ELISA BOARI DE LIMA

UMA METODOLOGIA PARA IDENTIFICAÇÃO DE MÓDULOS FORMADORES DE SEQUÊNCIAS DE PROTEÍNAS MOSAICAS DO *Trypanosoma cruzi* A PARTIR DO TRANSCRIPTOMA DO PARASITO UTILIZANDO A FERRAMENTA BLAST

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

LAVRAS
MINAS GERAIS – BRASIL
2008

ELISA BOARI DE LIMA

UMA METODOLOGIA PARA IDENTIFICAÇÃO DE MÓDULOS FORMADORES DE SEQUÊNCIAS DE PROTEÍNAS MOSAICAS DO *Trypanosoma cruzi* A PARTIR DO TRANSCRIPTOMA DO PARASITO UTILIZANDO A FERRAMENTA BLAST

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de concentração:

Bioinformática

Orientador:

Prof. Thiago de Souza Rodrigues

LAVRAS
MINAS GERAIS – BRASIL
2008

Ficha Catalográfica preparada pela Divisão de Processo Técnico da Biblioteca Central da UFLA

Lima, Elisa Boari de

Uma Metodologia para Identificação de Módulos Formadores de Sequências de Proteínas Mosaicas do *Trypanosoma cruzi* a partir do Transcriptoma do Parasito Utilizando a Ferramenta BLAST / Elisa Boari de Lima – Minas Gerais, 2008. 53p.

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

1. Bioinformática. 2. Proteínas Mosaicas. 3. *Trypanosoma cruzi*. 4. Transcriptoma. 5. BLAST. I. LIMA, E. B. II. Universidade Federal de Lavras. III. Título.

ELISA BOARI DE LIMA

UMA METODOLOGIA PARA IDENTIFICAÇÃO DE MÓDULOS FORMADORES DE SEQUÊNCIAS DE PROTEÍNAS MOSAICAS DO *Trypanosoma cruzi* A PARTIR DO TRANSCRIPTOMA DO PARASITO UTILIZANDO A FERRAMENTA BLAST

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em 18 de Novembro de 2008

_____
Prof. Joaquim Quinteiro Uchôa

_____
Profa. Marluce Rodrigues Pereira

_____
Prof. Thiago de Souza Rodrigues
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

Dedico este trabalho àqueles que sempre acreditaram em mim: meus pais.

Agradeço a Deus por sempre me guiar.

À minha família, em especial meus pais José Maria de Lima e Annete de Jesus Boari Lima, por me apoiarem durante mais esta jornada. Amo muito vocês.

Aos amigos, por me ajudarem a superar os obstáculos encontrados.

Aos colegas, por caminharem ao meu lado, pois nem todos os caminhos são para todos os viajantes.

Aos mestres, por partilharem seus conhecimentos.

Aos orientadores, Thiago e Thelma, pelo incentivo e motivação.

E a todos que participaram, torceram e acreditaram nesta conquista.

Obrigada!

"Para grandes realizações devemos não somente agir, mas também sonhar; não somente planejar, mas também acreditar" Anatole France

UMA METODOLOGIA PARA IDENTIFICAÇÃO DE MÓDULOS FORMADORES DE SEQUÊNCIAS DE PROTEÍNAS MOSAICAS DO *Trypanosoma cruzi* A PARTIR DO TRANSCRIPTOMA DO PARASITO UTILIZANDO A FERRAMENTA BLAST

RESUMO

Este trabalho propôs uma metodologia de identificação de módulos formadores de sequências nucleotídicas codificadoras de proteínas mosaicas do *Trypanosoma cruzi* utilizando a ferramenta BLAST. Para o desenvolvimento da metodologia, foi utilizada a família MASP de proteínas e aplicado inicialmente o conjunto de valores padrão dos parâmetros da ferramenta. Posteriormente foram estudadas diferentes combinações de valores de parâmetros a fim de comparação de resultados, incluindo valores indicados pela literatura. A metodologia desenvolvida provou ser eficaz para o objetivo proposto, obtendo melhores resultados quando aplicados valores diferentes dos valores padrão para filtro de regiões de baixa complexidade, *E-value* e tamanho inicial de palavra.

Palavras-chave: Bioinformática, Proteínas Mosaicas, *Trypanosoma cruzi*, Transcriptoma, BLAST.

A METHODOLOGY FOR IDENTIFICATION OF COMPONENT MODULES OF *Trypanosoma cruzi* MOSAIC PROTEIN SEQUENCES FROM THE PARASITE'S TRANSCRIPTOME USING BLAST

ABSTRACT

This paper proposed a methodology for the identifying component modules of nucleotide sequences that code *Trypanosoma cruzi* mosaic proteins using BLAST. For the development of the methodology, MASP protein family was used and a set of default BLAST parameter values was initially applied. Afterwards, different combinations of parameter values were studied for result comparison, including those indicated in literature. The developed methodology proved to be efficient for the proposed objective, obtaining better results when non-default parameter values for low complexity region filter, *E-value* and initial word size were applied.

Keywords: Bioinformatics, Mosaic Proteins, *Trypanosoma cruzi*, Transcriptome, BLAST.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1. INTRODUÇÃO

1.1. Contextualização e Motivação

O genoma é um depósito de informações biológicas. O Projeto Genoma, possibilitado pelo avanço tecnológico das últimas décadas, tem como principais objetivos o sequenciamento de todo o Ácido Desoxirribonucléico (DNA) do genoma, a criação de mapas físicos de alta resolução que descrevem as características químicas da molécula de DNA, a criação de bancos de dados para armazenamento das informações obtidas e o aperfeiçoamento de técnicas moleculares de modo a promover a melhoria da qualidade dos estudos envolvendo genoma. Por se tratar de bancos de dados muito extensos, é indispensável a utilização de plataformas computacionais eficientes para se realizar a análise dos dados e a interpretação dos resultados.

Pela diversidade e inter-relacionamento dos dados biológicos provenientes de estudos genômicos, esses são relativamente complexos em comparação àqueles derivados de outras áreas científicas. A comunidade científica busca a compreensão do conjunto de peças atuantes no funcionamento complexo de todo o organismo a partir do conhecimento fundamental do genoma. No entanto, isso é somente parcialmente possível no momento. Procura-se entender as estruturas moleculares das proteínas e suas interações, entre si e com diferentes moléculas biológicas (DNA, carboidratos, lipídios), além de obter conhecimento sobre o papel da variabilidade genética representada pelas várias formas de cada proteína e sobre as diversas vias metabólicas celulares.

Somente com o apoio da Informática toda a informação gerada pela ciência genômica pode ser organizada, analisada e interpretada. Por isso, a Bioinformática é indispensável para a manipulação de dados biológicos. Pode-se definir Bioinformática como uma modalidade que envolve todos os aspectos de aquisição, processamento, armazenamento, distribuição, análise e interpretação da informação biológica. Diversas ferramentas que auxiliam na compreensão do significado biológico representado pelos dados genômicos são elaboradas por meio da combinação de procedimentos e técnicas da Matemática, Estatística e Ciência da Computação. Além disso, a Bioinformática acelera a investigação em outras áreas como Medicina, Biotecnologia e Agronomia por meio da criação de bancos de dados com as informações já processadas.

Com o apoio da Bioinformática, foram iniciadas pesquisas sobre a biologia molecular do *Trypanosoma cruzi* (*T. cruzi*) em diversos centros de pesquisas simultaneamente, em especial no Brasil e na Argentina. O *T. cruzi* é um protozoário causador da Doença de Chagas, uma doença sem cura e debilitante que atinge milhões de pessoas na América Latina. O sequenciamento do genoma desse parasito permitiu o início de análises de sequências de nucleotídeos e aminoácidos derivadas buscando a identificação de novos alvos terapêuticos potenciais e, além disso, fornece dados estruturais para estudos funcionais posteriores como os dados sobre módulos presentes em proteínas mosaicas, que são formadas pelo rearranjo genético desses. Módulos podem ser definidos como conjuntos de aminoácidos invariáveis ou altamente conservados usados como "blocos de construção" em diversas proteínas. Cada módulo pode apresentar uma diferente função enzimática, sinalizadora, regulatória ou estrutural, o que faz com que a estrutura modular de proteínas permita sua evolução com funções complexas e altamente especializadas.

A infecção do hospedeiro pelo *T. cruzi* se dá por meio de estratégias adaptativas envolvendo diferentes famílias de proteínas de superfície, entre elas a família de Proteínas de Superfície Associadas a Mucinas (MASP – *Mucin Associated Surface Proteins*) em estudo. Em proteínas desse tipo é encontrado um número de diferentes módulos, no entanto na literatura não existem estudos que verifiquem a estrutura mosaica das proteínas da família MASP do *T. cruzi*.

O tratamento da Doença de Chagas é limitado a medicamentos utilizados desde o final da década de 1960 devido à grande variabilidade clínica e epidemiológica da doença e às características genéticas da população do *T. cruzi*. Esses medicamentos apresentam altas taxas de efeitos colaterais e eficácia variável durante a fase crônica da doença. Por essa razão, a identificação dos módulos constituintes das proteínas de famílias protéicas necessárias à sobrevida e à patogenicidade do parasito por meio da análise de seu transcriptoma abre caminho para a busca de novas estratégias terapêuticas e para a identificação de biomarcadores novos importantes para o desenvolvimento de novas drogas e prognóstico clínico da Doença de Chagas.

1.2. Objetivos

Este trabalho teve como objetivo geral o desenvolvimento de uma metodologia para identificar módulos formadores de sequências nucleotídicas codificadoras de proteínas

mosaicas e, dada uma família de proteínas do *T. cruzi*, para verificar se essas proteínas apresentam estrutura mosaica, ou seja, se são formadas por módulos que se repetem em diferentes proteínas da família. Para o desenvolvimento da metodologia de identificação de módulos foi utilizada a família MASP de proteínas do *T. cruzi*.

Os objetivos específicos deste trabalho são os seguintes:

- Desenvolvimento de um algoritmo para identificar módulos comuns a diversas proteínas de uma família protéica a partir das sequências nucleotídicas que as codificam;
- Aplicação do algoritmo desenvolvido para identificação dos módulos presentes nas proteínas da família MASP em estudo;
- Identificação, para cada módulo encontrado, das sequências da família protéica em questão que o apresentam e mapeamento da posição em que ocorre em tais sequências;
- Identificação, para cada sequência da família protéica em questão, dos módulos que ela apresenta e mapeamento das posições em que ocorrem;
- Análise e discussão dos resultados encontrados e da metodologia desenvolvida.

1.3. Estrutura do Trabalho

Este trabalho se encontra dividido em seus capítulos da seguinte forma:

- No Capítulo Dois são apresentados os conceitos da Biologia tomados como necessários para o melhor entendimento deste trabalho e dos ganhos obtidos;
- No Capítulo Três são explanados conceitos e técnicas da Bioinformática utilizados para realização do trabalho;
- No Capítulo Quatro são expostas a classificação da pesquisa e a metodologia utilizada para o desenvolvimento do trabalho;
- No Capítulo Cinco são apresentados resultados e discussão desses;
- No Capítulo Seis são apresentadas as conclusões do trabalho e propostas de continuidade do mesmo.

2. REFERENCIAL BIOLÓGICO

2.1. O *Trypanosoma cruzi*

O *T. cruzi* pertence à ordem Kinetoplatida, que compreende as famílias Bodonidae Hollande e Trypanosomatidae Kent. Essas famílias englobam flagelados com um ou dois flagelos que se originam de uma abertura conhecida como bolsa flagelar e normalmente contêm uma estrutura paraflagelar e uma estrutura proeminente conhecida como cinetoplasto, que corresponde a uma condensação de DNA localizado no interior de uma mitocôndria única e ramificada por todo o corpo do protozoário. A família Trypanosomatidae inclui os gêneros: *Blastocrithidia, Crithidia, Endotrypanum, Herpetomonas, Leishmania, Leptomonas, Phytomonas* e *Trypanosoma* (SOUZA, 2008).

O gênero *Trypanosoma* é um dos mais importantes dentro da família Trypanosomatidae por incluir uma série de espécies causadoras de doenças humanas como o *T. cruzi* (Figura 2.1), agente da doença de Chagas, *Trypanosoma rhodesiense* e *Trypanosoma gambiense*, agentes da doença do sono, e de doenças de animais como *Trypanosoma brucei* (*T. brucei*), *Trypanosoma equiperdum* e *Trypanosoma equinum*. Com base no comportamento do parasito nos seus hospedeiros, principalmente no vetor, o gênero *Trypanosoma* foi dividido em dois grupos: Stercoraria e Salivaria. O *T. cruzi* se inclui no grupo Stercoraria, que inclui tripanosomos que se desenvolvem no tubo digestivo do vetor, progredindo no sentido da porção intestinal com liberação de formas infectantes pelas fezes (SOUZA, 2008).

Figura 2.1 – O *Trypanosoma cruzi* Rodeado por Glóbulos Vermelhos
Fonte: Levy (2006)

A Doença de Chagas, ou tripanossomíase americana, descoberta em 1909 por Carlos Chagas em Lassance, MG, tem como agente etiológico o *T. cruzi* (NEVES et al., 2005). Esse parasito, cuja presença no planeta remonta há mais de 150 milhões de anos, é largamente distribuído na natureza e sua circulação ocorre entre insetos vetores e mamíferos silvestres. O *T. cruzi* é dotado de grande diversidade genética, de modo que suas populações e clones estudados têm em geral sido agrupados segundo estudos de perfil molecular e isoenzimático em três grupos ou linhagens maiores denominados GI, GIII (grupos basicamente de origem silvestre, vinculados naturalmente a marsupiais) e Z2 (encontrado na Doença de Chagas Humana da América do Sul e naturalmente ligado a primatas) (DIAS, 2006).

O *T. cruzi* infecta e se adapta ao hospedeiro vertebrado explorando estratégias evolucionárias para invadir células alvo e evadir (ou confundir) o sistema imunológico (ANDRADE & ANDREWS, 2005). O processo de invasão, evasão e infecção envolve diferentes famílias de proteínas de superfície (FRASCH, 2000). Uma estratégia chave é a geração e apresentação de antígenos de superfície variáveis (KAHN et al., 1999). O parasito pode tirar vantagem dessa estratégia para aderir a diferentes moléculas na membrana celular e matriz extracelular da célula hospedeira (FRASCH, 2000).

2.2. Expressão do Genoma

O genoma contém a informação biológica necessária para construir e manter um exemplar vivo de todo e qualquer organismo. No entanto, o genoma sozinho não é capaz de liberar essa informação para a célula, sendo necessária para isso uma série complexa de reações bioquímicas chamada expressão genômica (BROWN, 2002).

2.2.1. **Nucleotídeos**

DNA e Ácido Ribonucléico (RNA) são polímeros lineares não ramificados cujas subunidades monoméricas são quatro nucleotídeos quimicamente distintos que podem ser ligados em qualquer ordem em cadeias com centenas, milhares ou mesmo milhões de unidades de comprimento (BROWN, 2002). Nucleotídeos são componentes orgânicos constituídos de três estruturas combinadas: uma base nitrogenada, um açúcar e um grupo fosfato. O açúcar é uma ribose no RNA e uma desoxirribose no DNA (HIB et al., 2003). A estrutura de um nucleotídeo pode ser observada na Figura 2.2.

