Uma abordagem para detecção e remoção de artefatos em seqüências ESTs

Christian Baudet

Dissertação de Mestrado

Instituto de Computação
Universidade Estadual de Campinas

Uma abordagem para detecção e remoção de artefatos em seqüências ESTs

Christian Baudet[1]

01 de Dezembro de 2006

Banca Examinadora:

- Prof. Dr. Zanoni Dias (Orientador)
- Prof. Dr. Guilherme Pimentel Telles
  Instituto de Ciências Matemáticas e de Computação – ICMC – USP
- Prof. Dr. João Meidanis
  Instituto de Computação – IC – Unicamp
- Prof. Dr. Ricardo Dahab (Suplente)
  Instituto de Computação – IC – Unicamp

[1]**Apoio financeiro da Scylla Bioinformática.**

Uma abordagem para detecção e remoção de artefatos em seqüências ESTs

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Christian Baudet e aprovada pela Banca Examinadora.

Campinas, 01 de Dezembro de 2006.

Prof. Dr. Zanoni Dias (Orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Agradecimentos

Gostaria de dedicar este trabalho aos maiores responsáveis pela possibilidade de ele existir: meus pais. Sem o amor e o investimento que eles depositaram em mim durante todos estes anos, eu não teria conseguido chegar até aqui.

Também gostaria de agradecer ao professor Zanoni Dias pelo incentivo, pela disposição e pela paciência dispensados durante toda a orientação.

Não poderia esquecer de agredecer as pessoas que deram apoio e incentivo para que eu ingressasse na área de bioinformática. Agradecimentos ao professor João Meidanis e a todo o pessoal do Núcleo de Bioinformática da Embrapa Informática Agropecuária, principalmente ao Goran Neshich e ao Roberto Higa.

Agradecimentos especiais são reservados aos meus irmãos que sempre se esforçaram para me tirar do sério e fazer com que eu me tornasse uma pessoa "menos" estressada.

Finalmente, agradeço a todos os meus amigos que me acompanharam ao longo dessa jornada. Aos amigos da Fundação Bradesco, da Unicamp, da Scylla e do dia-a-dia deixo os meus mais sinceros agradecimentos.

Resumo

O seqüenciamento de ESTs (*Expressed Sequence Tag*) [2] é uma técnica que trabalha com bibliotecas de cDNAs tendo como objetivo a obtenção de uma boa aproximação para o índice gênico, que é a listagem de genes existentes no genoma do organismo estudado.

Antes da serem analisadas, as seqüências obtidas do seqüenciamento dos ESTs devem ser processadas para eliminação de artefatos. Artefatos são trechos que não pertencem ao organismo ou que possuem baixa qualidade ou baixa complexidade. Trechos de vetores, adaptadores e caudas poli-A podem ser citados como exemplos de artefatos.

A eliminação dos artefatos deve ser feita para que a análise das seqüências produzidas no projeto não seja prejudicada por estes "ruídos". Por exemplo, artefatos presentes em seqüências freqüentemente produzem erros em processos de clusterização, pois eles podem determinar se seqüências serão unidas em um mesmo cluster ou separadas em clusters diferentes.

Observando a importância da realização de um bom processo de limpeza das seqüências, o trabalho desenvolvido nesta dissertação teve como principal objetivo a obtenção de um conjunto eficiente de procedimentos de detecção e remoção de artefatos.

Este conjunto foi produzido a partir de uma nova estratégia de detecção de artefatos. Normalmente, cada projeto de seqüenciamento possui seu próprio conjunto de procedimentos dividido em várias etapas. Estas etapas são, em geral, ligadas entre si e o resultado de uma pode influenciar o resultado de outra. A nossa estratégia visa a realização destas etapas de forma totalmente independente.

Além da avaliação desta nova estratégia, o trabalho também realizou um estudo mais detalhado sobre dois tipos de artefatos: baixa qualidade e derrapagem. Para cada um deles, algoritmos foram propostos e validados através de testes com conjuntos de seqüências produzidas em projetos reais de seqüenciamento.

O conjunto final de procedimentos, baseado nos estudos desenvolvidos durante a escrita deste texto, foi testado com as seqüências do projeto SUCEST [100, 103, 113] e mostrou bons resultados. O clustering produzido com as seqüências processadas por nossos métodos apresentou melhores consistência interna e externa e menores taxas de redundância quando comparado ao clustering original do projeto.

Abstract

Expressed Sequence Tag (EST) Sequencing [2] is one technique that works with cDNA libraries. It aims to achieve a good approximation for the gene index of an organism.

Before analyzing the sequences obtained by sequencing ESTs, they must be processed for artifact removal. An artifact is a sequence that does not belong to the studied organism or that has low quality or low complexity. As example of artifacts, we have adapters, poly-A tails, vectors, etc.

Artifacts removal must be performed because their presence can produce "noises" in the sequencing project data analysis. For example, artifact can join two sequences in a same cluster inappropriately or separate them in two different clusters when they should be put together.

Motivated by the sequence cleaning process importance, our main objective in this work was to develop an efficient set of procedures to detect and to remove sequence artifacts.

Usually, each EST sequencing project has its own procedure set divided in many steps. These steps are, in general, linked and the result of one given step might influence the result of the next one. Our strategy was to perform each step independently assuring that any execution order of those steps would lead to the same result.

Additionally to the new strategy evaluation, this work also studied detailedly two type of artifacts: low quality and slippage. For each one, algorithms were proposed and validated through tests with sequences of real sequencing projects.

The final set of procedure, developed in this work, was evaluated using the sequences of the SUCEST project [100, 103, 113] and produced good results. The resulting clustering from our method has better external and internal consistency and lower redundancy rate than those produced by the SUCEST project clustering.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

Na última década do século XX, o mundo observou duas grandes evoluções. A primeira, refere-se ao crescimento do poder de processamento de computadores. A segunda que pode ser considerada, em parte, como uma das conseqüências da primeira, refere-se ao avanço das pesquisas genômicas, que passaram a realizar o estudo dos genomas dos organismos a partir do seqüenciamento em larga-escala de suas seqüências de DNA.

A operação de seqüenciamento consiste na determinação das seqüências de nucleotídeos de um trecho de DNA. A partir das seqüências de bases, os cientistas podem realizar diversas atividades como, por exemplo, a montagem do genoma, a determinação dos genes existentes no genoma e a descoberta de padrões de expressões gênicas.

Diversas técnicas foram desenvolvidas para obtenção de DNA para seqüenciamento. Uma dessas técnicas consiste na produção de bibliotecas de moléculas de cDNA (DNA complementar), que é produzido a partir do complemento da molécula de mRNA (RNA mensageiro).

O seqüenciamento de ESTs (*Expressed Sequence Tag*) [2] é uma técnica que trabalha com bibliotecas de cDNAs tendo como objetivo a obtenção de uma boa aproximação para o índice gênico, que é a listagem de genes existentes no genoma do organismo estudado.

Antes de serem analisadas, as seqüências obtidas do seqüenciamento dos ESTs devem ser processadas para eliminação de artefatos. Artefatos são trechos que não pertencem ao organismo ou que possuem baixa qualidade ou baixa complexidade. Trechos de vetores, adaptadores e caudas poli-A podem ser citados como exemplos de artefatos.

A eliminação dos artefatos deve ser feita para que a análise das seqüências produzidas no projeto não seja prejudicada por estes "ruídos". Por exemplo, artefatos presentes em seqüências freqüentemente produzem erros em processos de clusterização, pois eles podem determinar se seqüências serão unidas em um mesmo cluster ou separadas em clusters diferentes.

Observando a importância da realização de um bom serviço de limpeza das seqüências,

o trabalho desenvolvido nesta dissertação tem como principal objetivo obter um conjunto eficiente de procedimentos de detecção e remoção de artefatos.

Este conjunto é produzido a partir de uma nova estratégia de detecção de artefatos. Normalmente, cada projeto de seqüenciamento possui seu próprio conjunto dividido em várias etapas. Estas etapas são, em geral, ligadas entre si e o resultado de uma pode influenciar o resultado de outra. A nossa estratégia visa a realização destas etapas de forma totalmente independente.

Além da avaliação desta nova estratégia, o trabalho também realiza um estudo mais detalhado sobre dois tipos de artefatos: baixa qualidade e derrapagem. Para cada um deles, algoritmos são propostos e validados através de testes com conjuntos de seqüências produzidas em projetos de seqüenciamento.

Como principais contribuições desta dissertação, podemos citar:

- Desenvolvimento e validação de uma nova estratégia de detecção de artefatos;
- Proposta e avaliação de novos algoritmos para detecção de artefatos de derrapagem e determinação do método mais adequado;
- Estudo aprofundado sobre a remoção de artefatos de baixa qualidade que determinou a escolha de um algoritmo e de valores de parâmetros adequados a esta finalidade;
- Criação e validação de um conjunto completo de procedimentos para detecção e remoção de artefatos baseados nos resultados obtidos ao longo do desenvolvimento da dissertação.

Durante o desenvolvimento da dissertação, os resultados parciais foram submetidos a diversos congressos. O estudo de desenvolvimento e avaliação da nova estratégia foi apresentado no congresso "Brazilian Symposium on Bioinformatics 2005 (BSB2005)", realizado em Julho de 2005, em São Leopoldo – RS, sob o título "New EST Trimming Strategy". Um resumo estendido foi publicado nos anais do congresso [10]. O estudo completo foi depositado como relatório técnico, identificado pelo código "IC-05-09", no Instituto de Computação – Unicamp [11]. A proposta e avaliação de algoritmos para detecção de artefatos de derrapagem foram apresentadas como pôster no congresso "X-Meeting 2005", realizado em Outubro de 2005, em Caxambu – MG, sob o título "Analysis of slipped sequences in EST projects" e posteriormente aceitas como artigo de mesmo título para publicação na revista "Genetics and Molecular Research" [12]. O estudo de artefatos de baixa qualidade foi apresentado como pôster, entitulado "Low quality trimming on SUCEST ESTs", no congresso "X-Meeting 2006", realizado em Agosto de 2006, em Fortaleza – CE. Finalmente, a descrição do conjunto completo de procedimentos de detecção e remoção de artefatos, resultado final desta dissertação, foi apresentado como

pôster no congresso "14^{th} Annual International Conference On Intelligent Systems For Molecular Biology (ISMB2006)", também realizado Agosto de 2006, em Fortaleza – CE, sob o título "New EST trimming procedure applied to SUCEST sequences".

O texto desta disssertação está organizado da seguinte maneira. O Capítulo 2 apresenta uma série de conceitos básicos com o objetivo de melhor introduzir o contexto em que este trabalho está inserido. O Capítulo 3 descreve a nova estratégia de identificação e remoção de artefatos desenvolvida. O Capítulo 4 traz os estudos realizados para o desenvolvimento de métodos de identificação de artefatos de derrapagem. O Capítulo 5 discute o trabalho desenvolvido para melhora do procedimento de detecção de baixa qualidade. O Capítulo 6 descreve o conjunto completo de métodos de detecção e remoção de artefatos produzido a partir da combinação dos resultados obtidos nos capítulos anteriores. Finalmente, o Capítulo 7 apresenta as conclusões da dissertação, e propõe algumas extensões para o trabalho.

Capítulo 2

Conceitos Básicos

Neste capítulo faremos uma apresentação do contexto básico em que este trabalho está inserido, através da descrição de uma série de tópicos e termos comumente encontrados na área.

2.1 Genética

A genética é uma área da biologia que se dedica ao estudo de genes. Os genes são as principais unidades de informação biológica contidas no material genético de um organismo. Um gene armazena as instruções para sintetização de móleculas que participam de reações metabólicas que ocorrem na célula.

Todo organismo vivo apresenta características observáveis tais como cor, estrutura, comportamento, etc. Estas características formam o seu fenótipo, que é determinado pela interação entre o genótipo do organismo e o meio em que ele vive. O genótipo é o conjunto de informações contidas no material genético de um organismo. Estas informações ditam como o organismo será construído e mantido. Elas são replicadas a cada divisão celular e podem ser herdadas no momento da reprodução.

2.1.1 Mendel

O austríaco Gregor Mendel é considerado por muitos o primeiro geneticista da história. Em 1865 ele descreveu os princípios da herança genética [70].

Através de um estudo realizado com ervilhas, Mendel concluiu que cada pé de ervilha possuía dois alelos para cada gene, mas exibiam apenas um fenótipo. Um alelo é uma das duas ou mais formas diferentes de um mesmo gene. Estas formas diferentes de um gene podem resultar em diferentes fenótipos, como, por exemplo, as diferentes texturas (lisa ou rugosa) que a casca de uma ervilha possui.

Este resultado exibido por Mendel é fácil de observar quando as plantas são homozigotas para uma determinada característica, ou seja, quando elas possuem dois alelos idênticos de um gene. Neste caso, como existem duas cópias de um mesma forma de um gene, apenas um fenótipo pode ser expresso pela planta.

Contudo, Mendel mostrou que quando plantas homozigotas com fenótipos diferentes eram cruzadas, todas as plantas descendentes apresentavam o mesmo fenótipo. Neste caso, todas eram heterozigotas pois, no momento do cruzamento, receberam um alelo diferente de cada ancestral. O fato de um fenótipo apenas ser exibido, neste caso, é explicado pela existência de um fenótipo dominante, que é sempre exibido quando pelo menos uma cópia do alelo responsável por ele está presente no material genético do organismo, e de um fenótipo recessivo, que só é exibido quando o organismo é homozigoto para a característica recessiva.

Mendel prosseguiu com seus experimentos que permitiram que ele estabelecesse as duas Leis da Genética. A Primeira Lei da Genética diz que os alelos segregam randomicamente, ou seja, se os alelos dos pais são A e a, então um membro da geração F_1 tem a mesma chance de herdar A ou a. A Segunda Lei da Genética diz que pares de alelos segregam independentemente, o que significa que a herança dos alelos do gene X é independente da herança dos alelos do gene Y.

O grande legado de Mendel para a genética moderna foi a maneira correta como ele descreveu os genes e o modo como eles são herdados. Hoje sabemos que dois alelos de um gene são carregados por dois cromossomos homólogos em uma célula diplóide e que a maneira como são transmitidos para futuras gerações, como descrito por Mendel, corresponde ao evento da produção dos gametas haplóides durante o fenômeno da meiose que ocorre na célula. Sabemos também que o mecanismo dominante-recessivo descrito por Mendel pode ser complicado pela existência de dominância incompleta, que ocorre quando o organismo heterozigoto apresenta um fenótipo intermediário em relação aos pais homozigotos com alelos diferentes. Outro complicante é a codominância que permite que os indivíduos heterozigotos apresentem tanto um como outro fenótipo. A única falha no trabalho de Mendel é que a Segunda Lei não prevê a possibilidade dos alelos de dois genes serem herdados juntos por estarem no mesmo cromossomo.

