., STRAUCH M.A., MATHIOPOULOS C., SONENSHEIN A.L. **Transcriptional regulation of a *Bacillus subtilis* dipeptide transport operon.** Mol Microbiol, v. 5 n. 8, p. 1915 - 1940, 1991.

SMITH, M.C.M., MOUNTAIN, A., BAUMBERG, S. **Sequence analysis of the *Bacillus subtilis* argC promoter region information**. Gene, v. 49, n. 1, p. 53-60, 1986.

SRINIVAS, M., PATNAIK, L. M. **Adaptive probabilities of crossover and mutation in Genetic Algorithms**. IEEE Transations on Systems, Man and Cybernetics, v. 24, n. 4, p. 656-666, 1994a.

SRINIVAS, M. & PATNAIK, L.M. **Genetic algorithms: A survey.** IEEE Computer Society, v. 27, n. 6, p. 17–26, 1994b.

SUN, D., SETLOW, P. **Cloning, nucleotide sequence, and regulation of the *Bacillus subtilis* nadb gene and a nifs-like gene, both of which are essential for nad biosynthesis**. Journal of Bacteriology, v. 175, n. 5, p. 1423-1432, 1993.

STEARNS, S.C. **Evolução: uma introdução**. Atheneu: São Paulo, 2003.

STEINMETZ, M., COQ, D.L., AYMERICH, S. **Induction of saccharolytic enzymes by sucrose in *Bacillus subtilis*: evidence for two partially interchangeable regulatory pathways.** Journal of Bacteriology, v. 171, n. 3, p. 1519-1523, 1989.

STUDHOLME, D.J., BENTLEY, S.D., KORMANEC, J. **Bioinformatic identification of novel regulatory DNA sequence motifs in *Streptomyces coelicolor*.** BMC Microbiology, v.4, n. 14, 2004.

STÜLKE, J, MARTIN-VERSTRAETE, I, ZAGOREC, M, ROSE, M, KLIER, A, RAPOPORT, G. **Induction of the *Bacillus subtilis* ptsGHI operon by glucose is controlled by a novel antiterminator, GlcT.** Mol Microbiol. v. 25, n. 1, p. 65-78, 1997.

SUN, D. MARTHA, R., MARTINEZ, C., SETLOW, P. **Control of Transcription of the *Bacillus subtilis* spoIIIG Gene, Which Codes for the Forespore-Specific Transcription Factor of.** Journal of Bacteriology, v. 173, n. 9, p. 2977-2984, 1991.

SUSSMAN, M. D., SETLOW, P. **Cloning, nucleotide sequence, and regulation of the Bacillus subtilis gpr gene, which codes for the protease that initiates degradation of small, acid-soluble proteins during spore germination.** Jornal of Bacteriology, v. 173, n. 1, p. 291– 300, 1991.

SYSWERDA, G. **Uniform crossover in genetic algorithms.** Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann Puplishers, p. 2-9, 1989.

TATUSOV R.L., FEDOROVA N.D., JACKSON J.D., JACOBS A.R., KIRYUTIN B., KOONIN E.V., KRYLOV D.M., MAZUMDER R., MEKHEDOV S.L., NIKOLSKAYA A.N., RAO B.S., SMIRNOV S., SVERDLOV A.V., VASUDEVAN S., WOLF Y.I., YIN J.J., NATALE D.A. **The COG database: an updated version includes Eukaryotes.** BMC Bioinformatics, v. 4, 2003.

TERAI, G., TAKAGI, T., NAKAI, K. **Prediction of co-regulated genes in *Bacillus subtilis* on the basis of upstream elements conserved across three closely related species.** Genome Biology, v. 2, n. 11, p. 1 – 12, 2001.

THIJS G., MARCHAL, K., LESCOT, M., ROMBAUTS, S., DEMOOR, B., ROUZE, P., MOREAU,Y. **A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes**. J. Comput. Biol, v. 9, n. 2, p. 447–464, 2002.

TOMPA, M., LI, N., BAILEY, T.L., CHURCH, G.M., MOOR, B. D., ESKIN, E., FAVOROV, A.V., FRITH, M.C., FU, Y., KENT, J., MAKEEV V.J., MIRONOV, A.A., NOBLE, S.W., PAVESI, G., PESOLE, G., RÉGNIER, M., SIMONIS, N., SINHA, S., THIJS, G., HELDEN, J.V., VANDENBOGAERT, M., WENG, ZHIPING, WORKMAN, C., YE, C., ZHU, Z. **Assessing computational tools for the discovery of transcription factor binding sites**. Nature Biotechnology, v. 23, n. 1, 2005.