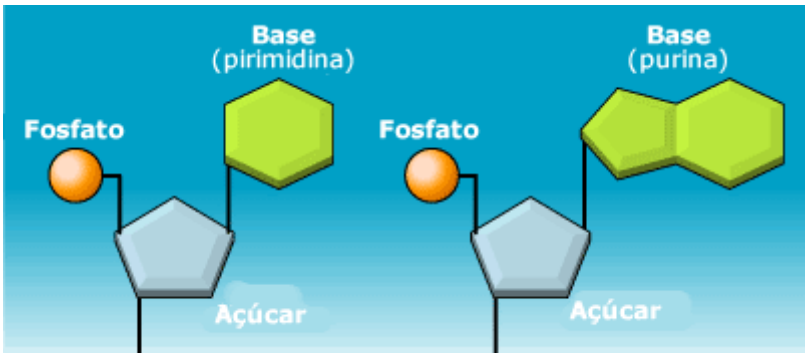


Figura 2.2 – Estrutura dos Nucleotídeos

Fonte: Freudenrich (2008)

Além de serem as unidades estruturais de moléculas de RNA e DNA, os nucleotídeos também funcionam como importantes co-fatores na sinalização e no metabolismo celular.

Nucleotídeos podem ser sintetizados com bases tanto purinas quanto pirimidinas por uma variedade de métodos *in vitro* e *in vivo*. No DNA as bases purinas são Adenina (A) e Guanina (G), enquanto as pirimidinas são Timina (T) e Citosina (C). Já o RNA usa Uracila (U) no lugar de Timina. Um nucleotídeo passa por vários estágios bioquímicos enquanto está sendo processado, adicionando e removendo átomos por meio do uso de diversas enzimas (ALBERTS et al., 2006).

A Figura 2.3 apresenta a estrutura do DNA. Para formar a estrutura de dupla hélice, o fosfato de um nucleotídeo é ligado ao açúcar de outro. As pontes de hidrogênio entre os fosfatos fazem o filamento do DNA se torcer. As bases nitrogenadas formam pares com bases complementares (purina com pirimidina).

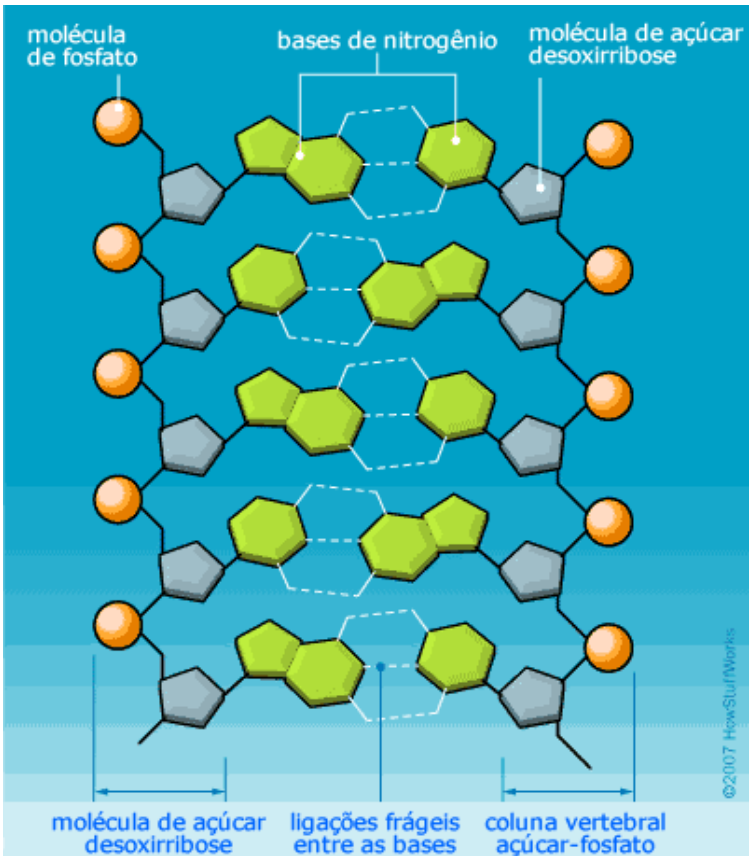


Figura 2.3 – Ligações entre Nucleotídeos para Formação do DNA

Fonte: Freudenrich (2008)

2.2.2. **Dogma Central da Biologia Molecular**

O produto inicial da expressão genômica é o transcriptoma, uma coleção de moléculas de RNA derivadas dos genes codificadores de proteínas cuja informação biológica é requisitada pela célula em um dado momento. Essas moléculas de RNA direcionam a síntese do produto final da expressão genômica: o proteoma, o repertório celular de proteínas, que especificam a natureza das reações bioquímicas que a célula é capaz de realizar (BROWN, 2002). A Figura 2.4 apresenta os produtos da expressão do genoma.

O transcriptoma é construído pelo processo chamado transcrição, no qual genes individuais são copiados para moléculas de RNA. A construção do proteoma envolve a tradução dessas moléculas de RNA em proteína (BROWN, 2002).

Figura 2.4 – Genoma, Transcriptoma e Proteoma

Após a determinação da estrutura do DNA no início da década de 1950, tornou-se claro o fato de que a informação genética nas células estava codificada na sua sequência de nucleotídeos. No entanto, mesmo antes da decodificação do DNA já se sabia que a informação genética era responsável pelo direcionamento da síntese das principais constituintes das células e determinantes tanto de sua estrutura quanto de seu funcionamento: as proteínas (ALBERTS et al., 2006).

DNA e proteínas, ambos polímeros com unidades repetidas, são macromoléculas com papel fundamental na vida celular. O primeiro armazena toda a informação genética como uma sequência de nucleotídeos e a transmite por meio de sua replicação. No entanto não é esse o responsável pela realização das funções vitais da célula, e sim as proteínas. Portanto é necessário que os quatro tipos de nucleotídeos sejam traduzidos para os vinte tipos de aminoácidos constituintes das proteínas. Essa etapa de tradução é crucial para a expressão genômica (KAMOUN et al., 2006).

Em cada gene do genoma a informação biológica é dividida em uma série de *exons* codificantes de proteínas separados por *introns* não codificantes. O RNA inicialmente sintetizado durante a expressão de um gene é uma cópia do gene completo, ou seja, inclui tanto *introns* quanto *exons*. No processo de *splicing* os *introns* são removidos desse pré-RNA mensageiro e os *exons* são unidos para formar o RNAm que, no fim, dirige a síntese protéica (BROWN, 2002).

Por meio de um mecanismo celular moléculas de DNA são transcritas para moléculas de RNA, que são então traduzidas para proteínas. Cada grupo de três nucleotídeos (um códon) é traduzido para um aminoácido no processo de tradução. Esses aminoácidos se unem por ligações peptídicas formando uma proteína. Considerando que existem vinte tipos de aminoácidos e 64 possíveis combinações dos quatro tipos de nucleotídeos três a três, observa-se que há aminoácidos codificados por mais de um códon. A Figura 2.5 mostra o Dogma Central da Biologia Molecular, formado pelos fluxos de

informação de DNA para DNA por meio do processo de replicação e de DNA para proteínas passando pelo RNA com os processos de transcrição e tradução (KANEHISA, 2000).

Figura 2.5 – Dogma Central da Biologia Molecular

Mecanismos de controle existem para regular cada um dos passos da expressão genômica, permitindo que a composição do transcriptoma e proteoma seja alterada de modo rápido e preciso e que a célula ajuste suas capacidades bioquímicas em resposta a variações no ambiente extracelular e a sinais recebidos de outras células (BROWN, 2002).

Parasitos da família do *T. cruzi* desenvolveram mecanismos de funcionamento próprios que lhes permitem escapar das defesas dos organismos hospedeiros e se reproduzir rapidamente. A principal diferença entre esses parasitos e os demais organismos com células nucleadas está no fato de que, no momento de se dividir e originar outra célula idêntica, esses protozoários seguem uma estratégia diferente. Na etapa inicial de síntese protéica, ao invés da decodificação de um gene por vez, são lidos todos os genes de uma única vez. Nesse momento, a longa molécula espiralada de DNA se espalha pela periferia do núcleo do parasito e somente após o término da cópia simultânea dos genes a mensagem de cada um é separada e a síntese das proteínas que formarão seus descendentes é iniciada (ZORZETTO, 2005). O *T. cruzi*, assim como outros tripanosomatídeos, regula a expressão de proteínas após a transcrição por meio de variações na estabilidade ou na eficiência da tradução dos RNAs mensageiros (SODRÉ et al., 2008).

2.2.3. **Transcriptoma**

Apesar de o transcriptoma constituir menos de 4% do RNA celular total, ele é o componente mais significativo porque contém os RNAs codificantes de proteínas, ou seja, que especificam a composição do proteoma e, portanto, determinam a capacidade bioquímica da célula. Um ponto importante a notar é que o transcriptoma nunca é sintetizado novamente. Toda célula recebe parte do transcriptoma de sua célula mãe

durante a divisão celular e mantém um transcriptoma durante sua vida. O processo de transcrição, desse modo, não resulta na síntese do transcriptoma, mas na manutenção dele por meio da substituição de RNAs mensageiros que tenham sido degradados e traz modificações à composição do transcriptoma ativando e desativando diferentes conjuntos de genes (BROWN, 2002).

Sequências de RNAm espelham a sequência de DNA dos genes dos quais foram transcritas. Consequentemente, por meio da análise do transcriptoma, pesquisadores podem determinar o momento e o local em que um gene é ativado ou desativado em vários tipos de células e tecidos. A depender da técnica utilizada é possível contar o número de transcrições para determinar a quantidade de atividade genética, ou nível de expressão, em certo tipo de célula ou tecido. Quanto maior o número de transcrições, geralmente mais importante aquela transcrição é para o funcionamento celular (NHGRI, 2008). Estudos com transcriptoma ajudam a explicar uma sequência de genoma, dando apoio à identificação dos genes cujos papéis no genoma não foram determinados por outros métodos (BROWN, 2002).

As variações de padrões de expressão gênica são a base da ampla variedade de diferenças físicas, bioquímicas e de desenvolvimento entre os vários tipos de células e tecidos. Por meio da coleta e comparação de transcriptomas de diferentes tipos celulares, pesquisadores podem adquirir conhecimento mais profundo sobre os constituintes de um tipo celular específico e sobre como alterações na atividade celular podem refletir ou contribuir para doenças. Além disso, é possível gerar, por meio do alinhamento do transcriptoma de cada tipo celular com o genoma, uma boa idéia de quais genes estão ativos em quais células (NHGRI, 2008).

Probst et al. (2007) realizou a análise com *microarray* do transcriptoma do *T. cruzi* em diferentes processos como diferenciação, ciclo de vida, resistência a drogas, ciclo celular, resposta a estresse e sobre-expressão protéica. Juntos, esses dados fornecem uma oportunidade de identificar conjuntos de RNAm co-regulados em situações distintas, permitindo a construção da primeira representação do reguloma do *T. cruzi*, com regulação genética envolvida em processos como replicação, transcrição e tradução, entre outros.

2.3. Proteínas Mosaicas

Proteínas mosaicas, de acordo com Avery et al. (1993), podem ser formadas por um ou mais tipos de módulos estruturais e possuem uma extensão de diferentes funções. Segundo Gaboriaud et al. (1998), muitas proteínas extracelulares são constituídas por um repertório limitado de padrões ou módulos de sequência, sendo chamadas de proteínas mosaicas e descritas como a justaposição linear de módulos e/ou domínios contíguos. A Figura 2.6 mostra a representação gráfica da estrutura de proteínas mosaicas.

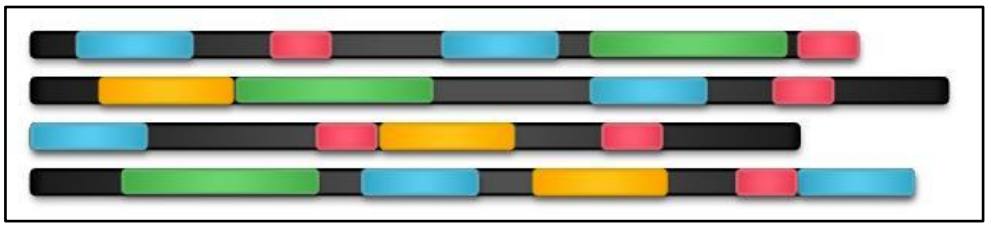


Figura 2.6 – Representação Gráfica da Estrutura de Proteínas Mosaicas

Módulos podem ser definidos como subconjuntos de domínios usados em diversas proteínas como "blocos de construção", e provavelmente apareceram devido à "mistura" de genes (HEGYI & BORK, 1997). Várias proteínas mosaicas desempenham um papel essencial na série de reações químicas da biologia extracelular (GABORIAUD, 1998).

Conforme Kolkman & Stemmer (2001), domínios discretos frequentemente envolvidos em funções específicas que contribuem para a atividade protéica geral compõem muitas proteínas. Uma forte correlação entre organização de domínio e estrutura *intron-exon* é revelada pela análise dos genes codificadores de proteínas mosaicas, ou seja, há uma tendência a domínios estarem codificados por um ou uma combinação de *exons* que iniciam e terminam no mesmo quadro de *splice*. Proteínas mosaicas aparentam ser criadas pela união de múltiplos domínios por meio do embaralhamento de *exons*.

Para Doolittle (1995) domínios encontrados em proteínas mosaicas se espalharam no decorrer da evolução e por isso ocorrem agora em proteínas que antes não estariam relacionadas. É proposto que proteínas mosaicas desempenharam papel importante na evolução da multicelularidade, visto que a maioria delas é extracelular ou constitui as partes extracelulares de proteínas ligadas à membrana (PATTHY, 1991).

2.4. Sequenciamento do Genoma do *T. cruzi*

A comunidade científica em torno de *T. cruzi*, *Leishmania major* (*L. major*) e *T. brucei*, com o advento dos projetos genoma de diversos organismos no cenário internacional no início dos anos 1990, passou a discutir a possibilidade de início dos projetos genoma desses parasitos (DEGRAVE, 2008). O conhecimento sobre a genética dos parasitos aumentou consideravelmente com os lançamentos desses projetos e também com outras iniciativas de sequenciamento em grande escala. O sequenciamento dos genomas de *T. cruzi, L. major* e *T. brucei* foi concluído em 2005, mas, mesmo antes de concluídos, esses projetos genoma já permitiram aos cientistas identificar diversos novos alvos terapêuticos potenciais, além de fornecer dados estruturais para estudos funcionais posteriores (GUIMARÃES, 2006).

Devido a dificuldades encontradas no Projeto Genoma do *T. cruzi* por características do genoma do parasito, o sequenciamento do genoma do *T. cruzi*, publicado na revista *Science* em 2005 juntamente com as sequências genômicas completas de *L. major* e *T. brucei*, foi apenas parcial. Foram preditas 22.570 proteínas, das quais 12.570 formam pares alélicos (DEGRAVE, 2008).

Em termos biológicos, o *T. cruzi* apresenta características bastante peculiares que se refletem na organização e função de seu genoma. Sua constituição genética demonstra a existência de grande polimorfismo, tendo consequentemente uma variação significativa na quantidade de DNA nuclear e no número de cromossomos entre diferentes isolados do parasito. Além disso, diferentemente da maioria dos organismos eucarióticos, os genes do *T. cruzi* e de outros tripanosomatídeos geralmente não são interrompidos por sequências de inserção (GOLDENBERG, 2008). Tais sequências são simplesmente sequências de DNA que se integram em diferentes pontos do genoma, provocando ou não modificação na função gênica.

As sequências repetitivas do DNA do *T. cruzi* representam pelo menos 50% de todo o seu genoma e são formadas principalmente pelas famílias de genes que compõem as proteínas de superfície. Esses totalizam 18% dos genes codificadores de proteínas do *T. cruzi*. A família MASP do *T. cruzi*, utilizada neste trabalho, é uma família de proteínas de superfície associadas a mucinas que contém 1.377 membros, o que corresponde a aproximadamente 6% do genoma diplóide do *T. cruzi*, e é caracterizada por regiões

centrais altamente variáveis e que frequentemente contêm sequências repetidas (EL-SAYED et al., 2005).