2.1.2 Cromossomos

Passado algum tempo após o trabalho de Mendel, em 1903, W. S. Sutton observou o comportamento dos cromossomos, que se duplicavam durante a divisão celular, e fez um paralelo com os padrões de herança dos genes.

Esta observação levou a hipótese de que os genes se encontravam nos cromossomos. Por volta da década de 1930, esta hipótese conhecida como “Teoria dos Cromossomos”

era universalmente aceita como correta.

Estudos citoquímicos realizados na mesma época mostraram que os cromossomos eram compostos por DNA e por proteínas. Naquela época, os pesquisadores imaginavam que o DNA era uma molécula que possuía pouca variabilidade e que, portanto, os genes deveriam ser feitos por proteínas, que são compostas por 20 aminoácidos e assumem diferentes formas.

Com o decorrer do tempo, os estudos mostraram que o DNA possuía grande variabilidade, mas ainda existia a dúvida sobre qual seria a composição do material genético. A resposta veio com dois experimentos [7, 46], realizados no meio do século XX, que mostraram que os genes eram compostos por DNA.

2.2 DNA & RNA

O DNA ou ácido desoxirribonucléico é um dos dois tipos de ácidos nucléicos encontrados dentro da célula de um organismo. Ele é um polinucleotídeo composto por quatro tipos de nucleotídeos diferentes. Cada nucleotídeo é composto por três partes: uma pentose denominada desoxirribose, um grupo fosfato e uma base nitrogenada que é diferente em cada um dos quatro tipos de nucleotídeos. As bases nitrogenadas podem ser pirimídicas (citosina e timina) e púricas (adenina e guanina). A Figura 2.1 exibe a estrutura molecular de um nucleotídeo que forma o DNA e as estruturas das possíveis bases.

O outro tipo de acído nucléico é o RNA ou ácido ribonucléico. A sua composição química é similar a do DNA. A diferença está no açúcar pentose que compõem o nucleotídeo, que é uma ribose no lugar da desoxirribose, e na existência do nucleotídeo que possui uma base nitrogenada chamada uracila, substituindo o nucleotídeo que possui a timina. A Figura 2.2 exibe a estrutura molecular de um nucleotídeo que forma o RNA e as estruturas das possíveis bases.

Um polinucleotídeo é formado pela ligação de nucleotídeos que são unidos entre si por uma ligação de ponte de fosfodiéster entre seus carbonos $5'$ e $3'$. Devido à polaridade deste tipo de ligação, a reação química necessária para extensão do polímero de DNA na direção $5' \rightarrow 3'$ é diferente da reação necessária para extensão na direção oposta. Todas as enzimas naturais de polimerização de DNA são capazes de lidar apenas com a reação na direção $5' \rightarrow 3'$. A Figura 2.3 mostra um pequeno polinucleotídeo, com destaque para as extremidades $5'$ e $3'$ e para a ponte de fosfodiéster.

2.2.1 A estrutura de dupla hélice

No meio do século XX, pesquisas mostravam que o DNA era composto por duas ou mais moléculas de polinucleotídeos arranjados de alguma maneira, que se descoberta

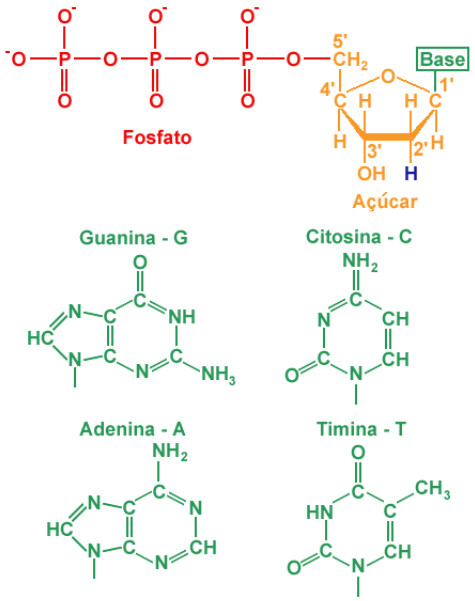

Figura 2.1: Nucleotídeos que formam o DNA.

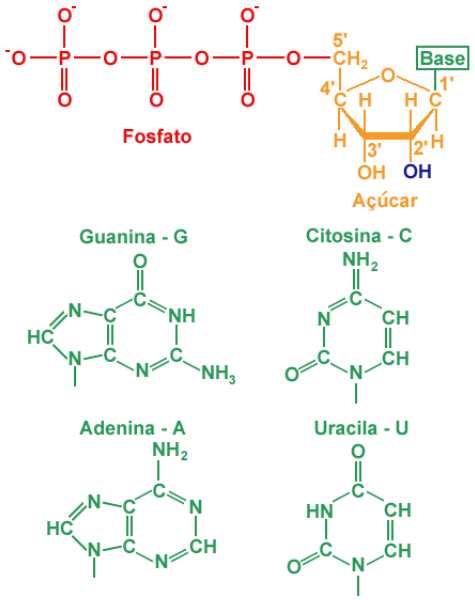

Figura 2.2: Nucleotídeos que formam o RNA. Note que a diferença do RNA para o DNA está na presença de um OH no lugar do H no açúcar e na presença da Uracila substituindo a Timina.

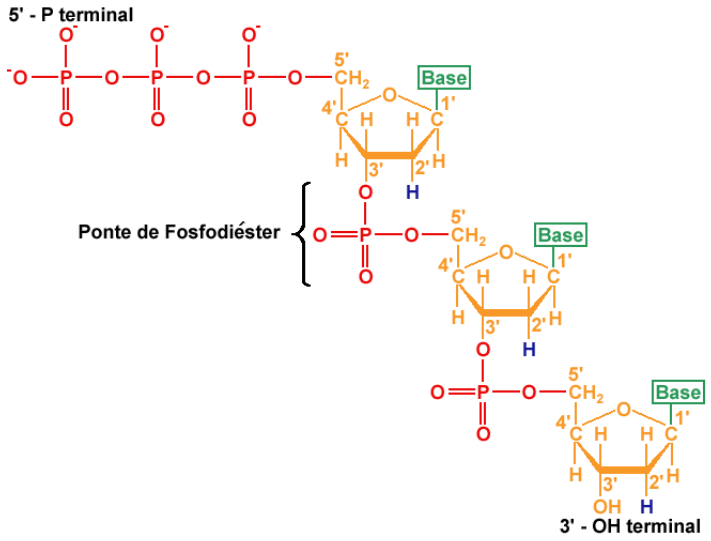

Figura 2.3: Representação de um pequeno polinucleotídeo com destaque para as extremidades 5′ e 3′ e para a ponte de fosfodiéster.

poderia fornecer pistas de como os genes funcionavam. Em 1953, Watson e Crick [114] no meio de uma corrida em busca da estrutura deste ácido nucléico, que envolveu diversos pesquisadores, publicaram um trabalho que exibia a estrutura de dupla hélice de DNA.

A dupla hélice de DNA é composta por duas fitas de polinucleotídeos que correm em direções opostas. Esta hélice é estabilizada por dois tipos principais de interações químicas. O primeiro tipo é o pareamento de bases, entre as duas fitas, que envolve a formação de pontes de hidrogênio entre uma adenina de uma fita e a timina da outra fita, ou entre a citosina de uma fita e a guanina da outra. O segundo tipo é a interação hidrofóbica que existe entre cada par de bases adjacentes e que adiciona estabilidade à dupla hélice. A Figura 2.4 exibe a estrutura de dupla hélice e as pontes de hidrogênio que se formam entre as bases dos nucleotídeos.

A razão da combinação de pares de bases (A com T e C com G) pode ser explicada, parcialmente pelas geometrias das bases nitrogenadas e pelas posições relativas dos grupos que participam das ligações químicas, parcialmente pelo fato de os pares terem que ser formados entre uma base púrica e uma pirimídica. Se o par fosse formado por duas bases púricas, ele seria muito grande para caber na hélice e, se fosse formado por duas bases pirimídicas, ele seria muito pequeno. Esta limitação, imposta pelos pares de bases possíveis, faz com que a replicação do DNA possa gerar cópias perfeitas de uma molécula pai a partir de fitas pré-existentes, que ditarão as seqüências das novas fitas como se fossem moldes. Esta síntese de DNA dependente de um modelo é utilizada por todas as enzimas

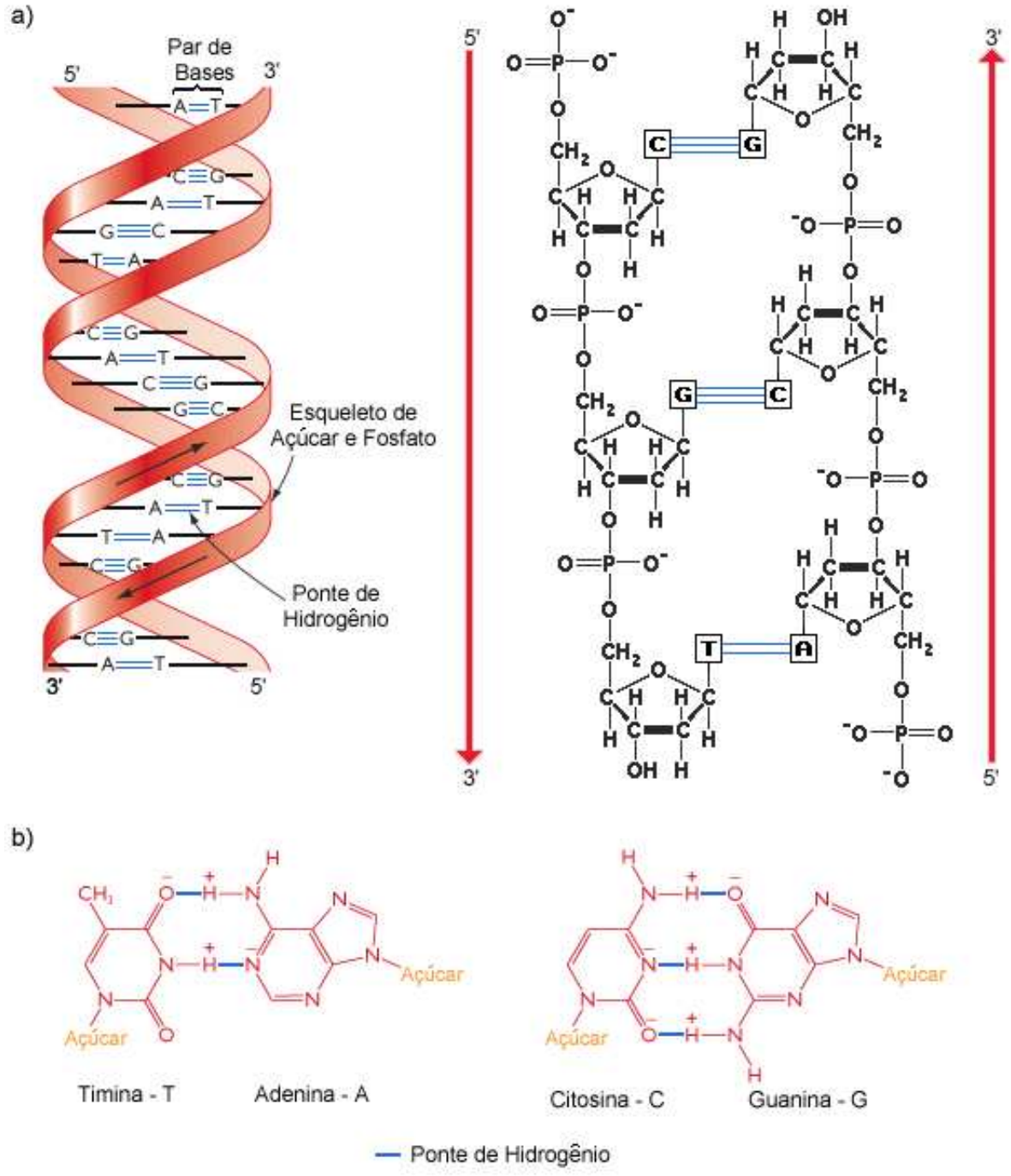

Figura 2.4: a) Estrutura de dupla hélice e esquema de ligação das duas cadeias de polinucleotídeos. b) Pontes de hidrogênio existentes entre os pares de bases.

celulares de polimerização de DNA. Da mesma forma, as enzimas de polimerização de RNA também se utilizam deste modelo para produção da fita de RNA a partir de um molde de DNA, o que permite que as cópias preservem a informação biológica contida no DNA genômico.

2.2.2 O DNA como material genético

A possibilidade de se produzir cópias perfeitas a partir de uma molécula de DNA faz com que ela seja perfeita para carregar as informações genéticas de um organismo. A conformação de dupla hélice força a manutenção da seqüência de nucleotídeos existente no genoma do organismo. Se o DNA fosse formado apenas por uma fita de polinucleotídeo, inserções e remoções de nucleotídeos poderiam ocorrer com grande freqüência, e além disso, a fita poderia ser facilmente partida em pedaços.

Por causa da característica citada acima, os genomas da maior parte dos organismos existentes são feitos por DNA (existem alguns vírus, conhecidos como retrovírus, que possuem o material genético composto por RNA).

2.3 Genoma

Todo organismo possui um genoma que contém toda a informação biológica necessária para construí-lo e mantê-lo vivo. Esta informação é codificada em seqüências de nucleotídeos em suas moléculas de DNA ou RNA e é dividida em unidades discretas chamadas genes.

A informação contida em um gene é lida por proteínas que se ligam ao genoma em posições apropriadas e iniciam uma série de reações bioquímicas conhecidas como expressão gênica. Para organismos com genoma feito de DNA, estas reações são divididas em dois estágios: transcrição e tradução.

2.3.1 Transcrição e tradução

A transcrição é a produção de uma cópia feita de RNA de um gene contido no genoma. Ela se inicia com a ligação da enzima RNA polimerase e outros fatores de transcrição ao genoma, próximo à localização do gene formando um complexo de transcrição. A partir deste complexo a fita de RNA é produzida. Esta fita de RNA é conhecida como mRNA ou RNA mensageiro.

A tradução é a síntese de proteínas a partir das cópias transcritas de RNA. A seqüência de aminoácidos da proteína é determinada com base no código genético do organismo. O código genético é uma tabela que possui a correspondência entre cada tripla de nucleotídeo,

denominada códon, e o aminoácido que será utilizado na síntese da proteína. Esta tabela é redundante, pois existem 64 códons possíveis distribuídos entre 20 aminoácidos mais o STOP códon, que é o códon que indica o término da tradução. A tabela não é universal, ou seja, não é a mesma para todos os organismos. Na página do NCBI [74] é possível encontrar 17 tabelas diferentes. A Figura 2.5 exibe o código genético padrão, utilizado pela maior parte dos organismos. Ela corresponde à tabela número 1 na página do NCBI.