VARÓN, D., BRODY, M.S., PRICE, W. ***Bacillus subtilis* operon under the dual control of the general stress transcription factor sigma B and the sporulation transcription factor sigma H**. Mol. Microbiol. n 20, p. 339-350, 1996.

VASCONCELOS A.T., FERREIRA H.B., BIZARRO C.V., BONATTO S.L., CARVALHO M.O., PINTO P.M., ALMEIDA D.F., ALMEIDA L.G., ALMEIDA R., ALVES-FILHO L., ASSUNCAO E.N., AZEVEDO V.A., BOGO M.R., BRIGIDO M.M., BROCCHI M., BURITY H.A., CAMARGO A.A., CAMARGO S.S., CAREPO M.S., CARRARO D.M., DE MATTOS CASCARDO J.C., CASTRO L.A., CAVALCANTI G., CHEMALE G., COLLEVATTI R.G., CUNHA C.W., DALLAGIOVANNA B., DAMBROS B.P., DELLAGOSTIN O.A., FALCAO C., FANTINATTI-GARBOGGINI F., FELIPE M.S., FIORENTIN L., FRANCO G.R., FREITAS N.S., FRIAS D., GRANGEIRO T.B., GRISARD E.C., GUIMARAES C.T., HUNGRIA M., JARDIM S.N., KRIEGER M.A., LAURINO J.P., LIMA L.F., LOPES M.I., LORETO E.L., MADEIRA H.M., MANFIO G.P., MARANHAO A.Q., MARTINKOVICS C.T., MEDEIROS S.R., MOREIRA M.A., NEIVA M., RAMALHO-NETO C.E., NICOLAS M.F., OLIVEIRA S.C., PAIXAO R.F., PEDROSA F.O., PENA S.D., PEREIRA M., PEREIRA-FERRARI L., PIFFER I., PINTO L.S., POTRICH D.P., SALIM A.C., SANTOS F.R., SCHMITT R., SCHNEIDER M.P., SCHRANK A., SCHRANK I.S., SCHUCK A.F., SEUANEZ H.N., SILVA D.W., SILVA R., SILVA S.C., SOARES C.M., SOUZA K.R., SOUZA R.C., STAATS C.C., STEFFENS M.B., TEIXEIRA S.M., URMENYI T.P., VAINSTEIN M.H., ZUCCHERATO L.W., SIMPSON A.J., ZAHA A. **Swine and poultry pathogens: the complete genome sequences of two strains of *Mycoplasma hyopneumoniae* and a strain of *Mycoplasma synoviae*.** Journal of Bacteriology, v. 187, n. 16, p. 5568–5577, 2005.

WALKER J.D.; FILE, P.E.; MILLER, C.J.; SAMSON, W.B. **Building DNA maps, a genetic algorithm based approach. Advances in molecular bioinformatics**. S. Schulze-Kremer, IOS Press. p. 179-199, 1994

WANG S.T., SETLOW B., CONLON E.M., LYON J.L., IMAMURA D., SATO T., SETLOW P., LOSICK R., EICHENBERGER. **The forespore line of gene expression in *Bacillus subtilis.*** J Mol Biol, v. 358, n. 1, p. 16-37, 2006.

WENG, M., NAGY, P.L., ZALKIN, H. **Identification of the *Bacillus subtilis* pur operon repressor (purine repressor/gene regulation/protein-DNA interaction/adenine phosphoribosyltransferase/phosphoribosyl pyrophosphate)**. Proc. Nat. Acad. Sci., v. 92, p. 7455-7459, 1995.

WETZSTEIN, M., VOLKER, U., DEDIO, J., LOBAU, S., ZUBER, U., SCHIESSWOHL, M., HERGET, C., HECKER, M., SCHUMANN, W. **Cloning, sequencing, and molecular analysis of the dnaK locus from *Bacillus subtilis.*** Journal of Bacteriology, v. 174, n. 10, p. 3300-3310, 1992.

WHITLEY, D. **A genetic algorithm tutorial**. Springer Science + Business Media B.V., Formerly Kluwer Academic. p. 65-85, 1994.