O baixo número de peptídeos detectados por abordagens proteômicas sugere que proteínas da família MASP podem conter extensivas modificações após o processo de tradução. Genes da família MASP podem ser expressos em estágios intermediários não representados nos dados do proteoma ou podem ser expressos de modo mutuamente exclusivo (EL-SAYED et al., 2005).

Ainda há um campo vasto a ser pesquisado em relação à regulação da expressão gênica em tripanosomatídeos. No entanto, com a determinação da sequência genômica dos três desses de maior relevância para a saúde humana (*T. cruzi*, *T. brucei* e *L. major*) e o uso de ferramentas de análise genômica e pós-genômica, além do avanço dos estudos voltados para epigenética, novos mecanismos devem ser evidenciados (GOLDENBERG, 2008).

3. TÉCNICAS E FERRAMENTAS

A busca por similaridade entre sequências de ácidos nucléicos (DNA e RNA) ou proteínas é a base da maioria das ferramentas computacionais utilizadas na Bioinformática. Ferramentas fundamentadas na busca de similaridade podem ser utilizadas, por exemplo, para inferir funções, visto que é provável que sequências similares possuam uma história evolutiva e funções em comum.

3.1. Alinhamento de Sequências por Pares

Alinhamento de sequências é a comparação de duas ou mais (alinhamento por pares e alinhamento múltiplo, respectivamente) sequências de ácidos nucléicos ou proteína buscando uma série de caracteres individuais ou padrões de caracteres que ocorrem na mesma ordem nas sequências (MOUNT, 2004).

Por meio do alinhamento de sequências o pesquisador pode determinar se sequências possuem similaridade suficiente para justificar uma inferência sobre homologia, que significa que as sequências apresentam um ancestral comum. Similaridade, um argumento forte para homologia, é uma medida da qualidade do alinhamento entre sequências com base em um dado critério, sendo simplesmente uma comparação das sequências com algum método, por exemplo, a contagem das posições idênticas entre duas sequências, não se referindo a nenhum processo histórico.

Para alinhar um par de sequências, essas são escritas em duas linhas e se busca fazer com que caracteres idênticos ou similares sejam posicionados na mesma coluna. Caracteres que não são idênticos podem ser colocados na mesma coluna, sendo considerado casamento sem êxito, ou em frente a um *gap* (lacuna) da outra sequência. Caracteres não-idênticos e *gaps* devem ser posicionados de modo que o maior número possível de caracteres idênticos ou similares estejam posicionados na mesma coluna. São chamadas de sequências similares aquelas sequências que podem ser alinhadas imediatamente dessa maneira (MOUNT, 2004).

O número de alinhamentos possíveis é exponencial ao tamanho das sequências, visto que a ocorrência de *gaps* é permitida. Assim, não é possível experimentar todos. Além disso, a presença de *gaps* também pode gerar alinhamentos sem sentido, sendo necessário diferenciar alinhamentos que ocorreram devido a homologia de alinhamentos que se espera ocorrer ao acaso.

Há duas formas de alinhamento por pares: global e local. O alinhamento global tenta alinhar toda a extensão das sequências, sendo apropriado para sequências bastante semelhantes e que possuem aproximadamente o mesmo tamanho. Já o alinhamento local alinha regiões de sequência com alta densidade de casamentos, gerando assim ilhas de casamentos ou sub-alinhamentos nas sequências alinhadas. Esse tipo de alinhamento é apropriado para sequências que se assemelham em apenas algumas partes, que possuem tamanhos diferentes ou que compartilham domínios ou regiões conservadas (MOUNT, 2004). A Figura 3.1 ilustra a diferença entre os dois tipos de alinhamento de sequências por pares.

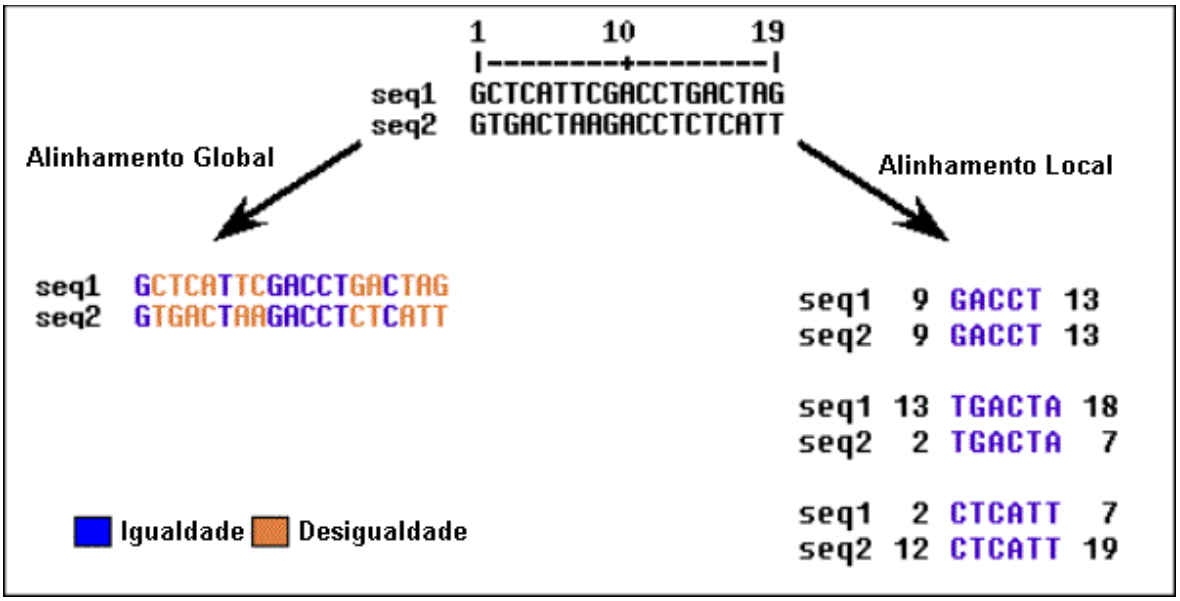


Figura 3.1 – Exemplo de Alinhamento Global (a) e Local (b)
Fonte: Prosdocimi et al. (2002)

O algoritmo Smith-Waterman é utilizado para produzir alinhamentos locais entre pares de sequências de aminoácidos ou nucleotídeos. Em um alinhamento local, o alinhamento é interrompido nas extremidades de regiões de alta similaridade e é dada uma prioridade muito maior à busca de tais regiões do que à extensão do alinhamento para incluir mais pares de aminoácidos ou nucleotídeos vizinhos. Esse tipo de alinhamento favorece a localização de padrões conservados de nucleotídeos em sequências de ácidos nucléicos ou de domínios de aminoácidos em sequências protéicas (MOUNT, 2004).

Segundo Prosdocimi et al. (2002) há diversas ferramentas computacionais que realizam alinhamentos de sequências, sendo que a grande maioria delas pode ser utilizada *online*, sem necessidade de instalação. As mais utilizadas são:

- BLAST, o programa de alinhamento mais utilizado no mundo, busca bancos de dados de ácidos nucléicos e proteínas por sequências homólogas e realiza alinhamento local por pares de sequências, e por isso foi escolhido como ferramenta deste trabalho;
- ClustalW, que é um programa de alinhamento múltiplo de sequências de DNA ou proteínas de propósito geral. Produz alinhamentos múltiplos biologicamente significativos de sequências divergentes. Calcula o melhor casamento para as sequências selecionadas, e as posiciona de forma que as identidades, similaridades e diferenças possam ser observadas.
- FASTA, que realiza alinhamento local buscando bancos de dados de ácidos nucléicos e proteínas. É o precursor dos programas de alinhamento. Pode ser muito específico identificando regiões extensas de baixa similaridade, especialmente para sequências altamente divergentes;
- MultAlin, que cria alinhamento múltiplo de sequências de um grupo de sequências relacionadas usando progressivos alinhamentos por pares.

3.2. BLAST: *Basic Local Alignment Search Tool*

O BLAST, uma das principais ferramentas da Bioinformática, utiliza um método heurístico baseado na determinação de trechos de similaridade local pela comparação de sequências de ácidos nucléicos ou proteínas contra sequências armazenadas em um banco de dados. Após as comparações é calculada a significância estatística para os resultados obtidos. O BLAST pode ser usado para auxílio à identificação de membros de famílias gênicas e para inferir relações funcionais evolutivas de várias sequências (KORF et al., 2003).

3.2.1. Os Programas do BLAST

O BLAST é basicamente um conjunto de programas que buscam similaridades estatisticamente significativas em bancos de dados de sequências. Os cinco programas tradicionais do BLAST são BLASTN, BLASTP, BLASTX, TBLASTN e TBLASTX. O primeiro trabalha com comparação de sequências de ácidos nucléicos, enquanto os demais

realizam comparação de sequências protéicas (KORF et al., 2003). Neste trabalho foi utilizado o programa BLASTN, descrito em detalhes a seguir. Os demais programas são descritos brevemente.

- BLASTN – Tem como entrada uma sequência de nucleotídeos e a compara com um banco de dados de sequências de nucleotídeos. O BLASTN é muito utilizado para procurar sequências que são muito conservadas, sendo tipicamente aplicado para classificação de elementos repetitivos, exploração de sequências entre espécies, explicação de DNA genômico e clusterização de estudos protéicos. Como a molécula de DNA tem fita dupla e genes podem ocorrer em ambas as fitas, quando uma sequência *query* é comparada com um banco de dados o BLASTN examina ambas as suas fitas: a sequência original com rótulo positivo e seu complemento reverso, com rótulo negativo. Como o BLAST alinha apenas caracteres e não tem nenhum modelo de genes ou outras características incluídas, é impossível determinar a partir de um alinhamento BLASTN em que fita o gene está (KORF et al., 2003).
- BLASTP – Compara uma sequência de aminoácidos com um banco de dados de sequências de aminoácidos.
- BLASTX – Compara uma sequência de nucleotídeos traduzida em proteína com um banco de dados de sequências de aminoácidos.
- TBLASTN – Compara uma sequência de aminoácidos contra um banco de dados de nucleotídeos traduzidos em proteína.
- TBLASTX – Compara uma sequência de nucleotídeos traduzidos em proteína contra um banco de dados de sequências de nucleotídeos traduzidos em proteína.

3.2.2. **Parâmetros do BLAST**

Uma série de parâmetros controla o algoritmo do BLAST (Seção 3.2.3), muitos dos quais possuem valores padrão e não precisam ser explicitamente determinados. Os parâmetros do BLASTN utilizados neste trabalho e tidos como mais relevantes são detalhados abaixo. Uma lista completa dos parâmetros pode ser encontrada no Anexo.

- **Filtro de Regiões de Baixa Complexidade (-F)**

A filtragem elimina do relatório do BLAST informações que, apesar de estatisticamente significantes, são desinteressantes biologicamente, mantendo apenas regiões biologicamente mais interessantes da sequência de consulta (*query*) para serem comparadas com as sequências do banco de dados. O BLAST usa por padrão a filtragem DUST para o BLASTN e SEG para os outros programas (MAYER, 2008).

- ***E-value* (-e)**

É a medida da probabilidade de um alinhamento ocorrer ao acaso, indicando a validade do alinhamento: quanto menor, mais provável de representar uma similaridade real ao invés de ser um alinhamento aleatório (MAYER, 2008). Um alinhamento com *E-value* de 1e-63, por exemplo, indica que a probabilidade de ocorrer ao acaso um alinhamento tão bom quanto o primeiro ou ainda melhor é mínima (CLARK, 2008). O BLAST calcula o *E-value* para cada alinhamento, e o valor deste parâmetro representa o limite máximo de *E-value* para um alinhamento ser incluído no resultado final. São relatados por padrão alinhamentos com *E-value* máximo dez.

- **Tamanho Inicial de Palavra (-W)**

O Tamanho Inicial de Palavra é um dos parâmetros mais importantes que dirigem a sensibilidade de busca do BLAST, visto que define o tamanho de palavra a ser considerado no segundo passo do algoritmo do BLAST (Seção 3.2.3). Os valores padrão são três para sequências de proteínas e 11 para sequências de ácidos nucléicos.

- **Sistema de Pontuação para Nucleotídeos (-q e -r)**

Muitas buscas de nucleotídeos usam um sistema de pontuação simples que consiste em uma "recompensa" para uma igualdade de nucleotídeos e uma "penalização" para desigualdade. A razão recompensa/penalização absoluta deve ser aumentada à medida que a divergência de sequências aumenta. Uma razão de 0,33 (1/-3) é apropriada para sequências que são perto de 99% conservadas; uma razão de 0,5 (1/-2), para sequências 95% conservadas; e uma razão de um (1/-1), para sequências 75% conservadas (MAYER, 2008). O BLAST usa por padrão o sistema 1/-3.

3.2.3. O Algoritmo do BLAST

O algoritmo do BLAST busca primeiro por palavras ou *k*-tuplas comuns à sequência *query* e a sequências de um banco de dados, e por isso aumenta a velocidade do

alinhamento de sequências. Essa busca é delimitada às palavras mais significativas, sendo o tamanho inicial de palavra padrão três para proteínas e 11 para ácidos nucléicos. Esses tamanhos são o mínimo necessário para alcançar uma pontuação de palavra alta o suficiente para ser significativa, mas não tão alta que padrões significativos curtos sejam perdidos (MOUNT, 2004).