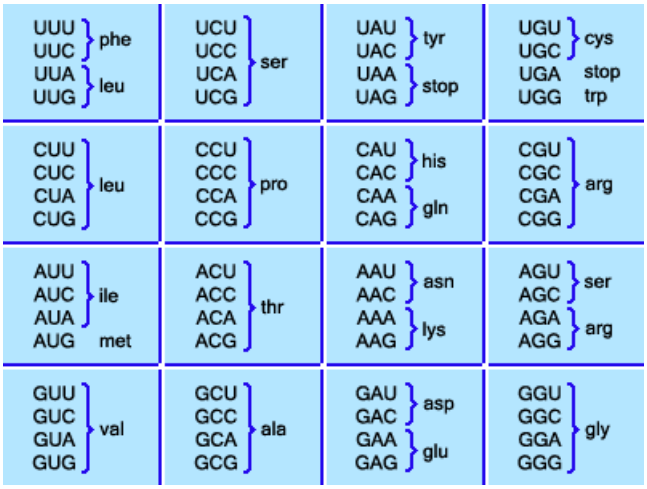

Figura 2.5: Código Genético utilizado pela maioria dos organismos. Cada tripla de nucleotídeos (códon) do mRNA corresponde a um aminoácido ou a um STOP códon. O ribossomo processará o mRNA produzindo a cadeia protéica segundo o código genético do organismo.

Esta descrição do processo de tradução é suficiente para a expressão gênica que ocorre em bactérias e arqueobactérias, pois elas possuem genomas mais simples. Em organismos mais complexos, antes da tradução ocorre o pré-processamento do mRNA (RNA mensageiro) para remoção dos íntrons, que são os trechos do gene que não codificam proteínas. O mRNA processado conterá apenas éxons, que são os trechos de DNA que serão realmente traduzidos.

Neste pré-processamento da molécula que será traduzida poderá ocorrer também o splice alternativo. O splice aternativo é a produção de um dos vários mRNAs possíveis a partir da combinação dos éxons existentes em um gene. Este é um evento comum em organismos mais complexos. Por exemplo, os cientistas que realizaram o seqüenciamento da mosca *Drosophila melanogaster* [1] verificaram que ela possui um número menor de genes que o verme *Caenorhabditis elegans*, algo incompatível com o fato da mosca apresentar maior complexidade física, o que torna necessário um conjunto mais amplo de proteínas.

Os cientistas verificaram então que a mosca possui uma quantia substancial de genes que produzem proteínas diferentes através do splice alternativo.

A tradução começa, de fato, com a ligação da fita de RNA já processada em um ribossomo, uma organela existente no interior da célula, e termina com a produção da proteína desejada. Esta proteína passará por um processamento e após isso adquirirá a sua conformação final.

Além dos genes que produzem proteínas, existem aqueles que produzem seqüências de rRNA (RNA ribossomal) e tRNA (RNA transportador).

A Figura 2.6 exibe o esquema do processo de expressão gênica.

2.3.2 Replicação do genoma

Uma cópia completa do genoma é feita toda vez que uma célula se divide. A replicação do DNA precisa ser altamente precisa para evitar a ocorrência de mutações. Contudo, algumas podem ocorrer devido aos erros na replicação ou aos efeitos de agentes mutagênicos químicos ou físicos que alteram diretamente a estrutura do DNA. Enzimas de reparação de DNA corrigem a maior parte dos erros, mas alguns escapam.

Se o organismo que sofreu mutação sobreviver, esses erros que escaparam do processo de correção podem se tornar características permanentes nas linhagens que descenderem deste organismo.

As mutações em conjunto com os eventos de rearranjo de genoma resultantes da recombinação de trechos do material genético permitem a evolução molecular, que dirige a evolução dos organismos vivos.

2.3.3 As diferenças entre procariotos e eucariotos

Os biólogos dividem os organismos vivos em dois grupos: procariotos e eucariotos. Os procariotos são os organismos que não possuem um núcleo celular organizado de maneira que o seu material genético fica todo espalhado dentro do citoplasma. Os eucariotos são os organismos que possuem um núcleo onde o material genético fica armazenado, de modo que ele fique separado do citoplasma.

Os procariotos possuem um genoma organizado de forma diferente dos eucariotos. Em geral, eles possuem uma única molécula de DNA, e esta molécula é circular. Além disso, alguns procariotos podem conter pequenas moléculas circulares ou lineares de DNA chamadas plasmídeos. Os genes contidos nos plasmídeos são úteis para codificar propriedades como, por exemplo, a resistência a antibióticos, contudo, os plasmídeos parecem ser dispensáveis pois um procarioto consegue viver sem a existência deles se ele estiver em um ambiente que ofereça todas as condições de vida e nenhum tipo de agressão.

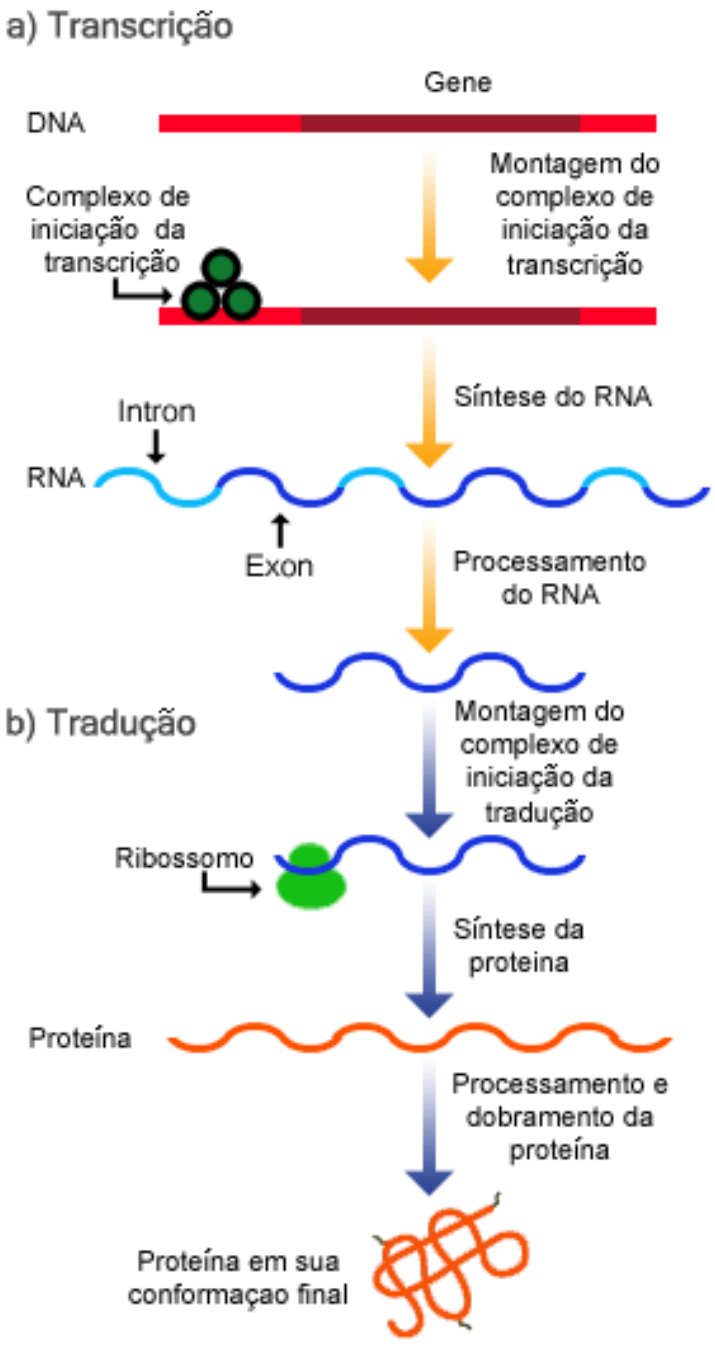

Figura 2.6: Expressão gênica: a) A transcrição é a produção do RNA a partir do gene existente no DNA. b) A tradução é a etapa que realiza a produção da proteína baseada no código genético e no RNA mensageiro produzido na transcrição.

O genoma dos eucariotos é divido em duas ou mais moléculas lineares de DNA, cada uma contida em um cromossomo. Adicionalmente, todos os eucariotos possuem um pequeno genoma mitocondrial, que é usualmente circular. As plantas e outros organismos fotossintéticos possuem um terceiro genoma localizado nos cloroplastos. A origem dos genomas destas duas organelas gerou muita especulação, mas, atualmente, a teoria aceita pela maior parte dos biólogos é a da endosimbiose. Proposta na década de 1960, esta teoria diz que a mitocôndria e o cloroplasto são relíquias de bactérias que viviam livremente e formaram uma associação simbionte com o precursor da célula eucariótica. Esta teoria se baseia na observação de que os processos da expressão gênica que ocorrem no interior destas organelas são muitas vezes equivalentes aos que acontecem em uma bactéria. Além disso, quando são feitas comparações entre seqüências de nucleotídeos, verifica-se que os genes das organelas são muito mais similares ao genes equivalentes de bactérias do que aos genes equivalentes encontrados em células eucariotas.

Uma grande variação de tamanhos pode ser encontrada entre os genomas dos eucariotos. O fungo *Saccharomyces cerevisiae* possui um genoma com tamanho de $12,1$ Mbp (Mbp - milhões de pares de bases), enquanto a planta ornamental *Fritillaria assyriaca* possui um genoma com tamanho de 120.000 Mbp. O ser humano possui um genoma com aproximadamente 3.200 Mbp. Os procariotos possuem genomas menores que giram em torno de $0,58$ Mbp (como na bactéria *Mycoplasma genitalium*) a 30 Mbp (como na bactéria *Bacillus megaterium*). Esta diferença de tamanho de genoma entre eucariotos e procariotos tem grande relação com a diferença de complexidade de genomas dos dois grupos. A Tabela 2.1 exibe uma lista com o tamanho de alguns genomas.

Os eucariotos possuem uma organização do genoma mais complexa que os procariotos. Os seus genes geralmente são localizados distantes um dos outros e a maior parte do genes é compostas por éxons separados por grandes íntrons. Além disso, no momento em que o mRNA de um gene eucarioto é sintetizado, ele sofre a adição de uma cauda poli-A, um trecho formado por dezenas de nucleotídeos do tipo Adenina, para aumento da estabilidade da molécula, que será transportada do núcleo para o citoplasma.

Os genes presentes nos genomas de organismos eucariotos costumam ocupar uma baixa porcentagem de toda a seqüência de DNA. No caso do ser humano, por exemplo, os genes ocupam somente 3% de todo o genoma nuclear. Além disso, os organismos eucariotos costumam ter a presença de um grande número de elementos repetitivos em seus genomas.

Os procariotos possuem menos genes que os genomas dos eucariotos. Além disso os seus genes costumam se localizar próximos uns dos outros e a imensa maioria dos genes não possuem íntrons. Os genes ocupam a maior parte do genoma de um procarioto. Outra característica é a baixa freqüência de seqüências repetitivas.

Estas diferenças nas características dos genomas destes dois grupos de organismos fazem com que diferentes estratégias sejam adotadas pelos projetos de seqüenciamento de

Procariotos	
Arqueobactérias	
Nanoarchaeum equitans	0,49 Mbp
Methanosarcina acetivorans	5,71 Mbp
Bactérias	
Mycoplasma genitalium	0,58 Mbp
Bacillus megaterium	30 Mbp
Eucariotos	
Fungos	
Saccharomyces cerevisiae (levedura)	12,1 Mbp
Aspergillus nidulans	25,4 Mbp
Protozoários	
Plasmodium falciparum (agente causador da malária)	23 Mbp
Tetrahymena pyriformis	190 Mbp
Invertebrados	
Caenorhabditis elegans	97 Mbp
Drosophila melanogaster (mosca da fruta)	180 Mbp
Bombyx mori	490 Mbp
Strongylocentrotus purpuratus	845 Mbp
Locusta migratoria	5.000 Mbp
Vertebrados	
Takifugu rubripes	400 Mbp
Homo sapiens (homem)	3.200 Mbp
Mus musculus (camundongo)	3.300 Mbp
Plantas	
Arabidopsis thaliana	125 Mbp
Oryza sativa (arroz)	430 Mbp
Zea mays (milho)	2.500 Mbp
Pisum sativum	4.800 Mbp
Triticum aestivum	16.000 Mbp
Fritillaria assyriaca	120.000 Mbp

Tabela 2.1: Tamanho dos genomas de algumas espécies de procariotos e eucariotos.

genoma de cada tipo de organismo.

2.4 Seqüenciamento de DNA

Os primeiros procedimentos rápidos e eficientes para seqüenciamento de DNA foram desenvolvidos no meio da década de 1970.

Um dos métodos é o da degradação química [67], que consiste na determinação da seqüência através do tratamento de uma fita dupla de DNA com produtos químicos que a cortam em posições específicas de nucleotídeos. Outro método é o da terminação de cadeia [86], que baseia-se em uma fita única de DNA cuja seqüência é determinada através da síntese de cadeias de polinucleotídeos complementares, que terminam em posições específicas de nucleotídeos. O segundo método é o mais utilizado por ser mais fácil de automatizar e também por que os produtos químicos utilizados no primeiro são muito tóxicos.

2.4.1 Método de terminação de cadeia

O princípio básico do método de terminação de cadeia é que moléculas de DNA que diferem em comprimento por apenas um nucleotídeo podem ser separadas umas das outras através da eletroforese em gel de poliacrilamida. Neste experimento, o primeiro passo é preparação de fitas únicas de DNA idênticas. Para isso, um pequeno oligonucleotídeo é ligado a uma mesma posição em cada fita de DNA disponível. A função deste oligonucleotídeo é de atuar como um primer (iniciador) para a síntese da fita complementar.

A síntese da fita complementar é catalisada pela enzima DNA polimerase e necessita da presença dos quatro tipos de nucleotídeos como substratos. Em condições normais, a síntese ocorreria até que milhares de nucleotídeos fossem polimerizados, no entanto, neste tipo de experimento também são adicionados para cada um dos 4 tipos de nucleotídeos (A, T, C, ou G) uma pequena quantidade do dideoxinucleotídeo equivalente, que é um nucleotídeo que não possui a extremidade $3'$ (OH - terminal, ver Figura 2.3) e que portanto, impede que a enzima continue a aumentar a fita de DNA complementar. Ou seja, a fita complementar cresce enquanto um dideoxinucleotídeo não é incorporado. Como um grande número de fitas é produzido, teremos fitas de diversos tamanhos, pois a incorporação do dideoxinucleotídeo é aleatória. Quatros experimentos são gerados, cada um contendo um tipo de dideoxinucleotídeo. Neste ponto, a eletroforese em gel de poliacrilamida entra em ação ao separar as fitas de diferentes tamanhos.