WISE, A.A., e PRICE, C.W. **Four additional genes in the sigB operon of *Bacillus subtilis* that control activity of the general stress factor sB in response to environmental signals**. Journal of Bacteriology, v. 177, n. 1, p. 123–133, 1995.

WRAY JR, L.V. FISHER, S.H.**. Analysis of *Bacillus subtilis* hut operon expression indicates that histidine-dependent induction is mediated primarily by transcriptional antitermination and that amino acid repression is mediated by two mechanisms: regulation of transcription initiation and inhibition of histidine transport**. Journal of Bacteriology, v. 176, n. 17, p. 5466-5473, 1994.

WRAY, JR., PElTTENGILL, F.K., FISHER, S.H. **Catabolite Repression of the Bacillus subtilis hut Operon Requires a cis-Acting Site Located Downstream of the Transcription Initiation Site.** Journal of Bacteriology, v. 176, n. 7, p. 1894-1902, 1994b.

WU, J.J., SCHUCH, R., PIGGOT, P.J. **Characterization of a *Bacillus subtilis* sporulation operon that includes genes for an rna polymerase a factor and for a putative dd-carboxypeptidase.** Journal of Bacteriology, v. 174, n. 15, p. 4885-4892, 1992.

YAMAMOTO, H., SERIZAWA, M., THOMPSON, J., SEKIGUCHI, J. **Regulation of the glv Operon in Bacillus subtilis: YfiA (GlvR) Is a Positive Regulator of the Operon That Is Repressed through CcpA and cre.** Journal of Bacteriology, v. 183, n. 17, p. 5110–5121, 2001.

YORK, K., KENNEY, T.J., SATOLA, S., MORAN JR., C.P., POTH,'t, H., YOUNGMAN, P. **SpoOA controls the &-a-dependent activation of *Bacillus subtilis* sporulation-specific transcription unit spoiie**. Journal of Bacteriology, v. 174, n. 8, p. 2648 – 2658, 1992.

YOSHIDA, K., AOYAMA, D., ISHIO, I. SHIBAYAMA, T., FUJITA, Y. **Organization and Transcription of the myo-Inositol Operon, iol, of Bacillus subtilis.** Journal of Bacteriology, v. 179, n. 14, p. 4591–4598, 1997

YOSHIDA, K., OHKI, Y., MURATA, M., KINEHARA, M., MATSUOKA, H., I SATOMURA, T., OHKI, R., KUMANO, M., YAMANE, K., FUJITA Y. ***Bacillus subtilis* LmrA is a repressor of the lmrab and yxagh operons: identification of its binding site and functional analysis of lmrb and yxagh**. Journal of Bacteriology, v. 186, n. 17, p. 5640–5648, 2004.

ZHENG, G., YAN, L.Z., VEDERAS, J.C., ZUBER, P. **Genes of the sbo-alb locus of *Bacillus subtilis* are required for production of the antilisterial bacteriocin subtilosin.** Journal of Bacteriology, v. 181, n. 23, p. 7346–7355, 2000.

PRINCIPAIS *LINKS* ACESSADOS

ATTESON, K. **Weight matrix**. Center for Molecular Medicine, Yale University School of Medicine, 1998. Disponível em <http://www.med.yale.edu/bcmm/Informatics/Jan20/weight.htm>. Acesso em 09/10/2006.

BLACK, P.E., **Dictionary of algorithms and data structures** [online]. U.S. National Institute of Standards and Technology, 2006. Disponível em <http://www.nist.gov>. Acesso em 15/01/2007.

BLAST. Disponível em <http://www.ncbi.nlm.nih.gov/BLAST/>. Acesso em 15/11/2006.

BOCKHOLT, B. **"Exhaustive search"**, in Dictionary of Algorithms and Data Structures [online]. U.S. National Institute of Standards and Technology, 2004. Disponível em <http://www.nist.gov/dads/HTML/exhaustiveSearch.html>. Acesso em 08/10/2006.

CMR. Disponível em <http://www.tigr.org/CMR>. Acesso em 15/11/2006.

COG. Disponível em <http://www.ncbi.nlm.nih.gov/COG/>. Acesso em 15/11/2006.

DBTBS. Disponível em <http://dbtbs.hgc.jp/> . Acesso em 07/01/2006.