Segundo Mount (2004), os passos do algoritmo do BLAST para alinhar uma sequência *query* com um banco de dados de sequências são:

1) Opcionalmente é aplicado à sequência *query* o filtro de regiões de baixa complexidade para remover regiões que podem não ser úteis para a produção de alinhamentos significativos;
2) Uma lista de palavras de tamanho *W* (11 por padrão para ácidos nucléicos) na sequência *query* é montada começando com as posições 1 a *W*; então 2 a *W*+1 e assim por diante até que as últimas posições disponíveis na sequência sejam alcançadas;
3) As palavras da sequência *query* e seus complementos reversos (no caso de sequências nucleotídicas) são avaliados buscando um casamento exato com uma palavra em qualquer sequência do banco de dados. Para palavras de ácidos nucléicos, é usado um sistema de pontuação padrão de +1 para igualdade de nucleotídeos e -3 para desigualdade. Para buscas envolvendo sequências protéicas, as palavras *query* também são analisadas em busca de casamentos com qualquer outra combinação de aminoácidos, com o objetivo de se criar uma lista de possíveis casamentos para cada palavra *query*. A Figura 3.2 traz uma representação gráfica dos passos um a três;

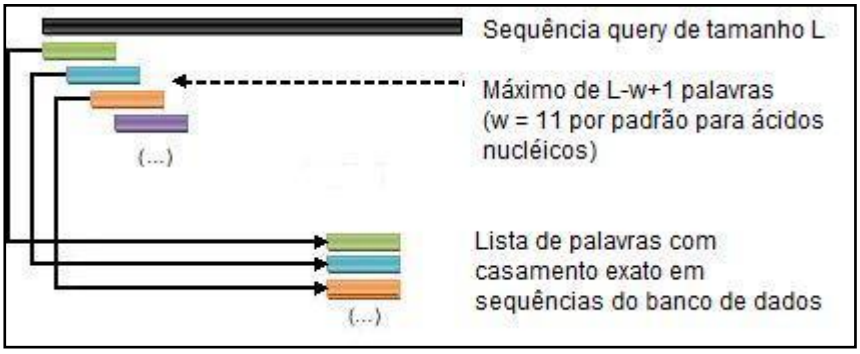


Figura 3.2 – Primeira Etapa do Algoritmo do BLAST (Passos Um a Três)

4) As palavras da sequência *query* e seus complementos reversos (no caso de sequências nucleotídicas) são avaliados buscando um casamento exato com uma palavra em qualquer sequência do banco de dados. Para palavras de ácidos nucléicos, é usado um sistema de pontuação padrão de +1 para igualdade de nucleotídeos e -3 para desigualdade. Para buscas envolvendo sequências protéicas, as palavras *query* também são analisadas em busca de casamentos com qualquer outra combinação de aminoácidos, com o objetivo de se criar uma lista de possíveis casamentos para cada palavra *query*. A Figura 3.2 traz uma representação gráfica dos passos um a três;
5) Para buscas com sequências protéicas, uma pontuação de corte chamada limiar de pontuação de palavra da vizinhança (*T*) é selecionada para reduzir o número de possíveis casamentos com a palavra *query* para apenas os mais significativos. Desse modo, apenas as palavras com pontuação maior ou igual a *T* são mantidas. A lista de possíveis casamentos da palavra *query* é desse modo reduzida de todas as possibilidades de casamento a apenas as de maior pontuação;
6) Para sequências de nucleotídeos, as palavras definidas nos passos dois e três são organizadas como tabelas indexadas ou dicionários da *query* e do banco de dados. O programa pode então encontrar rapidamente casamentos exatos iniciais com palavras *query* simplesmente procurando uma dada palavra no dicionário do banco de dados (INCOGEN, 2008);
7) Cada sequência do banco de dados é consultada em busca de um casamento exato com as palavras *query* definidas no passo três. Se um casamento é encontrado, ele é usado como semente de um possível alinhamento sem *gaps* entre a sequência *query* e as sequências do banco de dados. A Figura 3.3 mostra a representação gráfica deste passo;

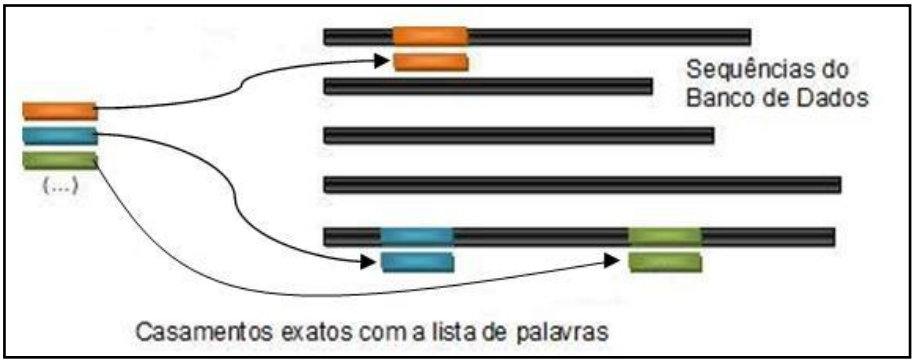


Figura 3.3 – Segunda Etapa do Algoritmo do BLAST (Passo Seis)

8) É realizada uma tentativa de extensão de um alinhamento em cada direção ao longo das sequências a partir de palavras casadas, continuando enquanto a pontuação aumentar, como mostrado na Figura 3.4. A extensão é interrompida quando a pontuação acumulada começa a cair. Nesse ponto, um trecho maior de sequência chamada *high-scoring segment pair* (HSP) ou par de segmentos de alta pontuação, que possui uma pontuação maior que a palavra original, pode ter sido encontrada;

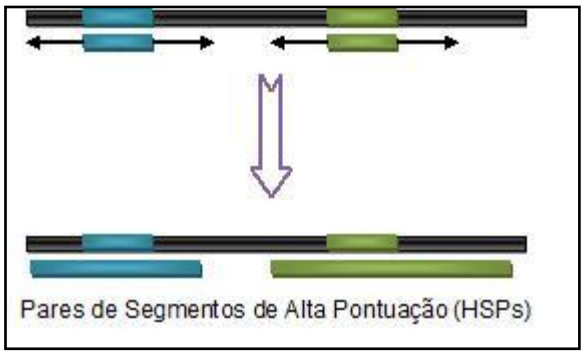


Figura 3.4 – Terceira Etapa do Algoritmo do BLAST (Passo Sete)

9) É determinado se cada pontuação de HSP encontrada tem valor maior que uma pontuação de corte *S*. Um valor apropriado para *S* é determinado empiricamente examinando a faixa de pontuações encontradas pela comparação de sequências aleatórias e pela escolha do valor significativamente maior. Os HSPs casados em todo o banco de dados são identificados e listados;
10) É determinada a significância estatística para cada pontuação de HSP e calculada a probabilidade de duas sequências aleatórias atingirem a pontuação de HSP;

11) São mostrados alinhamentos locais Smith-Waterman da sequência *query* com cada sequência casada no banco de dados. Versões iniciais do BLAST produziam apenas alinhamentos sem *gaps* que incluíam o HSP inicialmente encontrado. Se dois HSPs eram encontrados, dois alinhamentos separados eram produzidos porque as duas regiões não podiam ser alinhadas sem *gaps*. O BLAST2 produz um único alinhamento com *gaps* que inclui todas as regiões HSP inicialmente encontradas. A pontuação do alinhamento é obtida e o *E-value* (Seção 3.2.2) para aquela pontuação é calculado usando parâmetros estatísticos para alinhamentos com *gaps* que utilizam a mesma combinação de sistema de pontuação e penalidade de *gap* usada na busca de similaridade;
12) Quando o *E-value* para a pontuação do alinhamento local da sequência *query* com a sequência do banco de dados satisfaz o valor limite (que pode ser alterado pelo usuário), o casamento com a sequência do banco de dados é reportado. Os resultados da busca são mostrados como uma lista de casamentos ordenados pela pontuação do alinhamento e *E-value*, seguida pelos alinhamentos de sequências.

3.2.4. **Sistema de Pontuação para Sequências Nucleotídicas**

A extensão de alinhamentos semente necessita de um sistema de pontuação e de um procedimento para maximizar a pontuação localmente. O sistema de pontuação designa "recompensas" para igualdades de nucleotídeos e "penalidades" para desigualdades e para formação e extensão de *gaps*. Geralmente são utilizados os valores padrão, mas esses podem também ser fornecidos pelo usuário. A maximização local da pontuação (para conseguir o alinhamento final) acontece pela exploração de possíveis extensões, mantendo-se a par do alinhamento de maior pontuação encontrado e voltando a esse alinhamento quando a pontuação cai mais do que um determinado valor abaixo da pontuação mais alta corrente (CLARK, 2008).

A pontuação é a base da busca do BLAST. Para cada nucleotídeo em um alinhamento é dada uma pontuação dependendo se o seu casamento teve ou não êxito. Se o casamento não teve êxito (desigualdade), a pontuação é negativa; caso contrário (igualdade), é positiva. Para cada região de alinhamento todas as pontuações são somadas, embora a pontuação nunca possa ficar abaixo de zero (CLARK, 2008).

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Query	A	T	T	T	C	C	G	C	C	C	G	A	T	G	G	C	G	C	A	G	T	C	C	A	C
			T	T	C		G	C	C	C									A	G	T				
Sequência BD	C	G	T	T	C	A	G	C	C	C	A	T	A	A	A	A	C	T	A	G	T	T	G	T	G
Pontuação	−1	−1	1	1	1	−1	1	1	1	1	−1	−1	−1	−1	−1	−1	−1	−1	1	1	1	−1	−1	−1	−1
Pontuação Acumulada	0	0	1	2	3	2	3	4	5	6	5	4	3	2	1	0	0	0	1	2	3	2	1	0	0

Figura 3.5 – Exemplo de Pontuações de um Alinhamento

A Figura 3.5 apresenta um exemplo de pontuações de alinhamento. Nesse alinhamento simples, a primeira pontuação acumulada positiva está na posição três, que é então definida como o início da região de alta pontuação. A máxima pontuação acumulada ocorre na posição dez, que é então definida como o fim potencial da região de alta pontuação. O valor se tornaria negativo na posição 17, o que confirma o fim da região de alta pontuação. A próxima região positiva está na posição 19 e vai até a posição 21. Novamente, o resultado se tornaria negativo na posição 25, que também é o fim da sequência, logo 19-21 também é uma região de alta pontuação.

No exemplo foi usado um sistema simples +1/-1 para demonstrar o princípio de pontuação. O programa BLAST que executa uma busca nucleotídica (BLASTN) usa por padrão +1 para um casamento com êxito e -3 para um casamento sem êxito. Alinhamentos de ácidos nucléicos são simples porque há apenas alterações limitadas que podem ocorrer, podendo-se pontuar um casamento com êxito com um valor positivo e um sem êxito com um valor negativo (CLARK, 2008).

3.2.5. **Relatório do BLAST**

A saída produzida pelo BLAST é um relatório contendo informações de similaridade dos alinhamentos encontrados (SOUSA, 2007). Esse relatório consiste de três seções principais: cabeçalho contendo informação sobre a sequência *query* e o banco de dados buscado; descrições de cada sequência do banco de dados alinhada com a *query*; alinhamentos da *query* com cada sequência do banco de dados alinhada, podendo haver mais de um para a mesma sequência (KORF et al., 2003).

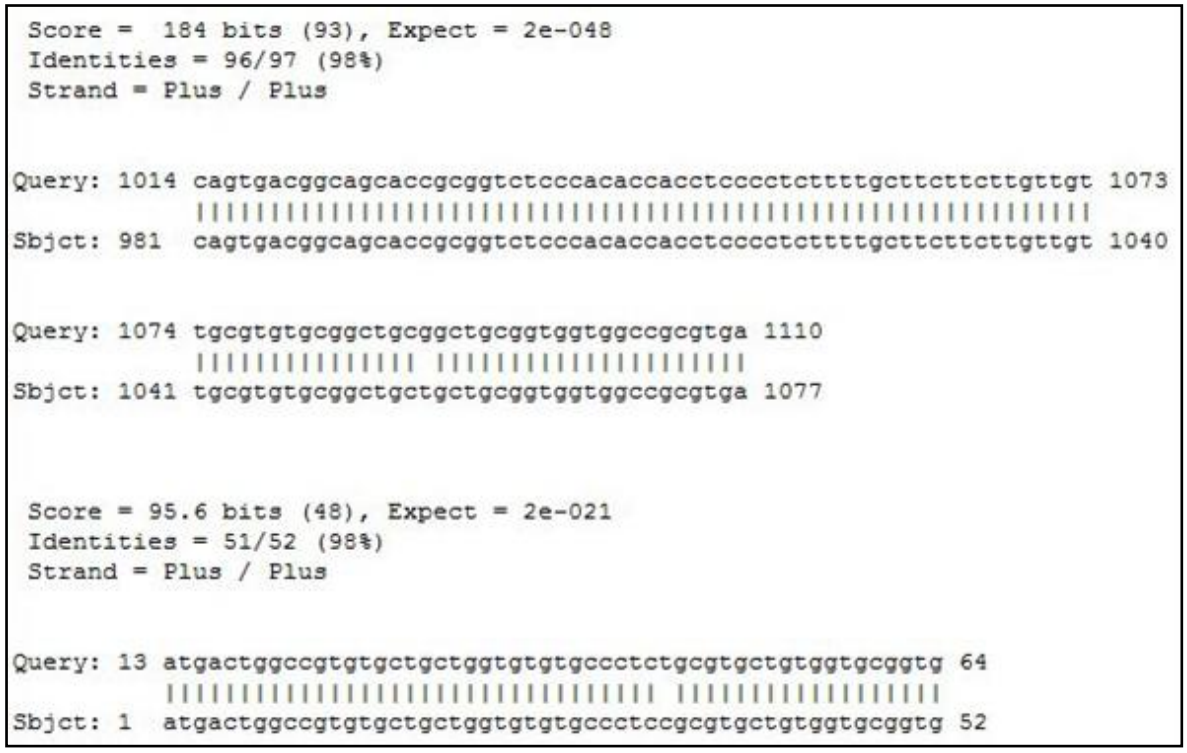


Figura 3.6 – Alinhamento Local de Par de Sequências em Relatório do BLAST

Os alinhamentos constituem a maior parte do relatório do BLAST. Na Figura 3.6 são apresentados dois alinhamentos locais entre uma *query* e uma sequência do banco de dados (*Sbjct*) gerados pelo programa BLASTN (Seção 3.2.1). Para cada alinhamento é apresentado um conjunto de valores que caracterizam similaridade: a pontuação (*Score*), o *E-value* (*Expect*) e o número e percentual de identidade entre nucleotídeos (*Identities)*. Além disso, são apresentadas as posições da região alinhada em cada sequência.

3.2.6. **Busca por Casamentos Curtos**

Sequências curtas (menos de 20 nucleotídeos) frequentemente não encontrarão casamentos significativos com as entradas do banco de dados sob as configurações padrão do BLAST. As razões gerais para isso são que o limiar de significância definido pelo *E-value* é estabelecido muito rigorosamente e o tamanho de palavra padrão é definido muito alto. Esses parâmetros devem ser ajustados para trabalhar com sequências curtas. O filtro de baixa complexidade também é removido visto que elimina porcentagens maiores de uma sequência curta, podendo até mesmo eliminar a *query* (INCOGEN, 2008).

Quanto menor o *E-value*, mais significativo é o alinhamento. No entanto, buscas envolvendo sequências curtas podem ser praticamente idênticas e apresentar *E-value* alto visto que o cálculo do *E-value* leva em consideração o tamanho da sequência *query* e, além

disso, sequências curtas têm alta probabilidade de ocorrer no banco de dados puramente ao acaso. Por isso quando se trabalha com sequências curtas deve ser definido um valor muito alto de *E-value* tanto para sequências de ácidos nucléicos quanto para sequências protéicas (MAYER, 2008).

A Tabela 3.1 apresenta um conjunto de parâmetros sugerido por NCBI (2008a) para buscas com sequências de nucleotídeos curtas.

Tabela 3.1 – Parâmetros do BLAST para Sequências Nucleotídicas Curtas

Parâmetro	Valor Padrão	Valor Indicado
Tamanho Inicial de Palavra	11	7
E-value	10	1000
Filtro de Baixa Complexidade	Ativado (T)	Desativado (F)

4. METODOLOGIA

Neste capítulo são apresentados o tipo de pesquisa em que se enquadra este trabalho, os dados tratados e os procedimentos metodológicos utilizados ao longo do trabalho.

4.1. Tipo de Pesquisa

Quanto à natureza, segundo Jung (2004) esta pesquisa é aplicada, visto que busca gerar novos conhecimentos por meio da utilização de conhecimentos e experiências adquiridos por estudiosos e profissionais da Bioinformática e aplica técnicas já existentes na literatura.

Como se estuda um assunto atual ainda pouco examinado entre as comunidades e se tem em vista a descoberta de teorias e práticas que modificarão as existentes, em relação ao objetivo esta pesquisa é exploratória (JUNG, 2004 apud ZAMBALDE, 2008).

Quanto aos procedimentos, segundo Jung (2004) esta pesquisa é experimental, visto que requere manipulação imparcial de dados, viabiliza a descoberta de novos métodos e técnicas e é utilizada para obtenção de novos conhecimentos.

4.2. Obtenção dos Dados

Neste trabalho foram utilizadas 810 sequências de nucleotídeos codificadoras das proteínas da família MASP do *T. cruzi* e constituintes do transcriptoma do parasito, organizadas no formato FASTA, que é o formato de entrada padrão para o BLAST. Os dados foram obtidos junto ao Departamento de Parasitologia do Instituto de Ciências Biológicas da Universidade Federal de Minas Gerais, que tem o *T. cruzi* como uma de suas linhas de pesquisa na subárea de Protozologia.