O gel é dividido em quatro faixas e no topo de cada faixa são colocados as fitas complementares de cada um dos 4 experimentos. Quando uma diferença de potencial é aplicada ao gel, as seqüências menores tendem a ir para direção oposta com mais facilidade

do que as maiores. Ao final do experimento, veremos no gel uma série de bandas em cada uma das faixas do gel. A menor seqüência será aquela que mais se distanciou do ponto de partida. Se esta seqüência estiver na faixa do nucleotídeo A, significa que a seqüência a ser determinada se inicia com o nucleotídeo A. A segunda banda mais distante do ponto de partida é equivalente a seqüência que possui o tamanho de um nucleotídeo a mais que a menor seqüência. Se esta banda estiver na faixa do nucleotídeo T, significa que temos até aqui a seqüência AT. Realizando esta análise sucessivamente, até que não seja mais possível distingüir as bandas, é possível determinar a seqüência complementar da fita original. A Figura 2.7a mostra um exemplo de um experimento com gel de eletroforese. Na figura, cada uma das faixas equivalem a um nucleotídeo e a seqüência de DNA pode ser determinada através da leitura das faixas de baixo para cima.

O método original de terminação de cadeia utilizava marcas radioativas para que os padrões de bandas no gel pudessem ser visto por autoradiografia. O método mais moderno utiliza marcadores fluorescentes. Cada dideoxinucleotídeo tem um marcador fluorescente associado e que pode ser reconhecido pelo detector de uma máquina de seqüenciamento. A utilização de marcadores fluorescentes foram fundamentais para a automatização do processo de seqüenciamento, pois permitem que as quatro reações com dideoxinucleotídeos possam ocorrer em um mesmo tubo, e que a leitura possa ser feita em uma única faixa de gel, já que o detector consegue diferenciar os sinais. A Figura 2.7b exibe um exemplo de um experimento com a mesma seqüência utilizada na Figura 2.7a. A Figura 2.7c mostra a representação gráfica da leitura dos sinais fluorescentes emitidos pelas bandas e captados pela máquina de seqüenciamento.

2.4.2 Clonagem

A clonagem é um procedimento que permite que cópias idênticas de uma molécula de DNA sejam produzidas. Como a técnica de terminação de cadeia necessita de uma grande quantidade de fitas únicas de DNA para funcionar, a clonagem é utilizada para amplificar a quantidade de fitas disponíveis. Existem diversas técnicas de clonagem e algumas delas serão explicadas a seguir.

Clonagem por vetor

Um vetor é uma molécula de DNA que possui a habilidade de se replicar dentro de uma célula hospedeira e pode ser usada para a clonagem de outros fragmentos de DNA. O método de clonagem por vetor mais comum é o que utiliza um plasmídeo como vetor.

Todo plasmídeo possui um local que é reconhecido como origem de replicação pelas enzimas e proteínas que realizam a replicação do DNA. Isto é uma característica que

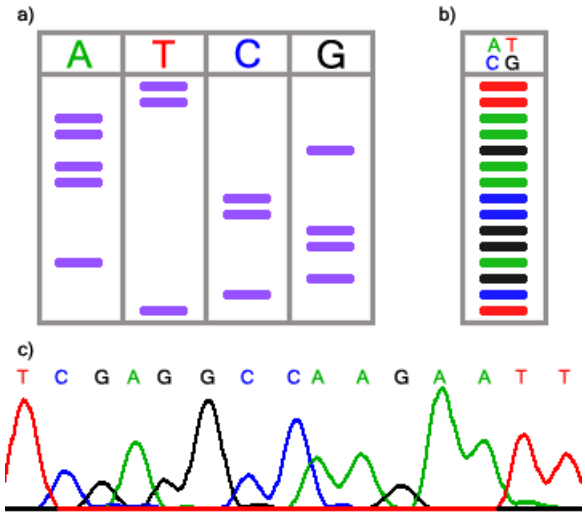

Figura 2.7: a) Gel de eletroforese feito com um segmento de DNA que contém a seqüência TCGAGGCCAAGAATT. b) Experimento de eletroforese que utiliza marcadores fluorescentes, realizado com o mesmo segmento de DNA. c) Representação gráfica da leitura dos sinais emitidos pelos marcadores fluorescentes, captados pela máquina de seqüenciamento.

permite que os plasmídeos sejam replicados pela bactéria e propagados para as células descendentes nos eventos de divisão celular.

O método consiste na inserção do trecho de DNA que se deseja replicar em um ponto específico do plasmídeo, que será inserido dentro de uma bactéria. Uma vez dentro da célula bacteriana, o plasmídeo será duplicado e cada cópia ficará com um descendente produzido pela divisão celular. Como as bactérias possuem um ciclo de vida muito curto e se reproduzem em grande velocidade, em pouco tempo é possível obter uma grande quantidade de cópias do DNA desejado.

Esta metodologia tem a vantagem de produzir fitas duplas, que ao serem separadas fornecem a seqüência de DNA em ambas as direções. No entanto, ela apresenta dificuldades na geração de plasmídeos sem contaminação por DNA ou RNA bacterial.

Clonagem por bacteriófagos

Outro técnica de clonagem de seqüências é a utilização de bacteriofagos M13. Os bacteriófagos M13 são vírus que possuem um genoma formado por uma única fita de DNA e que atacam bactérias.

Nesta técnica, o genoma do vírus realiza o papel de vetor, desempenhado pelo plasmídeo na metodologia de clonagem explicada anteriormente. O DNA alvo do seqüenciamento

é inserido no genoma do vírus, que ataca a célula bacteriana inserindo dentro dela o seu material genético. A bactéria replica o DNA viral, junto com o DNA alvo.

A desvantagem desta técnica é que ela não comporta seqüências maiores que 3 kbp.

Clonagem por PCR

A técnica PCR (*Polimerase Chain Reaction*) também pode ser utilizada para clonagem de seqüências. Ela consiste na utilização de primers que se ligam em posições específicas da fita de DNA e de uma enzima DNA polimerase que trabalha conforme a temperatura a que é imposta.

O experimento se inicia com o aquecimento da solução para que a fita dupla de DNA se separe em duas fitas. Com a diminuição da temperatura, os primers se ligam nas fitas simples e servem como ponto de início para que a enzima trabalhe realizando a polimerização das novas fitas. Após um certo tempo, a solução é aquecida novamente. A enzima pára de trabalhar e as fitas se soltam. A temperatura é diminuída e a polimerização se inicia novamente com o novo conjunto de fitas. Isso é repetido várias vezes, de modo que ao final sejam obtidas um grande número de cópias da fita original.

Neste tipo de experimento, dois primers são utilizados, um para cada direção da fita. Para produção de uma única fita, um dos primers pode receber um grupo metálico que permita que ele seja separado por magnetização.

2.5 Estratégias de seqüenciamento

Independentemente da técnica de clonagem ou de seqüenciamento utilizada, as seqüências produzidas pelos experimentos são muito pequenas se comparadas ao genoma de um organismo. Em geral as seqüências possuem algumas centenas de bases, enquanto um genoma pode ter vários milhões de bases. Por isso, um projeto de seqüenciamento trabalha de forma a obter as seqüências pequenas e realizar a montagem destas seqüências para conseguir a seqüência genômica completa. A montagem é a união de trechos de seqüências segundo critérios como, por exemplo, as sobreposições que os trechos possuem, para obtenção de uma seqüência maior.

2.5.1 Shotgun

Uma estratégia utilizada para a obtenção das seqüências e posterior montagem é o shotgun. Este método consiste na fragmentação de várias cópias do DNA genômico através da sonicação, que é o emprego de ondas sonoras de alta freqüência para cortar moléculas de DNA. Os fragmentos são então separados para obtenção de trechos que tenham tamanhos

dentro de um intervalo pré-determinado. Os pedaços selecionados são então clonados para aumento da redundância, que é necessária para que o DNA genômico seja todo coberto por eles. Após a clonagem, o seqüenciamento dos fragmentos é feito e a etapa seguinte é o processamento destas seqüências por computadores que operam de forma a unir os pedaços que se sobrepõem. O próximo passo é a obtenção dos trechos referentes ao buracos que não foram seqüenciados, para que o genoma possa ser completado.

O shotgun é uma técnica muito utilizada para o seqüenciamento rápido de genomas microbiais. Ela tem a vantagem de não necessitar de um mapeamento genético ou físico do genoma. A grande desvantagem é a complexidade da análise de dados do processo de união das seqüências. Este processo requer máquinas com um grande poder computacional e as seqüências repetitivas que podem ser encontradas em genomas podem provocar erros no processo. A técnica também é utilizada em seqüenciamento de genomas de organismos complexos. A empresa Celera Genomics [24] realizou parte do seqüenciamento do genoma humano com a utilização de shotgun [111].

2.5.2 Clone Contig

Um *Clone Contig* é uma coleção de clones que se sobrepõem. O seqüenciamento através de clone contig é a técnica tradicional usada para a obtenção da seqüência genômica de organismos eucariotos, mas também é utilizada com micróbios que foram previamente mapeados genética ou fisicamente.

A estratégia consiste na quebra das moléculas de DNA que serão seqüenciadas em segmentos, cada um com algumas centenas de kbp ou alguns Mbp de comprimento. Estes segmentos são replicados com a utilização de cosmídeos, vetores de clonagem de alta capacidade.

O clone contig é construído com base nas informações de sobreposições dos segmentos. O seqüenciamento dos segmentos pode ser feito com a utilização da estratégia shotgun.

Idealmente, os fragmentos clonados são ancorados de acordo com o mapa genético e/ou físico de modo que os dados obtidos do seqüenciamento possam ser verificados e interpretados através da observação de características como, por exemplo, a existência de genes em uma particular região.

A obtenção do genoma completo de organismo complexos como, por exemplo, o homem ou o milho é uma tarefa extremamente difícil. Por isso, outras técnicas são utilizadas para obtenção de informações do genoma sem a necessidade de seu seqüenciamento completo. Um exemplo é o seqüenciamento de ESTs, que visa obter as seqüências dos genes expressos pelo organismo e será explicado na Seção 2.7.

2.6 Projetos de Seqüenciamento de Genoma

Os projetos de seqüenciamento completo de genomas, conhecidos como Projetos Genomas, têm como principal objetivo a descoberta de toda a seqüência genômica de um organismo. Diversos projetos Genomas já foram e estão sendo realizados. Graças a esses projetos, hoje conhecemos o genoma de diversas espécies, incluindo o do ser humano.

A conclusão do seqüenciamento do genoma da bactéria *Xyllela fastidiosa* em 1999 [92, 120], em particular, colocou o Brasil em posição de destaque mundial, pois foi o primeiro país do mundo a concluir a montagem do genoma de um fitopatógeno, organismo causador de doenças em plantas.

Na página Entrez Genome [34] mantida pelo NCBI [73] é possível encontrar a listagem de todos os projetos de seqüenciamento de genoma que foram concluídos ou que estão em andamento. A cada dia, novas informações podem ser encontradas nesta página, ilustrando a evolução que os diversos projetos estão mostrando ao longo do tempo. Por exemplo, no início da escrita deste texto, a versão de 06 de Setembro de 2004 apresentava uma lista de genomas completos compostos por 1.358 vírus, 19 arqueobactérias, 176 bactérias e de diversos eucariotos como o homem e o rato. Já na versão de 15 de Agosto de 2006, próxima da conclusão da escrita do texto desta tese, o site exibia uma lista de 1.668 vírus, 28 arqueobactérias, 357 bactérias e 22 eucariotos com genomas completos.

Mas qual a utilidade da descoberta da seqüência genômica de um organismo? De posse da seqüência genômica os cientistas podem determinar quais são os genes existentes no organismo, o que cada gene produz, quais genes são relacionados a uma determinada característica boa ou ruim. Em um projeto genoma, a seqüência genômica é obtida através de experimentos e analisada em busca de regiões que apontem o início dos genes. Além disso, no caso de organismos eucariotos, uma análise extra precisa ser feita para identificação dos éxons e íntrons dos genes.

O estudo da seqüência genômica permite também que mutações que ocorreram ao longo da evolução do organismo possam ser analisadas para que informações filogenéticas ou relacionadas com doenças possam ser obtidas.

Curas ou mecanismos de prevenção de doenças poderão ser descobertas com a análise dos genes que estão relacionados com elas. Por exemplo, no caso da *Xylella fastidiosa* a obtenção de seu genoma poderá auxiliar na prevenção e na cura da doença do amarelinho que ela provoca na laranja, uma planta economicamente importante para o Brasil.

Estas são apenas algumas das muitas atividades que podem ser derivadas do estudo do genoma de um ser vivo.

2.7 Projetos de Seqüenciamento de ESTs

Os projetos de seqüenciamento de ESTs (*Expressed Sequence Tag*) [2] são realizados com o objetivo de rapidamente obter uma boa aproximação do índice gênico de um organismo, que é a listagem dos genes existentes em seu genoma.

A estratégia adotada por esta técnica é a de realizar o seqüenciamento de segmentos de cDNA (DNA complementar), que é uma fita de DNA produzida a partir do complemento do mRNA com a utilização da enzima transcriptase reversa.

Como já dito anteriormente, o mRNA é a molécula de RNA produzida pela célula, a partir da transcrição do gene contido no DNA, e que será utilizada para produção de proteínas na fase de tradução. Assim, o cDNA nada mais é que a seqüência de nucleotídeos de um gene existente no genoma do organismo.

O processo de seqüenciamento com a utilização de ESTs envolve a produção de bibliotecas de cDNA, a clonagem dos cDNAs com a utilização vetores (em geral bactérias), e o seqüenciamento dos clones através de uma única leitura em uma máquina de seqüenciamento, que torna esta técnica de baixo custo, em relação às outras técnicas existentes.

Ao realizar o seqüenciamento de cDNAs, os projetos ESTs produzem dados sobre os genes que estão sendo expressos no organismo de maneira imediata.

Os genes de um organismos não são expressos com igual freqüência. Existem genes que são expressos a toda hora pois produzem proteínas ligadas às vias metabólicas que regem as reações químicas que ocorrem no organismo, e existem genes que são expressos apenas quando o organismo é submetido à condições especiais, e, além disso, tecidos diferentes expressam genes diferentes. Em função destas características, este tipo de seqüenciamento necessita que sejam produzidas bibliotecas de cDNA com origem em diversos tecidos, extraídos sob diferentes condições, tais como, idade, ambiente, presença de doenças, etc. Técnicas como subtração [16] e normalização [96] são utilizadas para tentar equilibrar o seqüenciamento de forma que os genes mais expressos não sejam seqüenciados muitas vezes e que os raros possam ser seqüenciados em maior número.