EMBL. Disponível em <http://www.ebi.ac.uk/embl>. Acesso em 15/11/2006.

FASTA. Disponível em <http://www.ebi.ac.uk/fasta33/>. Acesso em 15/11/2006.

FREITAS, J.B. **Modelos Ocultos de Markov**. Universidade Católica de Pelotas (UCPel), 2002. Disponível em <http://descartes.ucpel.tche.br/WFC/2002/06-mom.pdf>. Acesso em 12/02/2007.

GENBANK. Disponível em <http://www.ncbi.nlm.nih.gov/Genbank/index.html>. Acesso em 15/08/2006.

NHGRI (National Human Genome Research Institute). **Talking glossary of genetic terms**. Disponível em <http://www.genome.gov/glossary.cfm>. Acesso em 23/01/2007.

RSAT. Disponível em <http://rsat.ulb.ac.be/rsat/>. Acesso em 05/01/2006.

WIKIPEDIA. **Sequence motif.** Disponível em <http://en.wikipedia.org/wiki/DNA_motif> . Acesso em 10/06/2007.

ZUBEN, F.J.V. **Computação evolutiva: uma abordagem pragmática**. Unicamp-SP+, 2000. Disponível em <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/tutorial/tutorialEC.pdf>. Acesso em 12/12/2006.

GLOSSÁRIO DE TERMOS DA BIOLOGIA

- ***Bacillus subtilis***: É uma espécie de bactéria gram-positiva comum do solo e da água. Organismo que pode formar esporo, não patogênico. Seu genoma contém 4.214.810 pares de bases, compreendendo 4100 genes (KUNST, et al., 1997) .

- **Bases**: O DNA é formado basicamente de moléculas de açúcar, moléculas de fosfato e moléculas chamadas de bases. As bases nitrogenadas são usualmente representadas por "letras" que identificam o código genético. No DNA existem quatros letras A, C, G e T, que são Adenina, Citosina, Guanina e Tinima, respectivamente. Estas bases sempre formam os pares AT e CG (NHGRI, 2007).

- **Genes**: Unidade hereditária funcional e física passada de pai para filho. Genes são partes do DNA que contém as informações necessárias para codificar uma proteína específica (NHGRI, 2007).

- ***Motif***: É uma seqüência comum de nucleotídeos ou aminoácidos que tem alguma significância biológica (WIKIPEDIA, 2007).

- ***Upstream***: Termo usado para designar as bases que ficam antes do início da região transcrita de um gene (BOLSHOY e NEVO, 2000).

- **RSATools**: O "*Regulatory Sequence Analysis Tools*" é um conjunto de ferramentas dedicadas a extrair informações de regiões "*upstream*" de vários organismos. Essas ferramentas podem ser acessadas pela internet no *site* http://RSATools.ulb.ac.be/RSATools/ (HELDEN, 2003).

- **Oligômeros:** São pequenas seqüências de bases em uma fita de DNA ou RNA (NHGRI, 2007).

- **TSS**: "*Transcription Start Site*" é o local onde inicia-se o processo de transcrição genética, que é a cópia, com o auxílio da molécula de RNA polimerase, de uma seqüência do DNA, para produzir uma molécula de RNA (LEVINE e TJIAN, 2003).

- **Super-representada**: É a tradução do termo em inglês "*over-represented*", que significa que um determinado oligômero repete-se em uma seqüência de DNA acima de um limiar de freqüência de ocorrência arbitrário (MWANGI e SIGGIA, 2003).

GLOSSÁRIO DE TERMOS DE INFORMÁTICA

- **BitString**: Estrutura de dados usada no Matlab para representar números binários (MATLAB, 2005).

- **MatLab**: É uma linguagem de programação de alto desempenho para computação técnica. Toda a programação é desenvolvida em um ambiente amigável, onde os problemas e programações são escritos em notações matemáticas familiares. Um dos pontos fortes da linguagem, é a disponibilidade de funções de manipulação de matrizes e as "*toolbars*" implementadas. Um exemplo de "*toolbar*" é o gatool, ferramenta pronta para trabalhar com algoritmo genético (MATLAB, 2005).

- **Matriz**: Vetor de duas dimensões. Por convenção, o primeiro índice é a linha e o segundo é a coluna (BLACK, 2006).

- **Vetor**: Conjunto de itens que são acessados por um índice numérico (BLACK, 2006).