4.3. Procedimentos Metodológicos

Como ainda não foi proposta na literatura uma metodologia para identificação de módulos formadores de proteínas mosaicas, neste trabalho foram empregados procedimentos para criação de uma estratégia que possibilitasse o desenvolvimento da metodologia proposta de identificação de módulos formadores de proteínas mosaicas do *T. cruzi* a partir do transcriptoma do parasito. Este trabalho foi realizado em paralelo com o

trabalho de Gomes & Souza (2008), que propõe uma metodologia de identificação de módulos de proteínas mosaicas do *T. cruzi* a partir do proteoma do parasito.

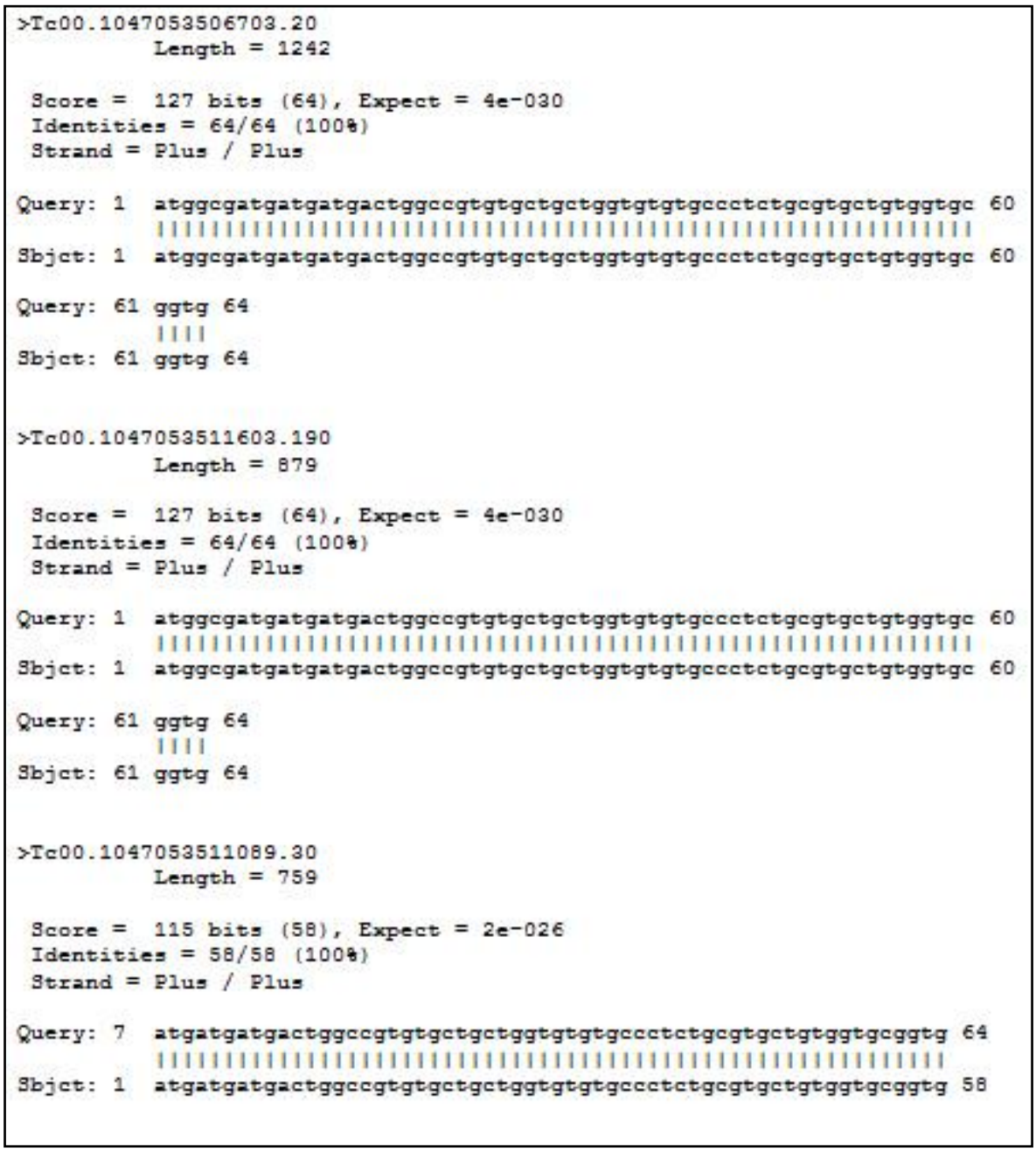
Neste trabalho foram utilizados os programas BLASTN e *formatdb* da versão 2.2.17 da ferramenta BLAST para realização de alinhamentos entre pares de sequências nucleotídicas e criação de bancos de dados a partir de arquivos contendo sequências, respectivamente.

O passo inicial adotado para o desenvolvimento da metodologia de identificação de módulos foi a realização do alinhamento das sequências nucleotídicas codificadoras das proteínas da família MASP todas contra todas e da análise dos resultados obtidos. Assim, o BLAST foi executado com valores padrão dos parâmetros para o conjunto S_1 das 810 sequências em estudo gerando o alinhamento A_1. Os resultados foram filtrados segundo a estratégia detalhada na Seção 4.3.1.2, eliminando desse modo os alinhamentos não considerados significativos, visto que, considerando a estrutura modular das proteínas, tais módulos estariam presentes nos alinhamentos filtrados.

O resultado do BLAST possibilitou a verificação visual da existência de regiões comuns a diversas sequências da família. A Figura 4.1 exemplifica isso, mostrando alinhamentos da mesma região de uma sequência *query* com três sequências distintas do banco de dados. Essas regiões foram extraídas para possibilitar a análise específica das mesmas, sendo utilizada para isso uma estratégia de corte detalhada na Seção 4.3.1.3. As regiões extraídas compõem o conjunto de sequências S_2 que são subsequências daquelas contidas em S_1.

Entre os alinhamentos filtrados de A_1 também foi possível observar a existência de transitividade, ou seja, considerando, por exemplo, regiões A, B e C de sequências, se a alinhou com B, que por sua vez alinhou com C, então A também alinhou com C. Como consequência, a ocorrência de redundância no conjunto S_2 é bastante evidente, sendo necessária a sua eliminação. Para isso as sequências de S_2 foram alinhadas com elas mesmas e foi realizada, posteriormente, a filtragem dos resultados. Foi criada então uma estratégia de separação dos resultados em grupos (Seção 4.3.1.4), de modo que sequências semelhantes, ou seja, que apresentam partes idênticas entre si, fossem inseridas no mesmo grupo. A separação dos grupos possibilitou a observação de que uma mesma sequência pode ser inserida em grupos diferentes, visto que pode conter mais de uma região comum às demais sequências. Para tratamento disso, a estratégia de corte foi aplicada aos grupos para se obter uma subsequência representativa de cada um, gerando o conjunto de

sequências S_3. Assim foi possível reduzir o número de sequências presentes em mais de um grupo.



```
>Tc00.1047053506703.20
          Length = 1242

 Score =  127 bits (64), Expect = 4e-030
 Identities = 64/64 (100%)
 Strand = Plus / Plus

Query: 1  atggcgatgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgc 60
          ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct: 1  atggcgatgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgc 60

Query: 61 ggtg 64
          ||||
Sbjct: 61 ggtg 64


>Tc00.1047053511603.190
          Length = 879

 Score =  127 bits (64), Expect = 4e-030
 Identities = 64/64 (100%)
 Strand = Plus / Plus

Query: 1  atggcgatgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgc 60
          ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct: 1  atggcgatgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgc 60

Query: 61 ggtg 64
          ||||
Sbjct: 61 ggtg 64


>Tc00.1047053511089.30
          Length = 759

 Score =  115 bits (58), Expect = 2e-026
 Identities = 58/58 (100%)
 Strand = Plus / Plus

Query: 7  atgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgcggtg 64
          ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct: 1  atgatgatgactggccgtgtgctgctggtgtgtgccctctgcgtgctgtggtgcggtg 58
```

Figura 4.1 – Alinhamento de uma Mesma Região da *Query* com Três Sequências Distintas do Banco de Dados

Foi notado que para se atingir os possíveis módulos da família protéica era necessária a adoção de uma abordagem iterativa dos passos descritos anteriormente, até que todos os grupos se tornassem unitários, ou seja, cada um contivesse apenas um

possível módulo “escolhido” ao longo do processo. A cada iteração é definido pela estratégia de corte um representante de cada grupo para ser colocado no novo conjunto de sequências gerado. O tamanho dos grupos é reduzido até que a sequência de um grupo só alinhe com ela mesma, sendo assim uma candidata a módulo que representa todas as demais sequências dos grupos dos quais ela participou ao longo do processo. Essa é a idéia central da metodologia desenvolvida neste trabalho.

Por último foi criada uma estratégia para definir quais módulos candidatos seriam considerados módulos da família de proteínas mosaicas (Seção 4.3.1.5).

4.3.1. **Estratégias Utilizadas**

Nesta seção são detalhadas as estratégias e rotinas criadas para a metodologia desenvolvida.

4.3.1.1. Formatação do Relatório do BLAST

Diversas informações sobre todos os alinhamentos significativos encontrados pelo BLAST são apresentadas em seu relatório, como descrito na Seção 3.2.5. As informações relevantes para a metodologia desenvolvida são apenas a identificação das sequências envolvidas no alinhamento, as posições alinhadas em cada uma delas e o valor de identidade do alinhamento. Esta rotina de formatação do relatório extrai do relatório essas informações e as organiza de modo a facilitar sua manipulação, como mostrado na Figura 4.2, em que cada linha representa um alinhamento e as colunas representam, respectivamente, as identificações da sequência *query* e da sequência do banco de dados, as posições iniciais e finais dos alinhamentos na *query* e na sequência do banco de dados e a porcentagem de identidade.

Tc00.1047053408547.10	Tc00.1047053506879.10	339	351	290	302	92
Tc00.1047053408547.10	Tc00.1047053508081.30	339	351	287	299	100
Tc00.1047053508869.59	Tc00.1047053510625.54	4	21	3	20	94
Tc00.1047053508869.59	Tc00.1047053508481.20	1	19	1	19	94
Tc00.1047053507071.349	Tc00.1047053510039.30	1	28	3	30	50

Figura 4.2 – Exemplo de Formatação do Relatório do BLAST

4.3.1.2. Filtragem dos Resultados

A estratégia de filtragem dos resultados elimina os alinhamentos que não obedecem a determinado patamar de identidade, conservando apenas os considerados significativos.

Foi definido que para sequências nucleotídicas apenas alinhamentos com 100% de identidade seriam considerados, visto que a variação de apenas um nucleotídeo leva à modificação do códon, que por sua vez pode codificar um aminoácido diferente, podendo assim modificar alguma característica da proteína codificada.

4.3.1.3. Estratégia de Corte

A estratégia de corte é aplicada tanto ao resultado do alinhamento das sequências originais quanto aos grupos criados durante as iterações. A idéia central dessa estratégia é, considerando os alinhamentos de uma dada região em uma dada sequência *query*, encontrar a subsequência da *query* que está presente na maioria dos alinhamentos ou mesmo em todos. Para isso são estudadas as posições da *query* alinhadas em cada um desses alinhamentos e são utilizadas como posições de corte aquelas que mais se repetem entre as posições iniciais e finais dos alinhamentos.

Uma vez definidas as posições de corte, essas são avaliadas para verificar se limitam uma subsequência com um dado tamanho mínimo. Para sequências nucleotídicas esse tamanho foi definido como 12 nucleotídeos, o que corresponde à codificação de quatro aminoácidos, para evitar a obtenção de módulos a partir de ocorrências aleatórias, o que pode ocorrer caso sejam considerados tamanhos menores. Se a subsequência definida pelas posições de corte obedecer a essa restrição de tamanho, é inserida no novo conjunto de sequências S_i sendo construído. Caso contrário, as posições de corte são descartadas e é buscado um novo par de posições que limite uma subsequência que satisfaça a restrição. Na ausência de tal par de posições, a região da *query* sendo trabalhada é descartada.

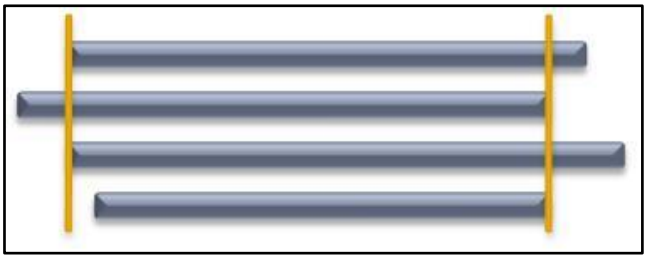


Figura 4.3 – Exemplo de Aplicação da Estratégia de Corte

A Figura 4.3 ilustra a aplicação da Estratégia de Corte. Considerando as linhas horizontais como alinhamentos de uma mesma região de uma sequência de consulta com diferentes sequências do banco de dados, as linhas verticais representam as posições de corte definidas por essa estratégia.

4.3.1.4. Separação de Grupos

Esta estratégia tem como objetivo o agrupamento de sequências semelhantes, ou seja, que apresentam partes idênticas entre si. Considerando as sequências A, B, C, D, E e F e os alinhamentos A-B, A-C, B-C, C-D, C-F e E-F, a separação em grupos é feita da seguinte forma: como inicialmente ainda não foram criados grupos, cria-se um novo grupo G_1 do qual A é cabeça; todas as sequências com as quais A se alinha são então inseridas no mesmo grupo. Para o exemplo, neste ponto $G_1 = \{A, B, C\}$.

Passa-se então aos alinhamentos com a *query* B. Inicialmente se busca todos os grupos aos quais B pertence (G_1 no caso do exemplo). Cada alinhamento de B é analisado a fim de se verificar se a sequência com que B se alinha pertence a algum grupo ao qual B pertence. Caso a sequência não esteja em nenhum grupo de B, ela é inserida no grupo em que B for cabeça; se tal grupo não existir é criado um novo grupo do qual B é cabeça e a sequência é inserida nesse novo grupo. Para o exemplo, verifica-se que C já pertence a G_1.

Continuando o processo se passa à análise dos alinhamentos da *query* C. É realizado o mesmo processo aplicado a B. Para o exemplo, ao analisar os alinhamentos de C é verificado que D não pertence a G_1. Como C ainda não é cabeça de grupo, é criado um novo grupo $G_2 = \{C, D\}$ e se passa ao próximo alinhamento (C-F). Como F não está em nenhum grupo a que C pertence mas C é cabeça do grupo G_2, F é inserido em G_2, que passa a ser $G_2 = \{C, D, F\}$.

A análise continua com os alinhamentos de E, visto que no exemplo não há alinhamentos cuja *query* é D. Neste ponto se atinge um novo caso: E não está em nenhum grupo, portanto é criado um novo grupo $G_3 = \{E, F\}$.

A Tabela 4.1 apresenta a configuração final dos grupos para o exemplo dado.

Tabela 4.1 – Exemplo de Separação em Grupos

G_1	G_2	G_3
A	C	E
B	D	F
C	F	

A estratégia de corte aplicada aos grupos define uma subsequência da cabeça de cada um como representante de todo o grupo. Para o exemplo dado, as posições de corte que definem a subsequência de A representante de G_1 são definidas com base nas posições dos alinhamentos A-B e A-C; a representante de G_2, com base nas posições de C-D e C-F; e a representante de G_3, com base nas posições do alinhamento E-F.

4.3.1.5. Definição de Módulos

Ao fim do processo iterativo são definidos os candidatos a módulos da família protéica. A estratégia de Definição de Módulos realiza o alinhamento dos possíveis módulos com as sequências originais e considera como módulos aqueles que alinham toda a sua extensão com 100% de identidade com pelo menos 1% das sequências da família. Essa porcentagem mínima de alinhamentos foi definida considerando que candidatos presentes em menos de 1% das sequências da família têm grande chance de ocorrer ao acaso.