Assim como os projetos de seqüenciamento de genomas, os projetos ESTs são cada vez mais difundidos, produzindo uma grande quantidade de informação. Por exemplo, no início da escrita do texto desta tese, o banco dbEST [29] mantido pelo NCBI, continha em sua versão 090304, de 3 de Setembro de 2004, 23.416.084 seqüências públicas de ESTs de 741 organismos diferentes. A versão 081106, de 11 de Agosto de 2006, apresentava 38.056.628 seqüências públicas de ESTs de 1.179 organismos diferentes. As Tabelas 2.2 e 2.3 exibem a lista dos 20 organismos com maior quantidade de ESTs depositados no dbEST nas versões 090304 e 081106, respectivamente.

Organismo	Número de ESTs
Homo sapiens (homem)	5.679.423
Mus musculus + domesticus (camundongo)	4.246.846
Ciona intestinalis (organismo urocordado)	684.280
Rattus sp. (rato)	683.238
Danio rerio (peixe paulistinha – zebrafish)	575.250
Triticum aestivum (trigo)	561.713
Gallus gallus (galinha)	495.092
Bos taurus (touro)	493.329
Xenopus laevis (sapo)	432.424
Xenopus tropicalis (sapo)	423.107
Zea mays (milho)	416.090
Drosophila melanogaster (mosca da fruta)	382.439
Hordeum vulgare + subsp. vulgare (cevada)	367.798
Sus scrofa (porco)	358.930
Glycine max (soja)	342.359
Arabidopsis thaliana (planta da família Brassicaceae)	322.651
Caenorhabditis elegans (organismo nematóide)	302.074
Oryza sativa (arroz)	284.057
Saccharum officinarum (cana-de-açúcar)	246.301
Sorghum bicolor (sorgo)	190.949

Tabela 2.2: Lista dos 20 organismos com maior número de seqüências no dbEST (versão 090304 de 3 de Setembro de 2004).

Organismo	Número de ESTs
Homo sapiens (homem)	7.887.827
Mus musculus + domesticus (camundongo)	4.719.943
Oryza sativa (arroz)	1.186.580
Danio rerio (peixe paulistinha – zebra fish)	1.091.817
Bos taurus (touro)	1.077.784
Xenopus tropicalis (sapo)	1.044.182
Zea mays (milho)	879.619
Rattus norvegicus + sp. (rato)	871.148
Triticum aestivum (trigo)	855.066
Ciona intestinalis (organismo urocordado)	686.396
Arabidopsis thaliana (planta da família Brassicaceae)	622.972
Gallus gallus (galinha)	599.140
Sus scrofa (porco)	584.507
Xenopus laevis (sapo)	537.424
Drosophila melanogaster (mosca da fruta)	498.214
Hordeum vulgare + subsp. vulgare (cevada)	437.321
Canis familiaris (cachorro)	365.946
Glycine max (soja)	358.905
Caenorhabditis elegans (organismo nematóide)	346.064
Pinus taeda (pinho amarelo)	329.469

Tabela 2.3: Lista dos 20 organismos com maior número de seqüências no dbEST (versão 081106 de 11 de Agosto de 2006).

2.7.1 Problemas encontrados nos Projetos ESTs

Os projetos baseados em EST também possuem problemas. Devido ao fato do seqüenciamento ser feito em apenas uma leitura, os ESTs possuem taxas de erros altas. Além disso, por causa da limitação da técnica, apenas as pontas $3'$ e $5'$ são seqüenciadas no método padrão e, normalmente, não conseguem cobrir todo o gene por possuírem apenas algumas centenas de bases de comprimento. Uma técnica de seqüenciamento denominada ORESTES [22] pode ser utilizada como complemento à técnica EST pois produz seqüências que tendem a se concentrar na parte central do gene.

Além dos erros de seqüenciamento, as seqüências sofrem diferentes tipos de contaminação, dependendo, em parte, de qual dos muitos protocolos foi utilizado na construção das bibliotecas de cDNA. Um problema típico é a inclusão de seqüências de vetores, utilizados no seqüenciamento, ou de sítios de restrição no final das seqüências. A contaminação por vetor também pode ocorrer devido aos eventos de rearranjo de DNA, dentro da bactéria hospedeira, que causam a inserção de seqüência bacterial no meio do EST.

As seqüências também podem ser contaminadas por outros organimos, como os vírus. Estes contaminantes aparecem, em geral, devido à contaminação do laboratório ou a infecção do tecido que originou a biblioteca.

Contaminações causadas por DNA genômico do próprio organismo fazem com que íntrons apareçam como regiões expressas, levando a predição de traduções alternativas inexistentes. Contaminação por DNA intergênico pode resultar na predição de falsos genes.

Outra forma comum de contaminação em ESTs é a contaminação por mRNA prematuro, que é o mRNA que não passou pelo processo de remoção de trechos que possuem origem em regiões de íntrons. Apesar de ESTs representando contaminação por pré-mRNA possam parecer retenção de íntrons, eles são muito mais artefatos do que seqüências exônicas reais.

Seqüências quiméricas são outro problema. Uma quimera é a concatenação de duas ou mais seqüências expressas de diferentes áreas e pode induzir a erros durante a fase de análise das seqüências em busca de genes.

2.8 Projetos de Seqüenciamento no Brasil

No Brasil existem diversos grupos de pesquisa realizando projetos de seqüenciamento completo ou ESTs. Os resultados produzidos por estes grupos colocaram o país em posição de destaque no cenário científico internacional.

Os principais projetos realizados no Brasil recebem apoio financeiro da FAPESP, do MCT e do CNPq, e serão citados a seguir.

2.8.1 FAPESP

A FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo [36] em parceria com outras intituições financia uma série de projetos. Em 1997 ela organizou a rede ONSA (Organization for Nucleotide Sequencing and Analisys), um instituto virtual de genômica formado inicialmente por 30 laboratórios ligados a instituições de pesquisa do Estado de São Paulo. A FAPESP participa de diversos projetos Genomas e de seqüenciamento de ESTs e alguns deles são listados abaixo:

- *Xyllela fastidiosa* - Primeiro projeto realizado por esta rede, foi anunciado em 1997 e teve início de suas atividades em 1998. O seqüenciamento e a anotação foram concluídos em 2000 e resultaram na publicação de um artigo que foi capa da revista Nature, por ser o primeiro organismo fitopatógeno a ser seqüenciado [92].

- Genoma Cana-de-Açúcar (SUCEST) - Segundo projeto da rede, teve início em 1999 e a sua etapa de seqüenciamento foi concluída em 2000 [100, 103, 113].

- *Xanthomonas citri* e *Xanthomonas campestri* - Duas bactérias fitopatógenas que tiveram o seqüenciamento concluído em 2002 [119].

- Projeto Genoma Humano do Câncer - Projeto EST realizado internacionalmente e que tem como objetivo a descoberta de genes que possuem ligação com os diversos tipos de câncer existentes. Este projeto teve início em 1999 e em seu primeiro ano identificou mais de 1 milhão de seqüências de genes de tumores mais freqüentes no Brasil. A parcela brasileira deste projeto é formada por diversos grupos, incluindo o Instituto Ludwig para Pesquisa do Câncer [105].

- Projeto FORESTs - Anunciado em 2001 este projeto visa o seqüenciamento de ESTs de eucalipto com o objetivo de descobrir possibilidades de melhora na matéria-prima utilizada no país para a fabricação de papel [37].

- Projeto *Schistossoma mansoni* - Projeto EST de seqüenciamento do organismo causador da esquistossomose, foi iniciado em 2001 e em 2002 concluiu a identificação de 200 novos genes ligados aos diferentes estágios de vida do parasita [89].

- Projeto Genomas Agronômicos e Ambientais - Projeto multi-genômico de seqüenciamento completo ou EST de organismos ligados ao ambiente e à agronomia. Este projeto começou em 2000 com o estudo de uma variedade de *Xylella* que ataca videiras. Em 2001 este projeto conclui o mapa genético da bactéria *Leifsonia xyli*. Este projeto também inclui o Projeto Genoma do Café e do organismo *Leptospira interrogans* [38].

- Projeto Genoma Funcional - Este projeto envolve diferentes sub-projetos que possuem um mesmo objetivo comum: avaliar a reposta de determinados organismos diante de variações produzidas no meio ambiente. Isso é feito através do estudo de como os genes dos organismos são expressos diante de pertubações provocadas no ambiente em que vivem.

2.8.2 MCT e CNPq

O Ministério da Ciência e Tecnologia (MCT) [69] e o Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) [27] financiam uma série de projetos em todo o país.

Genoma Nacional

A Rede Genoma Nacional [18] ou Genoma Brasileiro foi lançada em 2000 pelo MCT e pelo CNPq com a participação de 25 laboratórios de biologia molecular e um centro de Bioinformática distribuídos em todas as regiões geográficas do país.

Esta rede concluiu em 2003 o seqüenciamento e a anotação do genoma da bactéria *Chromobacterium violaceum* [31].

O segundo organismo seqüenciado e anotado foi a bactéria *Mycoplasma synoviae*.

Redes Regionais

Além da Rede Genoma Nacional o MCT e o CNPq promovem a implantação de diversas redes gênomicas, espalhadas por todas as regiões do país e que realizam vários projetos:

- Rede Genoma do Estado de Minas - Coordenada pela Fundação de Amparo à Pesquisa de Minas Gerais (FAPEMIG), realiza um Projeto EST do parasita *Schistosoma mansoni*, causador da esquistosomose [42].

- Programa Genoma Nordeste (ProGeNe) - Coordenado pela Universidade Federal de Pernambuco (UFPE), realiza o seqüenciamento de ESTs da *Leishmania chagasi*, uma das três espécies causadoras da leishmaniose visceral [81].

- Rede Genoma Centro-Oeste - Projeto EST para estudo do genoma do fungo *Paracoccidioides brasiliensis*, causador de um tipo de micose de alta prevalência na América Latina. É coordenada pela Universidade de Brasília (UnB) [76].

- Rede Genoma do Consórcio do Instituto de Biologia Molecular do Paraná, FIOCRUZ e Universidade de Mogi das Cruzes - Sob coordenação do Instituto de Biologia Molecular do Paraná (IBMP), destina-se a seqüenciar ESTs para o estudo dos

genes envolvidos no processo de diferenciação celular do *Trypanossoma cruzi*, além da análise de novos alvos quimeoterápicos [106].

- Rede Genômica do Estado da Bahia e São Paulo - Coordenada pela Unicamp, desenvolve o projeto EST de seqüenciamento do fungo *Crinipellis perniciosa*, causador da doença da vassoura de bruxa que ataca o cacaueiro [61].

- Rede Genoma do Rio de Janeiro (RioGene) - Projeto para seqüenciamento completo da bactéria fixadora de nitrogênio *Gluconacetobacter diazotrophicus*, coordenado pela Universidade Federal do Rio de Janeiro (UFRJ) [85].

- Programa Genoma do Estado do Paraná (GenoPar) - Projeto de seqüenciamento completo para estudo do genoma funcional e estrutural da bactéria fixadora de nitrogênio *Herbaspirillum seropedicae*, coordenado pela Universidade Federal do Paraná (UFPR) [43].

- Rede Sul de Análise de Genomas e Biologia Estrutural (PROGENESUL) - Projeto de seqüenciamento do genoma da bactéria *Mycoplasma hyopneumoniae*, causadora da pneumonia micoplásmica em suínos. A rede é coordenada pela Fundação de Amparo à Pesquisa do Rio Grande do Sul (FAPERGS) [82].

- Rede da Amazônia Legal de Pesquisas Genômicas (REALGENE) - Projeto EST de seqüenciamento da planta *Paullinia culpana*, o guaraná. A Rede é coordenada pela Universidade Federal do Amazonas (UFAM) [84].

O MCT financia ainda o Projeto Genolyptus [41] que é uma parceria público-privada para o seqüenciamento EST do eucalipto executado pela Rede Brasileira de Pesquisa do Genoma do Eucalipto, formada por 12 empresas, sete universidades e pela Empresa Brasileira de Pesquisa Agropecuária (Embrapa).

2.9 Bioinformática

A resolução dos problemas apresentados pelos projetos de seqüenciamento é uma importante área de atuação da Bioinformática. A Bioinformática é a área da Computação destinada a desenvolver ferramentas para análise de dados e resolução de problemas em aplicações biológicas.

A evolução que ocorreu na capacidade de processamento dos computadores permitiu que a Bioinformática desenvolvesse softwares para lidar com o imenso volume de dados produzidos pelos diversos projetos na área de Biologia, especialmente os projetos de seqüenciamento completos ou ESTs. Tais projetos produzem inúmeras seqüências que

precisam ser processadas de forma automática e com a menor taxa de erros possível. A automatização é extremamente importante pois o trabalho de processamento e análise dos dados de forma manual é inviável.

2.10 Bioinformática para Projetos ESTs

Entre as diversas atividades desenvolvidas pela Bioinformática na análise de dados de projetos de seqüenciamento de ESTs, podemos citar a detecção e remoção de artefatos, a verificação de contaminação e a clusterização.

2.10.1 Detecção e remoção de artefafos

Artefatos são trechos que podem ser encontrados na seqüências mas que possuem baixa qualidade ou baixa complexidade ou que não pertencem ao organismo que está sendo estudado pelo projeto. Essas regiões, portanto, devem ser identificadas e removidas para que não prejudiquem a análise dos dados produzidos.

Este processo de detecção e remoção de artefatos, também conhecido como "trimagem", é composto por diversas etapas que serão discutidas adiante. Antes, apresentaremos a origem dos diferentes tipos de artefatos que podem ser encontrados e os problemas que eles podem ocasionar nas análises.

As origens dos artefatos

O processo de obtenção da seqüência de nucleotídeos de um segmento de DNA envolve a realização de uma série de experimentos biológicos. O DNA alvo precisa ser replicado para que haja quantidade de material suficiente para processá-lo e isto é normalmente feito com a utilização de vetores como, por exemplo, os plasmídeos. O DNA alvo é inserido em um local específico na molécula de DNA do vetor.

Devido ao fato da técnica de seqüenciamento de ESTs produzir trechos nas extremidades dos genes, é normal a presença de trechos do vetor nas seqüências lidas pela máquina de seqüenciamento. Obviamente, este artefato não é interessante para o objetivo do projeto, que é a determinação dos genes expressos no organismo.

Seqüências de baixa complexidade são consideradas artefatos porque não fornecem informações relevantes. As seqüências de poli-A e poli-T são exemplos de seqüências deste tipo. No momento da produção do mRNA, nos organismos eucariotos, uma cauda poli-A é ligada ao seu final. O mRNA é utilizado para produção do cDNA, que será seqüenciado. Conforme a direção do seqüenciamento, isto é, conforme a fita do cDNA que for seqüenciada, trechos de poli-A ou poli-T podem aparecer.

Alguns tipos de vetores utilizados para clonagem necessitam da utilização de um pequeno segmento de DNA chamado adaptador para que a inserção do DNA alvo possa ser feita. Esta seqüência que não pertence nem ao vetor e nem ao organismo estudado é um artefato que deve ser identificado e removido.

Relacionados ao processo de leitura realizado pelas máquina de seqüenciamento temos os artefatos de baixa qualidade. O valor de qualidade de uma base indica a probabilidade dela estar correta. Quanto menor o valor de qualidade, maior a probabilidade de erro. A máquina de seqüenciamento percorre o gel do experimento lendo os sinais emitidos pelos marcadores fluorescentes dos nucleotídeos. A precisão da leitura depende da intensidade do sinal, que tende a ser mais fraca nas extremidades do gel e mais forte na porção central. Como produto da leitura, a máquina produz um arquivo chamado cromatograma ou eletroferograma. Este arquivo é lido por programas de base calling como, por exemplo, o phred [44], que operam de forma a determinar as bases lidas pela máquina.

Os sinais fracos emitidos por algumas bases podem gerar erros, mas os sinais fortes também podem acarretar em erros. Quando a máquina de seqüenciamento encontra uma região com picos de sinais muito altos, o fenômeno de derrapagem pode acontecer. Este fenômeno se caracteriza pela repetição de bases de maneira anormal. Devido aos fortes sinais, a máquina pode interpretar a existência de mais de uma base, onde na verdade só existe uma, como se a seqüência estivesse borrada.

A origem da derrapagem está geralmente associada à presença de longas cadeias de As ou Ts que apresentam problemas durante as reações de seqüenciamento de DNA [6]. As longas cadeias de As ou Ts podem não parear corretamente durante a reação de polimerização e isso gera fragmentos homopoliméricos de diferentes tamanhos que são seguidos pela mesma seqüência de bases.

Problemas causados pelos artefatos

A presença de artefatos nas seqüências podem influenciar negativamente os resultados das análises dos dados produzidos pelo projeto.

As seqüências de baixa qualidade são seqüências que possuem taxas de erros muito altas. A manuntenção de seqüências deste tipo seria uma atitude imprudente, pois não pode se dizer com uma boa margem de segurança que um trecho de baixa qualidade realmente represente a seqüência determinada pelo programa de base calling.

Em um processo de clusterização, as seqüências de vetores e as seqüências de baixa complexidade podem forçar a criação de clusters através do agrupamento errôneo de seqüências por causa da adição de similaridade não relevante ao processo [104]. Como o critério utilizado na montagem dos clusters é a sobreposição das seqüências, os trechos de baixa complexidade poderiam gerar sobreposições válidas no critério do software de clusterização, mas que na realidade não existem.

Os artefatos que representam vetores, adaptadores e derrapagens são seqüências que não pertencem ao organismo alvo do projeto. A presença destas seqüências podem ocasionar erros na identificação de genes.

Para a eliminação de possíveis problemas, os projetos de seqüenciamento operam de forma a remover estas seqüências utilizando diversas técnicas.

Técnicas de detecção e remoção de artefatos

Cada tipo de artefato possui formas de detecção e remoção apropriadas que discutiremos a seguir. Normalmente, após a aplicação destas técnicas o tamanho da seqüência restante é verificado. Se ela tiver tamanho menor que um certo valor mínimo, a seqüência é descartada de futuras análises.

Remoção de artefatos de baixa qualidade A remoção de trechos de baixa qualidade pode ser atacada de várias maneiras, que podem ser simples ou mais elaboradas. As soluções mais simples são obviamente as mais rápidas, um fator importante quando o volume de dados a ser processado é muito grande. Assim, a decisão da estratégia a ser utilizada depende do tempo que se deseja gastar com esta tarefa.

O valor de qualidade de uma base determinada por um programa de base calling como o phred ou o TraceTuner [108] é baseado na probabilidade de erro que essa base possui e é dada pela fórmula $Q = -10 \times \log_{10}(\textit{probabilidade de erro})$ [26]. Assim, quanto maior a probabilidade de erro, menor a qualidade.

Uma estratégia simples para a remoção de baixa qualidade é a utilização de um algoritmo para determinação da subseqüência máxima [66, Seção 5.8]. A seqüência determinada por esse algoritmo seria a seqüência com as pontas de baixa qualidade removidas. O próprio programa phred possui um parâmetro que faz com que ele indique qual é essa subseqüência, em uma seqüência que ele acabou de determinar. O algoritmo implementado pelo phred converte o valor de qualidades em probabilidades de erros e tenta minimizar a probabilidade de erro da subseqüência. Antes de executar o algoritmo cada base tem sua probabilidade de erro subtraída de $0,05$. Este valor equivale ao valor de qualidade 13, a mínima aceitável segundo esta implementação do algoritmo.

Muitos projetos utilizam a análise através de janelas deslizantes. Em geral, a seqüência é percorrida base a base, nas duas direções a partir das extremidades, por uma janela de um determinado tamanho em busca de trechos que possuam um número máximo de bases com qualidades menores que a mínima. No trabalho desenvolvido por Telles e da Silva [104], por exemplo, utilizou-se uma janela de tamanho 20, que devia ter no máximo 12 bases com qualidade abaixo de 10.

Alguns projetos de seqüenciamento utilizam programas específicos para o processo de detecção e remoção como, por exemplo o ESTprep [88].

Este programa faz a análise de qualidade em duas etapas. Na primeira etapa, o programa verifica se o trecho inicial de 20 bases da seqüência possui menos que 8 bases com qualidade maior que 20, caso em que será removido. Nesta mesma etapa, o software verifica se a qualidade média das 200 primeiras bases é menor que 20, caso em que toda a seqüência é descartada.

Na segunda etapa, após a remoção de outros tipos de artefatos, o ESTprep utiliza uma janela deslizante de 20 bases para identificar a primeira região com no máximo 8 bases com qualidade menor que 10 para determinação do ponto de corte na extremidade $3'$, que será o início da região encontrada.

Outro software de processamento de seqüências para identificação e remoção de artefatos é o LUCY [26], que é utilizado pelo TIGR - The Institute of Genomic Research. Ele possui uma estratégia de análise mais complexa.

Como o início e o final da seqüência são em geral de baixa qualidade, o LUCY age de forma a identificar estes trechos primeiro. A partir da ponta esquerda da seqüência uma janela de tamanho 10 percorrerá a seqüência até encontrar um trecho que tenha uma probabilidade de erro menor ou igual a 2%. O mesmo será feito na ponta direita da seqüência. Estes trechos identificados são removidos e o que sobrar passará pelo processo restante. Se a seqüência inteira não passar no teste, ela será inteiramente descartada.

O próximo passo é a identificação de trechos que possuem taxas de erros altas para que possam ser eliminados. Neste passo, duas janelas são utilizadas. A primeira tem tamanho 50 e um valor limite de probabilidade de erro igual a 8% e elimina os trechos grandes de baixa qualidade. A segunda tem tamanho 10 e um valor limite de probabilidade de erro igual a 30% e elimina os trechos pequenos que não são removidos pela primeira.

A primeira janela percorre a seqüência resultante do primeiro passo de limpeza. A partir do início desta seqüência, o programa calcula para a janela o valor médio de probabilidade de erro. Se o valor estiver dentro do valor limite a janela será adicionada à seqüência candidata, que continuará crescendo enquanto as janelas que percorrerem a seqüência estiverem com o valor dentro do limite. Se o valor estiver fora do limite, a seqüência candidata será terminada e separada para o próximo passo. A janela continuará a percorrer a seqüência até o final, se uma nova janela voltar a ter o valor dentro do limite, uma nova seqüência candidata será iniciada.

Cada seqüência candidata será percorrida pela segunda janela seguindo o mesmo critério. Após este processo, todas as seqüências candidatas com tamanho menor que o mínimo serão descartadas. Dentre as seqüências restantes, aquela que tiver uma probabilidade de erro geral menor ou igual que $2,5\%$ e probabilidades de erros nas extremidades menores que 2% será a seqüência final. A probabilidade de erros nas extremidades é avaliada com as duas últimas bases de cada ponta. No caso raro de mais de uma seqüência atender ao critério, a maior será mantida.

Remoção de vetores e adaptadores Uma maneira de se realizar a remoção de vetores e adaptadores é utilizando programas como o cross_match ou o swat [44].

Estes programas são utilizados para alinhamento da seqüência analisada com as seqüências dos vetores e adaptadores. O cross_match, por exemplo, tem a opção de realizar mascaramento (substituição das letras das bases por Xs) das regiões que apresentarem alinhamento, permitindo uma rápida análise das regiões que devem ser removidas.

O programa LUCY também realiza a remoção de vetores e adaptadores. Para este serviço, ele necessita das seqüências dos trechos do vetor e do adaptor no ponto onde o inserto é fixado (splice sites upstream e downstream, ou seja, as regiões vizinhas ao ponto onde a seqüência do vetor foi cortada para inserção da seqüência a ser replicada).

Como os artefatos relacionados aos vetores costumam se localizar no início da seqüência onde a qualidade das bases é geralmente baixa, uma comparação simples que busca pelo alinhamento mais longo pode não encontrar todos os trechos de vetor devido aos erros de base-calling. Assim, o software realiza uma busca adaptativa pelos valores médios de qualidades das bases. Nas regiões de baixa qualidade o programa permite que pequenos trechos de vetor sejam identificados enquanto em regiões de melhor qualidade apenas trechos maiores são identificados. Devido às diferentes regiões de qualidades existentes no início da seqüência o software considera três critérios diferentes. A busca é feita em áreas de 40, 60 e 100 bases com comprimentos mínimos de alinhamento de 8, 12 e 16 bases. Um alinhamento local ótimo dentro de cada área deve ter, pelo menos, o comprimento mínimo para ser considerado vetor. Estas janelas são colocadas no início da seqüência original para evitar que fragmentos de vetores sejam perdidos em seqüências que possuem um trecho de baixa qualidade muito longo no início.

O splice site upstream será procurado nas primeiras 200 bases. LUCY procurará pelo maior alinhamento com pelo menos três bases corretas para cada base incompatível, o que não significa que haverá 25% de erro porque apenas o alinhamento com maior pontuação local será utilizado. Alinhamentos menores à esquerda podem ser ignorados. Se ainda existirem bons alinhamentos após o melhor, o programa continuará a busca até que todos os fragmentos sejam identificados. Depois de terminar a busca pelo splice site upstream, o dowstream é também procurado, pois a seqüência pode conter um inserto pequeno. O splice site downstream é procurado utilizando-se o critério de alinhamento mínimo de 16 bases.

Remoção de poli-A e poli-T A buscas por caudas poli-A e poli-T também variam em complexidade. O programa LUCY, por exemplo, utiliza um esquema simples. Ele realiza a busca por caudas poli-A/T utilizando uma janela de 50 bases para identificação de trechos que possuam no mínimo 10 bases Ts ou As. Nesta busca, são permitidos no máximo três bases incompatíveis entre cada trecho de 10 Ts ou As.

O procedimento desenvolvido por Telles e da Silva realiza a remoção destes artefatos em diversas etapas. Algumas são efetuadas juntamente com a remoção de vetores e outras logo após. Estas diferentes etapas foram desenvolvidas com o objetivo de se detectar várias possibilidades de ocorrência destes tipos de artefatos. Elas utilizam o programa swat para realizar o alinhamento da seqüência analisada com seqüências compostas apenas por As ou Ts. Conforme a etapa, diferentes critérios de pontuação, tamanho e distância do artefato à extremidade são utilizados.

O ESTprep utiliza uma estratégia diferente. Em primeiro lugar, ele percorre a seqüência em busca do primeiro nucleotídeo da cauda após a identificação do sítio de restrição. A partir desta posição, uma seqüência maximal formada apenas por A/Ts é construída de tal forma que ela possua similaridade maior que um limite pré-estabelecido (95%) em relação a seqüência original. Desta maneira, a região mais rica em A/T é encontrada. Se esta região não termina com um A/T, ela é retraída em uma base, o que é repetido até que a última base seja um A/T. Se a cauda de poli-A/T não tiver tamanho suficiente (10 bases), a busca é refeita começando uma base à esqueda/direita do ponto de início original. Se a cauda é identificada, a busca é repetida utilizando-se um limite menor (94% do limite original) para evitar o truncamento de caudas poli-A/T grandes. Se depois de todos estes passos a cauda não foi identificada, o programa analisa uma janela com o tamanho da distância média entre o sítio de restrição e o trecho que identifica o tecido (18 bases) em busca de uma densidade suficiente de A/Ts (65%). Após a identificação do poly-A, o programa tenta localizar os sinais de poliadenilação. Os sinais procurados podem ser canônicos (AAUAAA ou AUUAAA) ou alternativos. Eles devem estar dentro de 11 a 30 nucleotídeos a partir do final da cauda poli-A.

Remoção de trechos derrapados No estudo que fizemos, apenas Telles e da Silva citam a detecção e remoção de trechos derrapados.

A detecção e remoção deste tipo de artefato é feita através da análise da seqüência em busca de grupos ecoados (regiões com 5 ou mais bases idênticas consecutivas). O método considera apenas seqüências que possuam pelo menos 8 destes grupos ecoados.

Para avaliar se os grupos formam uma derrapagem, o método realiza um produto do tamanho dos grupos sendo que aqueles que tiverem tamanho maior ou igual a 10 contribuem apenas com 10. Caso o produto seja maior do que 10^8 e a soma do tamanho dos grupos ecoados corresponda a 20% do tamanho da seqüência, o método considera a região ecoada. A remoção desta região é feita de acordo com a presença de caudas poli-A/T. Se existir um poli-T na seqüência, ela é totalmente descartada. Se houver um poli-A, apenas o trecho que vai do início da cauda poli-A até o final da seqüência é removido. Caso nenhuma cauda seja encontrada, toda a seqüência é descartada.

A diferença de tratamento, conforme o tipo de cauda encontrada, se justifica pela

posição normal destes artefatos na seqüência. Como as caudas poli-T se encontram no início da seqüência normalmente, considera-se que toda a seqüência foi comprometida. Já as caudas poli-A costumam ocorrer na porção final e, por isso, acredita-se que a porção inicial não foi comprometida podendo, portanto, ser preservada.