4.3.2. Valores de Parâmetros do BLAST Utilizados

Os parâmetros do BLAST utilizados neste trabalho foram apresentados na Seção 3.2.2. Foram buscados na literatura valores de parâmetros mais indicados para se trabalhar com sequências curtas, o que foi descrito na Seção 3.2.6. Os valores utilizados para comparação de resultados da metodologia desenvolvida aplicada à família MASP do *T. cruzi* são apresentados na Tabela 4.2. A coluna "Valor NCBI" corresponde aos valores para sequências nucleotídicas curtas indicadas por NCBI (2008a). A coluna "Outros Valores" corresponde a valores testados diferentes dos valores padrão e dos indicados pela literatura. O sistema de pontuação de nucleotídeos é apresentado no formato (recompensa/penalidade).

Tabela 4.2 – Valores de Parâmetros Utilizados para Comparação de Resultados

Parâmetro	Valor Padrão	Valor NCBI (2008a)	Outros Valores
E-value	10	1000	500
Filtro de Regiões de Baixa Complexidade	Ativado	Desativado	---
Tamanho Inicial de Palavra	11	7	8-10, 12, 13
Sistema de Pontuação	(1/-3)	---	(1/-2), (1/-1)

O *E-value* 500 foi escolhido para teste por ser um valor intermediário entre os valores padrão e indicado pela literatura. Os sistemas de pontuação (1/-2) e (1/-1) foram testados por serem indicados, respectivamente, para sequências 95% e 75% conservadas segundo Mayer (2008).

4.3.3. **Metodologia Desenvolvida**

O processo descrito para chegar à metodologia de identificação de módulos formadores de proteínas mosaicas proposta utilizou valores padrão dos parâmetros do BLAST. A metodologia desenvolvida neste trabalho é apresentada em forma de algoritmo na Figura 4.4. A entrada do algoritmo é um conjunto (S_1) de sequências nucleotídicas codificadoras de proteínas de uma família do *T. cruzi*.

Esse algoritmo foi implementado em C++ e executado para a família MASP do *T. cruzi* inicialmente com os valores padrão dos parâmetros do BLAST. Posteriormente se passou a estudar outros valores, incluindo aqueles sugeridos na literatura, comparando-se os resultados obtidos para cada combinação de parâmetros utilizada.

```
Início do Algoritmo
      Fazer i = 1;
      Fazer u = falso;
      Enquanto u for igual a falso, fazer:
            Executar o BLASTN para obter os alinhamentos de Si com Si;
            Filtrar os alinhamentos utilizando a estratégia de filtragem;
            Se i > 1
                  Separar os alinhamentos filtrados em grupos;
                  Se todos os grupos forem unitários fazer:
                        u = verdadeiro;
                        Interromper o loop;
                  Senão
                        Aplicar a estratégia de corte nos grupos gerando
                        o conjunto de sequências Si+1;
            Senão
                  Aplicar a estratégia de corte aos alinhamentos
                  filtrados gerando o conjunto Si+1;
            Fazer i = i + 1;
       Fim do Enquanto;
      Executar o BLASTN para obter os alinhamentos de Si contra S1;
      Definir os módulos utilizando a estratégia de definição de
módulos;
Fim do Algoritmo.
```

Figura 4.4 – Algoritmo Representativo da Metodologia Desenvolvida

5. RESULTADOS E DISCUSSÃO

O algoritmo desenvolvido neste trabalho, descrito na Seção 4.3.3, foi implementado em C++ e executado para as sequências nucleotídicas codificadoras das proteínas da família MASP do *T. cruzi* com diferentes conjuntos de valores para os parâmetros do BLAST. Para uma melhor avaliação dos resultados obtidos, o alinhamento dos possíveis módulos com as sequências originais foi utilizado para mapear, para cada módulo, as posições em que ocorre nas sequências da família e para calcular sua frequência de ocorrência.

Posteriormente o conjunto de sequências originais foi alinhado com o conjunto de módulos definidos, sendo mapeadas, para cada sequência, as posições em que os módulos que ela apresenta ocorrem e calculada a frequência de ocorrência. A Tabela 5.1 apresenta os resultados encontrados para cada combinação de parâmetros testada, onde os códigos dos parâmetros são *-F* para filtro de regiões de baixa complexidade, *-e* para *E-value*, *-W* para tamanho de palavra, *-r* para recompensa de igualdade de nucleotídeos e *-q* para penalidade de desigualdade de nucleotídeos.

Os conjuntos de valores de parâmetros testados foram:

- **C_1**: -F T, -e 10, -W 11, -r 1, -q -3;
- **C_2**: -F F, -e 10, -W 11, -r 1, -q -3;
- **C_3**: -F F, -e 100, -W 11, -r 1, -q -3;
- **C_4**: -F F, -e 500, -W 11, -r 1, -q -3;
- **C_5**: -F F, -e 1000, -W 11, -r 1, -q -3;
- **C_6**: -F F, -e 500, -W 10, -r 1, -q -3;
- **C_7**: -F F, -e 500, -W 9, -r 1, -q -3;
- **C_8**: -F F, -e 500, -W 8, -r 1, -q -3;
- **C_9**: -F F, -e 500, -W 12, -r 1, -q -3;
- **C_{10}**: -F F, -e 500, -W 13, -r 1, -q -3;
- **C_{11}**: -F F, -e 500, -W 12, -r 1, -q -2;
- **C_{12}**: -F F, -e 500, -W 12, -r 1, -q -1.

O valor sete indicado por NCBI (2008a) para tamanho inicial de palavra não consta entre os resultados apresentados, pois houve erro de execução do BLAST para todas as combinações de valores de parâmetro testadas envolvendo esse valor de tamanho de palavra. Isso se deve possivelmente à ausência de memória necessária à execução correta, devendo-se ressaltar que a configuração da máquina utilizada contava com processador Core 2 Duo de 2GHz e 4GB de memória RAM. Na impossibilidade de execução do algoritmo com o valor indicado na literatura, foi testado o tamanho inicial de palavra com valores oito e nove. Os resultados com combinações desses valores se mostraram insatisfatórios visto que poucos módulos foram encontrados e o mapeamento desses resultou em várias sequências desprovidas de módulos.

Tabela 5.1 – Comparativo de Resultados de Combinações de Valores de Parâmetros do BLAST

Conjunto de Valores de Parâmetros	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência	Máximo de Módulos por Sequência
C_1	496	29	14	39
C_2	527	28	16	60
C_3	1300	17	24	97
C_4	1045	24	28	69
C_5	1043	24	28	67
C_6	530	21	12	39
C_7	169	25	4	20
C_8	24	32	0	7
C_9	**2464**	**16**	**42**	**183**
C_{10}	1235	17	23	147
C_{11}	2364	17	42	168
C_{12}	1368	21	30	136

Para comparação de resultados foi atribuído maior peso ao número de módulos encontrados e às frequências de ocorrência (valores médios de ocorrência de módulos e de ocorrência de módulos por sequência), considerando melhores os maiores valores. Um resultado é considerado melhor que aqueles aos quais está sendo comparado quando apresenta pelo menos dois desses três valores maiores que os dos demais.

O primeiro dos quatro parâmetros testados (Seção 3.2.3) foi o Filtro de Regiões de Baixa Complexidade, tendo cada um dos demais parâmetros mantido seu valor padrão. A comparação de C_1 (valores padrão) com C_2, apresentada na Tabela 5.2, mostrou que a desativação do filtro de regiões de baixa complexidade levou a melhores resultados. Dessa forma todas as outras combinações de valores de parâmetros testados utilizam o filtro desativado.

Tabela 5.2 – Resultados da Ativação/Desativação do Filtro de Baixa Complexidade

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_1 – Ativado	496	**29**	14
C_2 – Desativado	**527**	28	**16**

O próximo parâmetro testado foi o *E-value.* A Tabela 5.3 mostra os resultados do valor padrão (dez) com um valor maior (cem). Os resultados da comparação de C_2 e C_3 indicam que a elevação do valor desse parâmetro leva à melhoria dos resultados.

Tabela 5.3 – Resultados do Aumento do *E-value*

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_2 – 10	527	**28**	16
C_3 – 100	**1300**	17	**24**

Como o aumento do *E-value* em relação ao valor padrão provoca melhoria dos resultados, testou-se valores ainda maiores desse parâmetro com o objetivo de encontrar o que produz os melhores resultados. Na Tabela 5.4, que apresenta resultados de C_3, C_4 e C_5. Observa-se que os valores quinhentos e mil são melhores em relação ao valor cem e são equivalentes entre si. No entanto, C_4 é ligeiramente melhor dado o número total de módulos encontrados, portanto as demais combinações de valores de parâmetros consideram o *E-value* de quinhentos.

Tabela 5.4 – Resultados do Aumento Adicional do *E-value*

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_3 – 100	**1300**	17	24
C_4 – 500	1045	**24**	**28**
C_5 – 1000	1043	**24**	**28**

Em seguida foi testada a redução do valor do parâmetro Tamanho Inicial de Palavra (Tabela 5.5). A comparação de C_4 com C_6, C_7 e C_8 mostra que a redução do valor desse parâmetro levou à produção de resultados piores que o uso do valor padrão.

Tabela 5.5 – Resultados da Diminuição do Tamanho Inicial de Palavra

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_4 – 11	**1045**	24	**28**
C_6 – 10	530	21	12
C_7 – 9	169	25	4
C_8 - 8	24	**32**	0

Foi testado então o aumento do Tamanho Inicial de Palavra, sendo os resultados apresentados na Tabela 5.6. Observou-se a melhoria dos resultados com C_9, mas a comparação com C_{10} mostra que a elevação de mais uma unidade no valor desse parâmetro provoca piora dos resultados. Sendo assim, as demais combinações de valores de parâmetro testados utilizam o Tamanho Inicial de Palavra 12.

Tabela 5.6 – Resultados do Aumento do Tamanho Inicial de Palavra

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_4 – 11	1045	**24**	28
C_9 – 12	**2464**	16	**42**
C_{10} – 13	1235	17	23

O último dos quatro parâmetros testados foi o Sistema de Pontuação de Nucleotídeos. Os resultados são apresentados na Tabela 5.7, onde se observa que o Sistema de Pontuação com melhores resultados é o padrão (1/-3), que é mais apropriado para sequências 99% conservadas, como mostrado na Seção 4.3.2.

Tabela 5.7 – Resultados de Diferentes Sistemas de Pontuação de Nucleotídeos

Valor	Total de Módulos	Média de Ocorrências	Média de Módulos por Sequência
C_9 – 1/-3	**2464**	16	**42**
C_{11} – 1/-2	2364	17	**42**
C_{12} – 1/-1	1368	**21**	30

A análise dos resultados dos conjuntos de valores de parâmetros testados permitiu verificar que a configuração que apresentou melhores resultados na execução da metodologia proposta para a família de proteínas MASP do *T.* cruzi foi a combinação C_9. Essa configuração associa Sistema de Pontuação padrão para igualdade e desigualdade de nucleotídeos, alto valor de *E-value*, desativação do Filtro de Regiões de Baixa Complexidade e ligeira elevação em relação ao valor padrão do Tamanho Inicial de Palavra.

O processamento da família MASP com o conjunto de valores C_9 levou quatro iterações do algoritmo, distribuídas ao longo dos 192 minutos de execução utilizando uma máquina de 4GB de RAM com processador Core 2 Duo de 2GHz. Aplicada a estratégia de

definição de módulos, foram definidos 2464 módulos, sendo que cada um ocorreu em média 16 vezes ao longo das sequências da família. A Tabela 5.8 apresenta os cinco módulos de maior incidência na família MASP.

O Apêndice mostra o mapeamento de módulos para a sequência de maior incidência desses (183 módulos). A Figura 5.1 apresenta uma visualização gráfica desse mapeamento, com os módulos em destaque, onde é possível observar a ocorrência de sobreposição de módulos, o que acontece devido ao próprio BLAST relatar alinhamentos sobrepostos por alinhar uma mesma região da *query* mais de uma vez com a mesma sequência do banco de dados. A sobreposição sugere que os módulos obtidos ao fim do algoritmo não constituem necessariamente módulos individuais, havendo a possibilidade de serem combinados para formar outros módulos em um processo de refinamento dos resultados.

Tabela 5.8 – Módulos de Maior Incidência nas Proteínas da Família MASP

Módulo	Número de Ocorrências
GTGGTGGCCGCGTGA	646
CGTGTGCTGCTGGTGTGTGCCCTCTGCGTG	616
CCCCTCTTTTGC	602
GCGATGATGATG	478
GGCGACAGTGAC	363

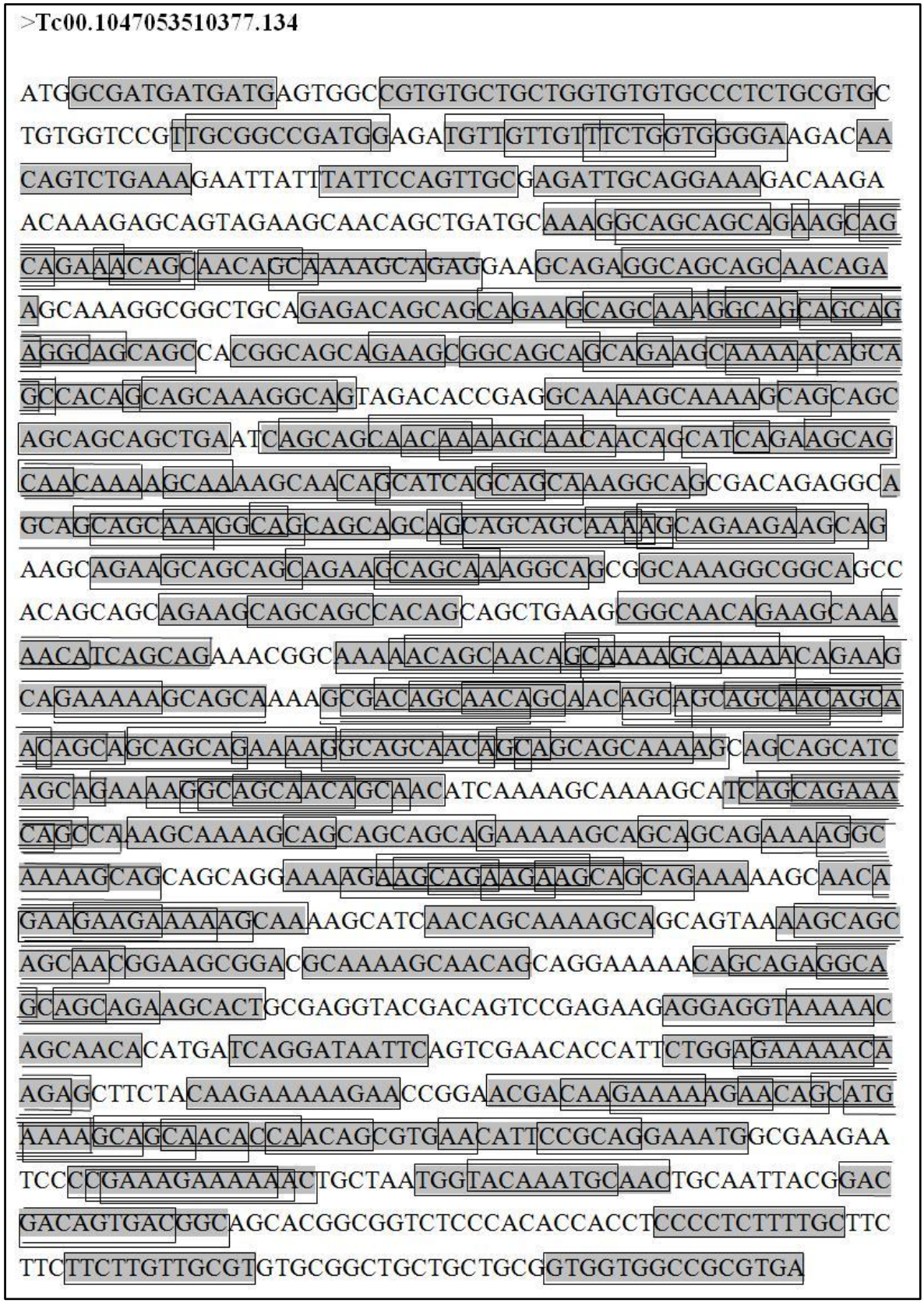


Figura 5.1 – Mapeamento de Módulos na Sequência Tc00.1047053510377.134

6. CONCLUSÃO

Este trabalho se propôs a criar uma metodologia para identificação de módulos formadores de sequências nucleotídicas codificadoras de proteínas mosaicas do *Trypanosoma cruzi* e constituintes do transcriptoma do parasito utilizando a ferramenta BLAST.