2.10.2 Verificação de contaminação

Após a detecção e remoção de artefatos, normalmente se realiza a análise da seqüência em busca de contaminação. A contaminação de seqüências é um problema extremamente sério em projetos de seqüenciamento. Ocorrências embaraçosas têm acontecido com freqüência, como, por exemplo, projetos de seqüenciamento em larga-escala que utilizaram bibliotecas de clones altamente contaminadas e tiveram que descartar uma quantidade enorme de seqüências. Outro exemplo, foi o anúncio, em 1994, de que DNA havia sido extraído com sucesso a partir de ossos de um dinossauro [117]. Hoje em dia, este anúncio é visto como, no mínimo, prematuro. As seqüências "extraídas" se mostraram, através de buscas realizadas em bancos de seqüências de DNA, muito mais semelhantes às seqüências de mamíferos, do que de aves ou crocodilos, sugerindo que o DNA utilizado na análise fosse, na verdade, uma contaminação humana e não DNA de dinossauros [19, 122].

Tipos de contaminação

Existem vários tipos de contaminação que variam conforme o protocolo utilizado na produção de bibliotecas e na clonagem das seqüências [98]. As contaminações podem ser separadas em dois grupos diferentes: contaminações causadas por seqüências de outros organismos e contaminações causadas por seqüências do próprio organismo.

Contaminação por seqüências de outros organismos O vetor utilizado na clonagem pode ser uma fonte de contaminação. Devido aos eventos de rearranjo de genoma, seqüências do vetor podem ser inseridas no meio do inserto, formando uma seqüência híbrida.

Em um laboratório de seqüenciamento, é comum a execução de experimentos com organismos diferentes. Acidentalmente, é possível que uma biblioteca de clones de seqüências de um organismo seja contaminada com seqüências de outro organismo estudado no mesmo laboratório. Assim, o seqüenciamento dos ESTs desta biblioteca pode produzir um conjunto formado por seqüências do organismo estudado pelo projeto e outro formado por seqüências de outros organismos estudados no laboratório.

Existem projetos de seqüenciamento que lidam com tecidos que podem estar contaminados. Por exemplo, é comum a preparação de ESTs de tecidos atacados por alguma doença para obtenção dos genes que são expressos quando um organismo está doente. No

meio do conjunto de seqüências do organismo podem existir ESTs originários em mRNAs do organismo patógeno.

Outra possibilidade de contaminação ocorre quando se estuda organismos que vivem relações simbiontes. Existe a possibilidade de contaminação por seqüências do organismo que vive com o organismo estudado, pois durante a coleta de material existe a possibilidade da obtenção de DNA de ambos.

Contaminação por seqüências do próprio organismo Seqüências do próprio organismo também podem causar contaminação. Ela ocorre quando existe a formação de ESTs contendo trechos de seqüência que não possuem origem no mRNA processado.

O EST é uma seqüência produzida através do mRNA, mas durante o processo de produção das bibliotecas, pode ocorrer do rRNA ser utilizado pela enzima transcriptase reversa para produção do cDNA.

As células eucariotas tem mitocôndrias e as células eucariotas de vegetais também possuem cloroplastos. Estas organelas possuem um genoma próprio e conforme a natureza do projeto de seqüenciamento, a obtenção das seqüências de genes destas organelas pode ser desnecessário ou, até mesmo, indesejado.

Como vimos anteriormente, o mRNA, após a transcrição, é processado para remoção dos íntrons. Pode acontecer de um mRNA prematuro originar cDNA, que apesar de ter origem em um mRNA, contém trechos que não pertecem à porção codificante do gene.

Problemas no protocolo de produção de bibliotecas podem gerar cDNAs contendo trechos de DNA genômico, que não fazem parte de genes, algo indesejado quando se deseja obter o indice gênico de um organismo.

Finalmente, eventos de rearranjo de genoma podem gerar seqüências quiméricas. Estas seqüências se caracterizam por conter trechos de dois ou mais genes que possuem origens em pontos diferentes do genoma.

2.10.3 Técnicas de detecção de contaminação

A maior parte dos projetos utilizam a similaridade para a detecção de contaminação [79]. Normalmente o programa BLAST [5] é utilizado neste método, que consiste na comparação da seqüência a ser analisada com as seqüências existentes em um banco formado por possíveis organismos contaminantes.

Os critérios para detecção de contaminação através da similaridade podem variar entre diferentes projetos. Um exemplo de critério seria a identificação de contaminantes a partir de uma similaridade de pelo menos 98% ao longo de uma janela de 75 bases e com e-value menor ou igual a 10^{-15}.

A similaridade também é utilizada na detecção de seqüências provenientes de rRNA ou de genes pertencentes a mitocôndrias ou cloroplastos.

Ao indicar que uma seqüência é similar a de um contaminante existente no banco, este método pode dizer que ela é um possível contaminante, contudo, nada pode-se dizer das seqüências que não apresentaram similaridade com nenhuma do banco. Existe a possibilidade de algumas seqüências pertencerem a organismos não existentes no banco de contaminantes. Além disso, no caso do banco de contaminantes ser muito grande, a busca por similaridade pode ser muito custosa.

Além da detecção de contaminação através da similaridade, existem as técnicas que aplicam as características encontradas nos genomas dos organismos como critério. A abordagem destas metodologias é classificar as seqüências em dois grupos (seqüências pertencentes ao organismo alvo e seqüências não pertencentes ao organismo alvo) de acordo com as características obtidas pela análise delas em comparação com as obtidas através de um conjunto de treino formado por seqüências do próprio organismo e, opcionalmente, dos organismos contaminantes.

Diversas características podem ser utilizadas por estas metodologias. O trabalho desenvolvido por White *et al.* [116], por exemplo, utiliza a composição de hexâmeros. O estudo realizado por Piazza e Setubal [79, 80] emprega uma gama maior de características com o objetivo de melhorar a precisão da detecção de contaminantes.

Este tipo de metodologia é mais indicado para detecção de contaminações por DNA do próprio organismo. A análise é feita comparando-se as assinaturas apresentadas pelos ESTs contra as verificadas em genes do organismo. As assinaturas dos genes costumam ser bastante diferentes do restante do genoma, e esse fato pode auxiliar na detecção de contaminação por DNA genômico ou por mRNA prematuro.

As metodologias baseadas em características são menos utilizadas que as baseadas em similaridade. Elas apresentam uma taxa de erros maior e possuem a desvantagem de necessitarem de treino com seqüências do organismo, o que nem sempre é possível.

A detecção de seqüências quiméricas requer uma análise cuidadosa, pois trata-se de seqüências que são formadas pela fusão de dois ou mais genes. Como as quimeras são formadas geralmente por concatenação de genes de diferentes regiões do genoma, a maneira mais apropriada seria a utilização da comparação com o genoma completo do organismo, o que nem sempre é possível, especialmente no caso de projetos ESTs. Quimeras de genes conhecidos podem ser identificadas ao observar-se a concatenação de trechos de genes não relacionados.

2.10.4 Clusterização

A clusterização é o processo de agrupamento de seqüências em conjuntos chamados clusters, com o objetivo de identificar os genes e diminuir a redundância de informações produzidas pelo projeto de seqüenciamento. Este processo é extremamente importante para a identificação do conjunto de genes expressos no organismo.

Os clusters são formado por duas ou mais seqüências agrupadas através da utilização de critérios de similaridade. As seqüências que não se agrupam com outras, formando "clusters" de tamanho um, são normalmente denominadas singletons.

Técnicas de clusterização

As técnicas de clusterização utilizam a similaridade como principal ferramenta. Através das comparações entre cada par de seqüências é possível determinar aquelas que se sobrepõem e que, portanto, podem ter origem no mesmo gene.

Diferentes implementações de processos de clusterização estão disponíveis e elas diferem na estratégia adotada para otimizar o desempenho sem comprometer a qualidade dos clusters produzidos.

Os programas de clusterização normalmente realizam uma avaliação inicial para identificação das seqüências que podem se sobrepor. Feita a identificação, as seqüências são alinhadas para construção dos clusters.

O ideal seria que todas as seqüências pudessem ser unidas através de um alinhamento múltiplo, no entanto, este é um problema NP-completo. Isso explica a necessidade de se fazer uma avaliação inicial em busca de seqüências que se sobrepõem e a aplicação de uma série de heurísticas realizada por diversos softwares.

Os clusters produzidos podem ter ou não associados a eles seqüências consensos. A seqüência consenso é derivada através da análise das seqüências que formam o cluster e é aceita como a seqüência com maior probabilidade de ser a do gene existente no organismo.

Alguns programas de clusterização podem realizar o alinhamento das seqüências existentes nos clusters com o objetivo de produzir melhores consensos.

Um dos programas mais utilizados é o CAP3 [54], mas existem muitos outros, como o Phrap [44], o TIGR Assembler [101], o UIcluster [109] e o TGICL [78].

Capítulo 3

Nova estratégia de detecção e remoção de artefatos

A maioria dos projetos de seqüenciamento realizam a detecção e remoção de artefatos em diversas etapas. Normalmente, essas etapas são executadas de maneira que o resultado de uma serve como entrada para a etapa seguinte. Neste tipo de padrão de execução, freqüentemente ocorrem situações em que o artefato detectado por uma etapa não é encontrado em função da identificação ou não de um artefato na etapa anterior.

Por exemplo, suponha que a etapa de detecção de poli-A é executada após a remoção de artefatos de vetor. Suponha também, que a detecção de poli-A é feita quando se encontra uma região de pelo menos 15 As que se situa no máximo a 10 bases da extremidade $3'$ da seqüência resultante da etapa anterior. Com a utilização destes tipos de critérios, a detecção de um poli-A pode ser prejudicada, por exemplo, por regiões de baixa qualidade, que impedem a identificação completa do vetor. Se o vetor não é identificado completamente, a distância entre o artefato poli-A e a extremidade da seqüência pode fica maior do que a exigida pelo método, implicando na não detecção do artefato.

Baseado neste problema, decidimos desenvolver um novo conjunto de procedimentos de remoção e detecção de artefatos baseado em uma nova estratégia. Ela consiste na execução independente das etapas de detecção de artefatos. A apresentação deste conjunto de procedimentos e desta nova estratégia é o objetivo deste capítulo

O nosso trabalho utilizou como base inicial o conjunto de procedimentos desenvolvidos para o projeto de seqüenciamento de ESTs da cana-de-açúcar (SUCEST). Estes procedimentos serão apresentados na Seção 3.1.

Na Seção 3.2 descreveremos o nosso conjunto de métodos desenvolvido com o intuito de verificar a viabilidade da utilização desta nova abordagem de detecção e remoção de artefatos.

A Seção 3.3 apresentará os resultados produzidos a partir da utilização de nosso con-

junto de procedimentos na análise de ESTs obtidos do projeto Cattle EST [8, 23, 64], que estudou seqüências do boi (*Bos taurus*).

Finalmente, na Seção 3.4 discutiremos os resultados obtidos para avaliar a validade da nova abordagem.

O estudo desenvolvido neste capítulo foi apresentado no congresso "Brazilian Symposium on Bioinformatics 2005 (BSB2005)", realizado em Julho de 2005, em São Leopoldo – RS, sob o título "New EST Trimming Strategy". Um resumo estendido foi publicado nos anais do congresso [10]. O estudo completo foi depositado como relatório técnico, identificado pelo código "IC-05-09", no Instituto de Computação – Unicamp [11].

3.1 Métodos de detecção e remoção de artefatos utilizados no Projeto SUCEST

O nosso conjunto de procedimentos de detecção e remoção de artefatos baseou-se no conjunto utilizado no projeto SUCEST. Este conjunto foi desenvolvidos por Telles e da Silva e apresentado no trabalho "Trimming and clustering sugarcane ESTs" [104]. Ele é composto por oito etapas que serão descritas a seguir:

3.1.1 Remoção de RNA ribossomal

3.1.2 Mascaramento de vetor e adaptador

3.1.3 Remoção de vetor e de poli-A

3.1.4 Remoção de pontas de baixa qualidade

3.1.5 Remoção de vetor próximo à extremidade

3.1.6 Remoção de trecho derrapado

3.1.7 Remoção de poli-A grande ou próximo à extremidade

3.1.8 Remoção de seqüências curtas e de baixa qualidade

3.1.1 Remoção de RNA ribossomal

A primeira etapa é destinada a detecção de seqüências que possuam RNA ribossomal.

Seqüências ribossomais são identificadas através da execução do BLAST [5] da seqüência analisada contra um banco formado por seqüências ribossomais. Se a seqüência apresentar ao menos um hit com e-value menor ou igual a 10^{-10}, considera-se que ela possui conteúdo ribossomal e ela é descartada completamente.

3.1.2 Mascaramento de vetor e adaptador

As seqüências que não foram descartadas no primeiro passo são submetidas ao mascaramento das seqüências de vetores e adaptadores.

O mascaramento é feito através do alinhamento da seqüência analisada com as seqüências de vetores utilizando o software cross_match com parâmetros `-minmatch` 12 `-minscore` 20 `-penalty` -2 `-screen` e o esquema de pontuação de alinhamento padrão do programa.

Se o projeto utilizar algum adaptador na construção das bibliotecas, a seqüência de vetor deverá incluí-lo. Dessa maneira, o adaptador será removido junto com o vetor.

Todos os trechos mascarados (marcados com X) serão considerados como candidatos a artefatos de vetor.

3.1.3 Remoção de vetor e de poli-A

De posse dos candidatos a artefatos de vetor, o próximo passo é analisar as regiões vizinhas para decidir quais trechos serão efetivamente descartados.

A primeira parte desta análise é a verificação do tamanho da região não coberta por vetor. O tamanho de cada trecho mascarado no passo anterior é somado e o valor obtido é subtraído do tamanho da seqüência. Se o valor restante for menor que o tamanho mínimo aceitável (100 bases), a seqüência é inteiramente descartada.

Se o tamanho da seqüência não coberta por vetor for maior que o mínimo aceitável, inicia-se a segunda parte, que analisa diversos casos:

1. Existem exatamente duas regiões mascaradas: as regiões que vão do início da seqüência até o final da região mascarada mais próxima a extremidade $5'$ e do início da região mascarada mais próxima a extremidade $3'$ até o final da seqüência são descartadas como artefatos de vetor.

2. Existem mais de duas regiões mascaradas: neste caso, nenhuma alteração é feita na seqüência.

3. Existe apenas uma região de vetor (mascarada):

   (a) Se a posição inicial da região de vetor estiver entre as 50 primeiras bases da seqüência e o seu tamanho for menor ou igual a 300, descarta-se o trecho que vai do início da seqüência até o final da região.

   (b) Se a posição inicial da região estiver entre as 50 primeiras bases e o seu tamanho for maior que 300, toda a seqüência é marcada como artefato de vetor e, portanto, descartada completamente.

(c) Se a posição inicial da região estiver entre as bases 51 e 300 da seqüência e seu tamanho for menor que 300, não é possível decidir se realmente existe vetor e, por isso, nenhuma alteração é feita na seqüência.