O algoritmo para a metodologia foi implementado e executado com diferentes combinações de parâmetros do BLAST a fim de comparação dos resultados obtidos. Como medidas de comparação, foram utilizados o número total de módulos encontrados e os valores médios de ocorrência de módulos e de ocorrência de módulos por sequência. Pela observação dos resultados se concluiu que a metodologia provou ser eficaz para identificação de módulos formadores de proteínas mosaicas a partir das sequências nucleotídicas que as codificam e que a combinação de desativação do filtro de regiões de baixa complexidade, alto valor de *E-value*, ligeira elevação do tamanho inicial de palavra em relação ao valor padrão e sistema de pontuação de nucleotídeos padrão apresentou os melhores resultados para a família de proteínas MASP do *T. cruzi*.

A partir dos resultados obtidos se concluiu também que foi confirmada a estrutura mosaica das proteínas da família MASP, visto que o mapeamento dos módulos encontrados possibilitou a visualização desses em todas as sequências da família com uma média de 42 módulos por sequência.

É proposto como trabalho futuro a comparação dos resultados obtidos neste trabalho com os encontrados por Gomes & Souza (2008), cuja identificação de módulos de proteínas mosaicas do *T. cruzi* se baseia no proteoma do parasito.

Como foi observada a sobreposição e ocorrência em série de alguns módulos, é proposto como trabalho futuro ainda o estudo da ocorrência condicional de módulos, ou seja, da possibilidade de ocorrência de um módulo estar condicionada à ocorrência de outro, o que possibilitaria o refinamento dos resultados obtidos neste trabalho por meio da redefinição como módulo único de módulos que se sobrepõem ou que ocorrem sempre em série. Além disso, o estudo da ocorrência condicional de módulos e da presença de um mesmo conjunto de módulos em diferentes sequências pode trazer informações importantes para estudiosos do *T. cruzi* e da Doença de Chagas.

APÊNDICE – Mapeamento de Sequência

Este apêndice apresenta o mapeamento dos módulos encontrados na família MASP do *T. cruzi* e das posições em que ocorrem na sequência que apresenta maior número de módulos (183).

Tc00.1047053510377.134

GCGATGATGATG	4	15
CGTGTGCTGCTGGTGTGTGCCCTCTGCGTG	22	51
TTGCGGCCGATG	62	73
TGCGGCCGATGG	63	74
GATGTTGTTGTT	76	87
TGTTGTTGTTTCTG	78	91
GTTGTTTCTGGTG	82	94
TTCTGGTGGGGA	87	98
AACAGTCTGAAA	103	114
TATTCCAGTTGC	123	134
AGATTGCAGGAAA	136	148
AAAGGCAGCAGCA	186	198b
GGCAGCAGCAGA	189	200
GCAGCAGCAGAAGCAGCA	190	207
AAGCAGCAGAAA	200	211
CAGCAGAAACAG	203	214
AGCAGAAACAGC	204	215
AACAGCAACAGC	210	221
ACAGCAACAGCA	211	222
AACAGCAAAAGCA	216	228
GCAGAGGCAGCAGC	235	248
GGCAGCAGCAAC	240	251
GCAGCAACAGAA	244	255
GAGACAGCAGCA	271	282
CAGAAGCAGCAA	281	292
GCAGCAAAGGCA	286	297
CAGCAAAGGCAG	287	298
AAAGGCAGCAGCA	291	303
GGCAGCAGCAGA	294	305
GCAGCAGCAGAGGC	295	308
CAGCAGAGGCAG	299	310
GCAGAGGCAGCAGC	301	314
CGGCAGCAGAAG	317	328
GAAGCGGCAGCAG	325	337
GGCAGCAGCAGA	330	341
GCAGAAGCAAAA	337	348
GAAGCAAAAACA	340	351
CAAAAACAGCAG	344	355
AAAAACAGCAGC	345	356

CAGCAGCCACAG	350	361	
GCAGCAAAGGCA	361	372	
CAGCAAAGGCAG	362	373	
GCAAAAGCAAAA	385	396	
AAGCAAAAGCAG	389	400	
CAGCAGCAGCAG	398	409	
CAGCAGCAGCAG	401	412	
AGCAGCAGCTGA	405	416	
CAGCAGCAACAA	419	430	
AGCAGCAACAAA	420	431	
CAACAAAAGCAA	425	436	
ACAAAAGCAACA	427	438	
AAAAGCAACAACA	429	441	
AACAACAGCATCA	435	447	
CATCAGAAGCAG	443	454	
CAGAAGCAGCAA	446	457	
AGCAGCAACAAA	450	461	
CAACAAAAGCAA	455	466	
GCAAAAGCAACAG	463	475	
CAGCATCAGCAG	473	484	
GCATCAGCAGCA	475	486	
GCAGCAAAGGCA	481	492	
CAGCAAAGGCAG	482	493	
AGCAGCAGCAAA	504	515	
GCAGCAAAGGCA	508	519	
CAGCAAAGGCAG	509	520	
AAAGGCAGCAGCA	513	525	
CAGCAGCAGCAG	518	529	
CAGCAGCAGCAG	521	532	
CAGCAGCAGCAG	524	535	
AGCAGCAGCAAA	528	539	
GCAGCAGCAAAA	529	540	
CAGCAGCAAAAG	530	541	
AAAAGCAGAAGA	537	548	
AAGCAGAAGAAG	539	550	
AGCAGAAGAAGCA	540	552	
CAGAAGAAGCAG	542	553	
AGAAGCAGCAGC	558	569	
GCAGCAGCAGAAGCAGCA	562	579	
CAGAAGCAGCAA	569	580	
GCAGCAAAGGCA	574	585	
CAGCAAAGGCAG	575	586	
GCAAAGGCGGCA	589	600	
AGAAGCAGCAGC	612	623	
CAGCAGCCACAG	617	628	
CGGCAACAGAAG	638	649	
GAAGCAAAAACA	646	657	
AAACATCAGCAG	653	664	
AAAAACAGCAACA	672	684	

AACAGCAACAGC	675	686
ACAGCAACAGCA	676	687
AACAGCAAAAGCA	681	693
GCAAAAGCAAAA	685	696
CAAAAGCAAAAA	686	697
GCAAAAACAGAA	691	702
GAAGCAGAAAAA	700	711
GAAAAAGCAGCA	706	717
GCGACAGCAACA	721	732
CGACAGCAACAGC	722	734
ACAGCAACAGCA	724	735
AGCAACAGCAAC	726	737
AACAGCAACAGC	729	740
ACAGCAACAGCA	730	741
AGCAGCAGCAAC	738	749
AGCAGCAACAGC	741	752
GCAGCAACAGCA	742	753
AGCAACAGCAAC	744	755
AACAGCAACAGC	747	758
ACAGCAACAGCA	748	759
CAGCAGCAGCAG	755	766
GCAGCAGAAAAG	760	771
GAAAAGGCAGCA	766	777
AAGGCAGCAACA	769	780
GGCAGCAACAGC	771	782
GCAGCAACAGCA	772	783
AGCAGCAGCAAA	780	791
GCAGCAGCAAAA	781	792
CAGCAGCAAAAG	782	793
AGCAGCATCAGC	795	806
CAGCATCAGCAG	797	808
GAAAAGGCAGCA	808	819
AAGGCAGCAACA	811	822
GGCAGCAACAGC	813	824
GCAGCAACAGCA	814	825
AGCAACAGCAAC	816	827
TCAGCAGAAACA	844	855
CAGCAGAAACAG	845	856
AGCAGAAACAGC	846	857
CAGAAACAGCCA	848	859
AAGCAAAAGCAG	860	871
CAGCAGCAGCAG	869	880
GAAAAAGCAGCA	880	891
GCAGCAGAAAAG	889	900
AAAAGGCAAAAG	896	907
AGGCAAAAGCAG	899	910
AAAAGAAGCAGA	918	929
AGAAGCAGAAGA	921	932
AAGCAGAAGAAG	923	934

AGCAGAAGAAGCA	924	936
CAGAAGAAGCAG	926	937
AAGAAGCAGCAG	929	940
AAGCAGCAGAAA	932	943
AACAGAAGAAGA	948	959
AGAAGAAGAAAA	951	962
GAAGAAGAAAAAG	952	964
GAAGAAAAAGCAA	955	967
AACAGCAAAAGCA	975	987
AAGCAGCAGCAA	995	1006
AGCAGCAGCAAC	996	1007
AACGGAAGCGGA	1005	1016
GCAAAAGCAACAG	1018	1030
CAGCAGAGGCAG	1040	1051
GCAGAGGCAGCAGC	1042	1055
GGCAGCAGCAGA	1047	1058
AGCAGAAGCACT	1053	1064
AGGAGGTAAAAA	1088	1099
AAAAACAGCAACA	1095	1107
TCAGGATAATTC	1113	1124
CTGGAGAAAAAC	1139	1150
AGAAAAACAAGA	1143	1154
GAAAAACAAGAG	1144	1155
CAAGAAAAAGAA	1162	1173
ACGACAAGAAAA	1179	1190
CAAGAAAAAGAA	1183	1194
GAAAAAGAACAG	1186	1197
AACAGCATGAAAA	1193	1205
GCATGAAAAGCA	1197	1208
ATGAAAAGCAGCA	1199	1211
AAAAGCAGCAACA	1202	1214
GCAGCAACACCA	1206	1217
CAACACCAACAG	1210	1221
CAACAGCGTGAA	1216	1227
AACATTCCGCAG	1226	1237
CCGCAGGAAATG	1232	1243
CCGAAAGAAAAA	1255	1266
CGAAAGAAAAAA	1256	1267
GAAAGAAAAAAC	1257	1268
TGGTACAAATGC	1275	1286
TACAAATGCAAC	1278	1289
GACGACAGTGAC	1300	1311
GACAGTGACGGC	1303	1314
CCCCTCTTTTGC	1340	1351
TTCTTGTTGCGT	1358	1369
GTGGTGGCCGCGTGA	1387	1401

ANEXO – Parâmetros do BLAST

Esse anexo apresenta a relação dos parâmetros do BLAST usados para o BLASTN, indicando para cada um o valor padrão que é utilizado caso não seja explicitamente definido e os programas do BLAST que o utilizam (NCBI, 2008b).

-a [inteiro]

Padrão: 1 **Programas: Todos**

Define o número de processadores a ser utilizado.

-A [inteiro]

Padrão: blastn 0, outros 40 **Programas: Todos**

Define o tamanho da janela de *hits* múltiplos. Quando o BLAST é definido para o modo duplo-*hit*, essa opção requere dois *hits* de palavra na mesma diagonal estarem [inteiro] letras uma da outra para estender de alguma delas. Quanto maior o [inteiro], mais sensível será o BLAST. Definir [inteiro] como zero define o comportamento padrão de 40, exceto para o BLASTN, cujo padrão é o *hit* de palavra única.

-b [inteiro]

Padrão: 250 **Programas: Todos**

Trunca o relatório para [inteiro] alinhamentos. Não há aviso quando se excede esse limite, logo é geralmente bom definir um [inteiro] muito alto a não ser que se esteja interessado apenas nos melhores *hits*.

-B [inteiro]

Padrão: 0 **Programas: blastn, tblastn**

Especifica o número de *queries* concatenadas.

-d [banco de dados]

Padrão: nr **Programas: Todos**

Identifica o banco de dados a ser buscado. [banco de dados] já deve estar formatado pelo formatdb. Pode-se combinar múltiplos bancos de dados em um único banco de dados virtual colocando-se os bancos de dados individuais entre aspas. Não se pode misturar bancos de dados de nucleotídeos e aminoácidos. As estatísticas reportadas são baseadas nos tamanhos dos bancos de dados combinados. Bancos de dados virtuais podem exceder os limites de tamanho de arquivo impostos pelo sistema operacional.

-e [número real]

Padrão: 10 **Programas: Todos**

Define o limiar do *E-value* para manter alinhamentos. Esse valor descreve a frequência com que um alinhamento com uma dada pontuação é esperado de acontecer aleatoriamente.

-E [inteiro]

Padrão: blastn 2, outros 1 **Programas: Todos**

Define a penalidade para extensão de um gap. O parâmetro –G controla a penalidade inicial de abertura de um gap. O valor padrão de penalidade de gap para programas que não o blastn depende da matriz de pontuação.

-F [T/F], -F [string]

Padrão: T, ver abaixo **Programas: Todos**

Filtra a sequência *query* para subsequências de baixa complexidade. A filtragem de complexidade é geralmente uma boa idéia, mas pode quebrar HSPs longas em várias HSPs menores devido a segmentos de baixa complexidade. Isso pode fazer com que alguns alinhamentos caiam abaixo do limiar de significância e sejam perdidos. Para prevenir isso, desativa-se o filtro (não recomendado) ou se usa mascaramento em que o filtro somente é usado na fase de semeamento de palavras, mas não na fase de extensão.

-g [T/F]

Padrão: T **Programas: blastn, blastp, blastx, tblastn**

Realiza alinhamento com gaps. Definindo como F invoca o estilo mais antigo, sem gaps, de alinhamento.

-G [inteiro]

Padrão: blastn 5, outros 11 **Programas: Todos**

Penalidade inicial para abrir um gap. Penalidades para extensão do gap são controladas pelo parâmetro –E. –G 0 invoca o comportamento padrão, e definir –G como zero é impossível a menos que –g F seja definido, o que desabilita o alinhamento com gaps. As penalidades de gap padrão para programas que não o blastn dependem da matriz de pontuação.

-i [arquivo de entrada]

Padrão: stdin **Programas: Todos**

Se –i não é incluído na linha de comando, o BLAST espera entrada do stdin (ou seja, ele irá esperar indefinidamente até que o usuário informe um arquivo FASTA pelo teclado). Se o arquivo de entrada contém múltiplas sequências, o BLAST executará em cada sequência em ordem, e a saída conterá relatórios concatenados do BLAST.

-I [T/F]

Padrão: F **Programas: Todos**

Mostra os números de identificação GenInfo (GI) nas linhas de definição. Um GI é um identificador numérico único dado para uma sequência no GenBank.

-K [inteiro]

Padrão: 0 – Desativado **Programas: Todos**

O número de melhores *hits* a manter de uma região. Essa opção é útil quando se deseja limitar o numero de alinhamentos que podem se aglomerar em uma sessão da *query*. É mais útil quando os valores de –b ou –v são baixos, e os alinhamentos abundantes fazem com que alinhamentos com pontuação mais baixa sejam retirados do relatório. Se ativado, o valor de 100 é recomendado.

-L [string]

Padrão: Opcional **Programas: Todos**

O local na sequência *query*. Essa opção permite que a busca seja limitada a uma subsequência da sequência *query*. Por exemplo, para buscar apenas as letras de 21 a 50, adicione o parâmetro –L “21,50”. Os alinhamentos não se estenderão fora da região especificada.

-o [arquivo de saída]

Padrão: Opcional **Programas: Todos**

Determina um arquivo de saída para os resultados da busca. Caso não seja usado, a saída é impressa no stdout.

-p [nome do programa]

Padrão: Nenhum: parâmetro obrigatório **Opções: blastn, blastp, blastx, tblastn, tblastx, psitblastn**

-P [0/1]

Padrão: blastn 1, outros 0 **Programas: Todos**

Especifica o algoritmo de *hit* duplo ou *hit* único. A opção *hit* duplo necessita de dois *hits* de palavra na mesma diagonal para estender a partir de qualquer delas. Quando definido no modo *hit* duplo, o parâmetro *–A* especifica quão próximo os dois *hits* devem estar para iniciar a extensão. –P 0 especifica *hit* duplo; -P 1 especifica *hit* único.

-q [inteiro negativo]

Padrão: -3 **Programas: blastn**

Define a penalidade para uma desigualdade de nucleotídeos. Ver também *–r*. A escolha de [inteiro] para *–q* e *–r* é muito importante porque eles determinam as frequências alvo. Os valores padrão *–r* 1 *–q* -3 são mais efetivas para alinhar sequências que são 99% idênticas.

-r [inteiro]

Padrão: 1 **Programas: blastn**

Define a pontuação para uma igualdade de nucleotídeos. Veja o parâmetro *–q*.

-S [1..3]

Padrão: 3 **Programas: blastn, blastx, tblastx**

Determina qual fita das sequências baseadas em DNA será buscada.

Opções:

1 – fita de entrada.
2 – complemento reverso da entrada.
3 – ambas.

-T [T/F]

Padrão: F **Programas: Todos**

Produz saída HTML com links do sumário no topo do relatório para os alinhamentos mais abaixo.

-v [inteiro]

Padrão: 500 **Programas: Todos**

Define o número de sequências do banco de dados para as quais mostrar as descrições resumo de linha única no topo do relatório do BLAST.

-W [inteiro]

Padrão: blastn 11, outros 3 **Programas: Todos**

Define o tamanho da palavra para a busca inicial de palavras. O valor mínimo de palavra para o blastn é 7. Tamanhos de palavras para blastp, blastx, tblastn e tblastx são 2 ou 3.

-X [inteiro]

Padrão: blastn 30, outros 15 **Programas: Todos, exceto tblastx**

Define o valor X2 de *dropoff* para alinhamentos com gaps. O valor é medido em bits. Valores menores de X2 resultam em terminações antecipadas de extensões. O ajuste desse parâmetro é geralmente desnecessário.

-y [inteiro]

Padrão: blastn 20, outros 7 **Programas: Todos**

Define o valor X1 de *dropoff* (em bits) para extensões. Quando mais baixo X1 é definido, mais curta será a extensão. É raramente necessário ajustar esse parâmetro.

-Y [número real]

Padrão: 0 **Programas: Todos**

Tamanho efetivo do espaço de busca. É o tamanho do banco de dados multiplicado pelo tamanho da *query*. Se *–y* não é definido ou é definido como zero os tamanhos reais do banco de dados e da *query* são usados.

-z [número real]

Padrão: 0 **Programas: Todos**

Tamanho efetivo do banco de dados. Essa opção é útil para manter a estatísticas consistentes ao longo do tempo à medida bancos de dados crescem. Se *–z* não é definido ou é definido como zero o tamanho real do banco de dados é usado.

-Z [inteiro]

Padrão: 25 **Programas: Todos**

Define o valor X3 de *dropoff* (em bits) para extensões, mas é limitado pelo valor de X2. Geralmente não é necessário ajustar esse parâmetro.

REFERENCIAL BIBLIOGRÁFICO

ALBERTS, B.; BRAY, D.; HOPKIN, K.; JOHNSON A.; LEWIS, J.; RAFF M.; ROBERTS, K.; WALTER, P. **Fundamentos da Biologia Celular.** 2.ed. Porto Alegre: Artmed, 2006. 740 p.

ANDRADE, L. O.; ANDREWS, N. W. The *Trypanosoma cruzi* host cell interplay: location, invasion, retention. **Nature Reviews Microbiology**, v. 3, n. 10, p. 819-823, oct. 2005.

AVERY, V. M.; ADRIAN, D. L.; GORDON, D. L. Detection of mosaic protein mRNA in human astrocytes, **Immunology and Cell Biology**, v. 71, n. 3, p. 215-219, june 1993.

BROWN, T. A. **Genomes**. 2. ed. Oxford: BIOS Scientific Publishers, 2002. 572 p.

CLARK, F. **An Introduction to BLAST**. 2006. Disponível em: <http://clarkfrancis.com/blast/Blast_what_and_how.html>. Acesso em: 05 out. 2008.

DEGRAVE, W. ***Trypanosoma cruzi*:** o genoma. Rio de Janeiro. Disponível em: <http://www.fiocruz.br/chagas/cgi/cgilua.exe/sys/start.htm?sid=14>. Acesso em: 05 maio 2008.

DIAS, J. C. P. Notas sobre o *Trypanosoma cruzi* e suas características bio-ecológicas, como agente de enfermidades transmitidas por alimentos. **Revista da Sociedade Brasileira de Medicina Tropical**, v. 39, n. 4, p. 370-375, jul/ago 2006.

DOOLITTLE, R.F. The multiplicity of domains in proteins. **Annual Review of Biochemistry**, v. 64, p. 287-314, july 1995.

EL-SAYED N.M.; MYLER, P.J.; BARTHOLOMEU, D.C.; NILSSON, D.; AGGARWAL, G.; TRAN, A.N.; GHEDIN, E.; WORTHEY, E.A.; DELCHER, A.L.; BLANDIN, G.; WESTENBERGER, S.J.; CALER, E.; CERQUEIRA, G.C.; BRANCHE, C.; HAAS, B.; ANUPAMA, A.; ARNER, E.; ASLUND, L.; ATTIPOE, P.; BONTEMPI, E.; BRINGAUD, F.; BURTON, P.; CADAG, E.; CAMPBELL, D.A.; CARRINGTON, M.; CRABTREE, J.; DARBAN, H.; DA SILVEIRA, J.F.; DE JONG, P.; EDWARDS, K.; ENGLUND, P.T.; FAZELINA, G.; FELDBLYUM, T.; FERELLA, M.; FRASCH, A.C.; GULL, K.; HORN, D.; HOU, L.; HUANG, Y.; KINDLUND, E.; KLINGBEIL, M.; KLUGE, S.; KOO, H.; LACERDA, D.; LEVIN, M.J.; LORENZI, H.; LOUIE, T.; MACHADO, C.R.; MCCULLOCH, R.; MCKENNA, A.; MIZUNO, Y.; MOTTRAM, J.C.; NELSON, S.; OCHAYA, S.; OSOEGAWA, K.; PAI, G.; PARSONS, M.; PENTONY, M.; PETTERSSON, U.; POP, M.; RAMIREZ, J.L.; RINTA, J.; ROBERTSON, L.; SALZBERG, S.L.; SANCHEZ, D.O.; SEYLER, A.; SHARMA, R.; SHETTY, J.; SIMPSON, A.J.; SISK, E.; TAMMI, M.T.; TARLETON, R.; TEIXEIRA, S.; VAN AKEN, S.; VOGT, C.; WARD, P.N.; WICKSTEAD, B.; WORTMAN, J.; WHITE, O.; FRASER, C.M.; STUART, K.D.; ANDERSSON, B. The genome sequence of *Trypanosoma cruzi*, etiologic agent of Chagas disease. **Science**, v. 309, n. 5733, p. 409-415, july 2005.

FRASCH, A. A. C. Functional diversity in the trans-sialidase and mucin families in *Trypanosoma cruzi*. **Parasitology Today**, v. 16, n. 7, p. 282-286, july 2000.

FREUDENRICH, C. **Como Funciona o DNA.** Disponível em: < http://saude.hsw.uol.com.br/dna1.htm>. Acesso em: 04 maio 2008.

GABORIAUD, C.; ROSSI, V.; FONTECILLA-CAMPS, J. C.; ARLAUD, G. J. Evolutionary Conserved Rigid Module-domain Interactions can be Detected at the Sequence Level: The Examples of Complement and Blood Coagulation Proteases. **Journal of Molecular Biology**, v. 282, n. 2, p. 459-470, sep 1998.

GOLDENBERG, S. ***Trypanosoma cruzi*:** Regulação da expressão gênica. Rio de Janeiro. Disponível em: <http://www.fiocruz.br/chagas/cgi/cgilua.exe/sys/start.htm?sid=14>. Acesso em: 05 maio 2008.

GOMES, A. S.; SOUZA, T. R. **Uma Metodologia para Identificação de Módulos Formadores de Sequências de Proteínas Mosaicas do *Trypanosoma cruzi* a partir do Proteoma do Parasito Utilizando a Ferramenta BLAST**. 2008. 47p. Monografia (Graduação em Ciência da Computação) – Universidade Federal de Lavras, Lavras, MG.

GUIMARÃES, A. C. R **Identificação, Classificação e Anotação de Enzimas Análogas em Tripanosomatídeos**. 2006. 122 p. Dissertação (Mestrado em Ciências) – Instituto Oswaldo Cruz/Fundação Oswaldo Cruz, Rio de Janeiro.

HEGYI, H.; BORK, P. On the classification and evolution of protein modules. **Journal of Protein Chemistry**, v. 16, n. 5, p. 545-551, july 1997.

HIB, J.; PONZIO, R.; ROBERTIS, E. M. F. **Biologia Celular e Molecular**. 14. ed., Rio de Janeiro: Guanabara Koogan, 2003. 432 p.

INCOGEN. **NCBI Blastn.** Disponível em <http://www.incogen.com/public_documents/vibe/details/NcbiBlastn.html>. Acesso em: 13 ago. 2008.

JUNG, C. F. **Metodologia Para Pesquisa & Desenvolvimento**. Rio de Janeiro: Axcel Books do Brasil, 2004. 312 p.

KAHN, S. J.; NGUYEN D.; NORSEN, J.; WLEKLINSKI, M.; GRANSTON, T.; KAHN, M. *Trypanosoma cruzi*: monoclonal antibodies to the surface glycoprotein superfamily differentiate subsets of the 85-kDa surface glycoproteins and confirm simultaneous expression of variant 85-kDa surface glycoproteins. **Experimental Parasitology**, v. 92, n. 1, p. 48-56, may 1999.

KAMOUN, P.; LAVOINNE, A.; VERNEUIL, H de. **Bioquímica e Biologia Molecular**. Rio de Janeiro: Guanabara Koogan, 2006. 444 p.

KANEHISA, M. **Post-genome Informatics**. Oxford: Oxford University Press, 2000. 148 p.

KOLKMAN, J. A.; STEMMER, W. P. C. Directed evolution of proteins by *exon* shuffling. **Nature Biotechnology**, v. 19, n. 5, p. 423-428, may 2001.

KORF, I.; YANDELL, M.; BEDELL, J. **BLAST:** An essential guide to the Basic Local Alignment Search Tool. Sebastopol: O'Reilly, 2003. 339 p.

LEVY, B. **Estudo Aponta Possibilidade de Quimioterapia Natural para Doença de Chagas**. Rio de Janeiro, 2006. Disponível em: <http://www.ioc.fiocruz.br/pages/informerede/corpo/noticia/2006/fevereiro/23_02_06_02.htm>. Acesso em: 05 maio 2008.

MAYER, H. **A collection of evaluated bioinformatics programs and databases:** Sequence Similarity. Disponível em <http://homepage.univie.ac.at/herbert.mayer/>. Acesso em 11 ago. 2008.

MOUNT, D. W. **Bioinformatics:** sequence and genome analysis. 2. ed. Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 2004. 692 p.

NCBI – NATIONAL CENTER FOR BIOTECHNOLOGY INFORMATION. **Basic Local Alignment Search Tool.** Disponível em: <http://www.ncbi.nlm.nih.gov/blast/Blast.cgi?CMD=Web&PAGE_TYPE=BlastFAQs>. Acesso em : 13 ago. 2008a.

NCBI – NATIONAL CENTER FOR BIOTECHNOLOGY INFORMATION. **Program Parameters for Blastall.** Disponível em: <http://www.ncbi.nlm.nih.gov/staff/tao/URLAPI/blastall/blastall_node21.html>. Acesso em: 16 ago. 2008b.

NHGRI – NATIONAL HUMAN GENOME RESEARCH INSTITUTE. **Transcriptome**. 2008. Disponível em: <http://www.genome.gov/13014330>. Acesso em: 06 maio 2008.

NEVES, D. P.; MELO, A. L.; LINARDI, P. M.; VITOR, R. W. A. **Parasitologia Humana**. 11. ed. São Paulo: Atheneu, 2005. 494 p.

PATTHY, L. Modular exchange principles in proteins. **Current Opinions in Structural Biology**, v. 1, p. 351-361, 1991.

PROBST, C.; PAVONI, D. P.; GÓES, V. M.; MANHÃES, L.; DALLAGIOVANNA, B. M.; MORTARA, R. A.; ROMANHA, A. J.; SCHENKMAN, S.; BUCK, G. A.;GOLDENBERG, S.; KRIEGER, M. A. Modulation of *Trypanosoma cruzi* Transcriptome Assayed by Microarrays: Insights into the Regulome. In: REUNIÃO ANNUAL DA SBBq, 36, 2007, Salvador. **Anais…** XXXVI Reunião Annual da SBBq.

PROSDOCIMI, F.; CERQUEIRA, G. C.; BINNECK, E.; SILVA, A. F.; REIS, A. N.; JUNQUEIRA, A. C. M.; SANTOS, A. C. F.; NBANI JÚNIOR, A.; WUST, C. I.; CAMARGO FILHO, F.; KESSEDJIAN, J. L.; PETRETSKI, J. H.; CAMARGO, L. P.; FERREIRA, R. G. M.; LIMA, R. P.; PEREIRA, R. M.; JARDIM, S.; SAMPAIO, V. S. FOLGUERAS-FLATSCHART, A. V. Bioinformática: Manual do Usuário. **Biotecnologia Ciência & Desenvolvimento**, v. 29, p. 12-25, 2002.

SODRÉ, C. L; KALUME, D. E.; SILVA, M. E. R.; FERNANDES O. ***Trypanosoma cruzi*:** Proteoma. Disponível em: <http://www.fiocruz.br/chagas/cgi/cgilua.exe/sys/start.htm?sid=81>. Acesso em: 05 maio 2008.

SOUSA, D. X.; LIFSCHITZ, S. **A avaliação do *E-value* para execução do BLAST sobre bases de dados fragmentadas.** 2007. 15 p. Monografia (Graduação em Ciência da Computação) – Pontífica Universidade Católica, Rio de Janeiro.

SOUZA, W. **Morfologia:** Métodos morfológico. Rio de Janeiro. Disponível em: <http://www.fiocruz.br/chagas/cgi/cgilua.exe/sys/start.htm?sid=12>. Acesso em: 05 maio 2008.

ZAMBALDE, A. L.; PÁDUA, C. I. P. S.; ALVES, R. M. **O documento científico em Ciência da Computação e Sistemas de Informação**. Lavras, MG: DCC/UFLA, 2008.

ZORZETTO, R. **Reprodução desvendada: Identificação de região do núcleo do *Trypanosoma cruzi* pode facilitar o combate ao mal de Chagas**. 2005. Disponível em: <http://revistapesquisa.fapesp.br/index.php?art=2763&bd=1&pg=1&lg=>. Acesso em: 05 maio 08.