(d) Se a posição inicial da região estiver entre as bases 51 e 300 da seqüência e seu tamanho for maior que 300 ou se ela estiver após a base 300 da seqüência, descarta-se o trecho que vai do início da região até a extremidade final da seqüência.

Nas situações onde houve descarte de artefatos de vetor em pelo menos uma das extremidades da seqüência, este passo realiza também a remoção de caudas poli-A/T.

Para isso, ele utiliza o programa swat para realizar o alinhamento da seqüência sem vetor com seqüências de prova compostas somente por As ou por Ts. O esquema de pontuação utilizado considera 1 ponto para cada acerto, -2 para cada erro e -8 para cada buraco aberto.

Serão descartados como artefatos poli-A/T todas as regiões que apresentarem alinhamento com pontuação mínima 8 e se estiverem localizadas no máximo a 10 bases de distância de uma das extremidades da seqüência sem vetor. A pequena região situada entre o poli-A e a extremidade é considerada como um provável vetor e também é descartada.

3.1.4 Remoção de pontas de baixa qualidade

A remoção de baixa qualidade é feita através da execução de um algoritmo de janela deslizante sobre a seqüência resultante da etapa anterior.

A janela utilizada para analisar a seqüência possui tamanho 20. Ela percorre a seqüência a partir da extremidade $5'$, em direção a extremidade oposta, em busca da primeira região que apresentar no máximo 12 bases com qualidade menor do que 10. Quando esta região é encontrada, todo o trecho percorrido pela janela antes da região é descartado como artefato de baixa qualidade. O procedimento é repetido a partir da extremidade $3'$.

3.1.5 Remoção de vetor próximo à extremidade

Após a remoção de baixa qualidade, a seqüência resultante é analisada novamente em busca de artefatos de vetor. Se existirem regiões candidatas localizadas no máximo a 10 bases de distância de uma das extremidades, elas são removidas como artefatos de vetor, junto com o trecho situado entre a região e a extremidade da seqüência.

3.1.6 Remoção de trecho derrapado

A derrapagem é removida através da análise da seqüência produzida no passo anterior em busca de regiões ecoadas. Uma região ecoada é aquela que possui um número pré-determinado de bases idênticas e consecutivas. No caso deste método, são consideradas regiões ecoadas aquelas que apresentarem tamanho maior ou igual a 5.

A partir da lista de regiões ecoadas, o método calcula o produto de seus tamanhos da seguinte forma:

1. O valor inicial do produto é 1.
2. Se a região identificada tiver tamanho maior que 10, $produto = produto \times 10$.
3. Se a região identificada tiver tamanho menor ou igual a 10, $produto = produto \times$ *tamanho da região*.

Se o produto tiver valor maior que 10^8 e a cobertura das regiões (soma do tamanho dos trechos / tamanho da seqüência) for maior ou igual a 20% considera-se que a seqüência está derrapada.

Para decidir qual trecho será descartado, o primeiro passo é a execução do swat com os mesmos parâmetros utilizados anteriormente, comparando a seqüência contra um arquivo fasta contendo uma longa seqüência de Ts. Se um alinhamento for encontrado e sua pontuação for maior ou igual a 40, toda a seqüência será descartada como derrapagem.

Caso nada tenha sido encontrado, o swat é novamente executado para comparação da seqüência contra um arquivo fasta contendo uma longa seqüência de As. Se um alinhamento for encontrado e a sua pontuação for maior ou igual a 40, um artefato é criado indicando derrapagem na extremidade $3'$. O artefato vai do início do alinhamento ao final da seqüência.

A diferença de tratamento entre a detecção utilizando o poli-T e o poli-A deve-se ao fato das suas posições relativas na seqüência. Acredita-se que a derrapagem possa ser conseqüência da presença de uma cauda poli-A/T de altíssima qualidade que provoca erros de leitura da máquina de seqüenciamento. Como a cauda poli-T geralmente aparece no início da seqüência, ela acaba por afetar a seqüência toda e por isso a seqüência é marcada como sendo totalmente derrapada. A cauda poli-A, por outro lado, ocorre, geralmente, no final da seqüência afetando apenas a porção final. Assim, ao marcar apenas a região que começa no início do alinhamento e termina na posição final da seqüência, o procedimento tenta preservar a parte inicial da seqüência que, provavelmente, não foi afetada pela derrapagem.

Na situação em que nenhum dos dois tipos de cauda são encontrados, toda a seqüência é descartada pois não se pode afirmar exatamente qual é a região que foi prejudicada pela derrapagem.

Uma observação importante sobre esta etapa, é que ela não exige que os grupos ecoados estejam fisicamente consecutivos ou mesmo próximos uns dos outros.

3.1.7 Remoção de poli-A grande ou próximo à extremidade

As seqüências que foram preservadas pelo processo de remoção de derrapagem, são submetidas a uma nova etapa de remoção de caudas poli-A/T.

Primeiro, o alinhamento da seqüência com uma seqüência de prova composta por Ts é feito utilizando, novamente, o programa swat. Caso exista uma região que apresente pontuação 8 e esteja no máximo a 20 bases da extremidade, ela é descartada junto com o trecho entre a região e a extremidade.

Feito isso, verifica-se se existe uma região composta por pelo menos 50 As consecutivos. Se existir, o alinhamento da seqüência com uma seqüência de prova composta por As é realizado. Se algum trecho apresentar alinhamento com pontuação maior ou igual a 70, este será descartado.

Finalmente, repete-se procedimento semelhante ao primeiro passo desta etapa, mas utilizando uma seqüência de prova composta por As.

3.1.8 Remoção de seqüências curtas e de baixa qualidade

A última etapa do conjunto de procedimentos, desenvolvido por Telles e da Silva, realiza a eliminação de seqüências curtas ou com qualidade menor do que a desejada.

Após passar por todas as etapas anteriores, a seqüência deverá apresentar um tamanho mínimo de 100 bases e possuir pelo menos 50 bases com qualidade maior ou igual a 20.

3.2 Procedimento básico de métodos de detecção e remoção de artefatos

O nosso conjunto de procedimentos utilizou como base o trabalho de Telles e da Silva. A partir da análise dos procedimentos descrito por eles, decidimos construir métodos que promovessem a detecção e remoção de artefatos ribossomais, de baixa qualidade, de vetores, de adaptadores e de caudas poli-A/T.

Note que, neste momento, não trabalhamos com artefatos de derrapagem. Devido às características do artefato e devido à pouca literatura disponível sobre ele, decidimos realizar um estudo mais aprofundado posteriormente. Este estudo está descrito no Capítulo 4.

Para a confecção dos procedimentos utilizamos uma nova estratégia em que as etapas de detecção de artefatos são independentes entre si. A remoção dos artefatos é feita,

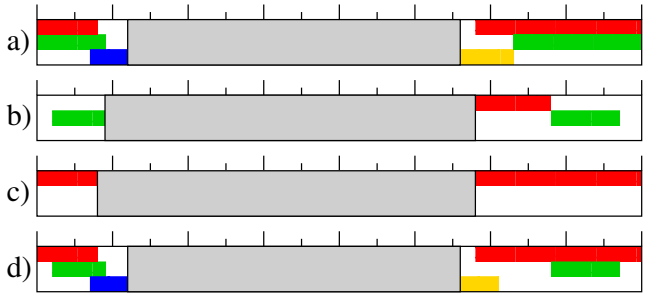

Figura 3.1: Esquemas para demonstração de ESTs e seus artefatos. a) Seqüência original: a região cinza corresponde ao inserto, as regiões vermelhas denotam as pontas de baixa qualidade, as regiões verdes indicam os trechos de vetores, a região azul indica o adaptador e a região amarela a cauda poli-A. b) Resultado do processamento da seqüência original por um conjunto que realiza a detecção de artefatos de vetor, baixa qualidade, adaptador e poli-A (nesta ordem) de modo que o resultado de uma etapa é a entrada da etapa seguinte. c) Resultado do processamento da seqüência original por conjunto semelhante ao anterior mas com a ordem de vetor e de baixa qualidade trocadas. d) Resultado do processamento da seqüência original segundo a nossa estratégia.

de fato, na última etapa, que é responsável pela combinação dos artefatos detectados e identificação da região de boa qualidade da seqüência.

Esta nova estratégia tem o objetivo de tentar diminuir a ocorrência de erros do tipo falso negativo que ocorrem quando um artefato não é detectado em função de algum problema na detecção de outro.

Para melhor entendimento do efeito que a inter-dependência de etapas pode produzir no resultado final da limpeza de seqüências, vamos analisar as seqüências da Figura 3.1.

A Figura 3.1a exibe um EST típico com todos os artefatos e o inserto marcados. Nesta figura, a região cinza é o inserto, as regiões vermelhas são os trechos de baixa qualidade, as regiões verdes são os trechos de vetor, a região amarela é o trecho de poli-A e a região azul o trecho de adaptador.

Suponha um procedimento de detecção de artefatos que identifica os artefatos na seguinte ordem: detecção de vetores, detecção de baixa qualidade, detecção de adaptadores e detecção de poli-A. Suponha também que a entrada de cada etapa é resultado da saída da etapa anterior. O resultado do processamento da seqüência da Figura 3.1a, segundo este procedimento, é demonstrado pela Figura 3.1b. Nesta situação, podemos ver que a etapa de detecção de vetores foi capaz de encontrar pedaços dos trechos de vetor (os trechos de vetor não foram detectados completamente devido à baixa qualidade da seqüência na região em que eles se encontram). Após a detecção de trechos de vetor, a seqüência resultante é analisada em busca da baixa qualidade e, neste caso, apenas a baixa qualidade da extremidade $3'$ é encontrada, pois a baixa qualidade $5'$ foi removida junto com o

trecho de vetor. Contudo, após a execução destas duas etapas, nós podemos ver que os trechos de adaptador e de poli-A que sobram são muito pequenos o que inviabiliza as suas detecções pelas etapas que buscam estes tipos de artefatos. Assim, o inserto resultante é maior do que o esperado.

Agora, suponha um procedimento semelhante ao anterior, mas que inverte a ordem de detecção de artefatos de vetor e de baixa qualidade. A Figura 3.1c mostra que, nesta situação, as pontas de baixa qualidade seriam encontradas inteiramente. Contudo, os trechos de vetor, adaptador e poli-A ficariam muito pequenos para serem detectados pelas etapas subseqüentes. Logo, nós teríamos como resultado, novamente, um inserto maior do que o real.

A Figura 3.1d mostra o resultado quando a seqüência é processada pelas etapas de maneira independente. As pontas de baixa qualidade e o adaptador são identificadas completamente e os vetores e a cauda poli-A são encontrados como trechos menores do que os originais devido à baixa qualidade da região onde se encontram. Contudo, a composição final dos artefatos permite a identificação correta do inserto.

Além desta diferença de abordagem, nosso conjunto de métodos procurou simplificar e diminuir o número de passos de detecção de artefatos apresentados por Telles e da Silva. Isso foi feito com o objetivo de se produzir um conjunto de procedimentos básicos para obtenção de resultados que permitissem a identificação de passos que realmente necessitam de maior especialização.

O nosso conjunto de procedimentos é formado por cinco etapas de detecção de artefatos e uma etapa de combinação de artefatos para identificação da região de boa qualidade da seqüência. Cada uma dessas etapas será apresentada a seguir.

3.2.1 Detecção de artefatos ribossomais em seqüências ESTs

A etapa de detecção de seqüências ribossomais é idêntica à utilizada por Telles e da Silva.

Nós consideramos o procedimento adotado por eles como adequado devido à grande conservação apresentada pelas seqüências ribossomais. Este fato permite a utilização do programa BLAST para identificação destes tipos de seqüências.

Contudo, deve-se observar que a detecção pode ser melhorada quando se utiliza seqüências de organismos filogeneticamente próximos ao organismo que originou a seqüência.

Em seu trabalho, Telles e da Silva utilizaram seqüências de organismos filogeneticamente próximos à cana-de-açúcar e obtiveram bons resultados.

3.2.2 Detecção de artefatos de baixa qualidade

Para identificação de regiões de baixa qualidade decidimos utilizar o mesmo algoritmo implementado pelo programa de base-calling phred [44].

O algoritmo processa uma seqüência de valores derivada da seqüência de qualidades em busca da subseqüência de máxima pontuação.

A seqüência que será processada é obtida através da subtração do valor de probabilidade de erro de cada base do valor $0,05$. O valor de probabilidade de erro pode ser obtido através da fórmula $Q = -10 \times \log(\textit{probabilidade de erro})$, onde Q é a qualidade da base.

Ao subtrair o valor da probabilidade de erro do valor $0,05$ a faixa que determina o limiar para início e fim de uma subseqüência é deslocada para o valor equivalente à qualidade 13. Isso significa que, segundo este critério, a probabilidade máxima de erro aceitável é de 5%.

A subseqüência de máxima pontuação encontrada pelo algoritmo é determinada como sendo a região de boa qualidade. Assim, as regiões que ficam entre as extremidades da seqüência e as extremidades da região de boa qualidade são marcadas como artefatos de baixa qualidade.

A escolha deste algoritmo está relacionado ao melhor comportamento das qualidades das bases das extremidades da região de boa qualidade definida por ele. Este algoritmo garante que as qualidades da primeira e da última base da região de boa qualidade serão maiores ou iguais à qualidade mínima aceitável (13 no caso). Além disso, as bases próximas às extremidades não podem ter qualidades muito baixas pois, caso isso ocorresse, a inclusão delas na região de máxima pontuação não aconteceria.

No caso do algoritmo de janela deslizante, como descrito no trabalho de Telles e da Silva, isso não ocorre. Como a janela aceita um determinado número de bases de baixa qualidade, sem nenhuma restrição extra, nada pode-se dizer quanto ao comportamento das qualidades das bases nas extremidades da região determinada como sendo de boa qualidade.

Outro aspecto vantajoso do algoritmo de subseqüência máxima é que ele analisa a seqüência inteira. O algoritmo de janela deslizante interrompe a análise assim que encontra uma região que atenda aos seus critérios mínimos.

3.2.3 Detecção de artefatos de vetor

A remoção de trechos de vetor é feita com a utilização do programa cross_match [44] que é executado com os seguintes parâmetros:

cross_match [fasta_seqüência] [fasta_vetor] -minmatch 12 -minscore 20

O programa é executado com a utilização dos valores padrões de pontuações para os alinhamentos. A pontuação para cada base correspondente é 1 e para cada substituição é -2. A abertura de buraco na primeira seqüência é penalizada com -4 e a sua extensão recebe a pontuação -3.

A saída do programa é analisada para identificação dos trechos que apresentaram alinhamentos com a seqüência do vetor. Cada trecho encontrado será considerado um artefato de vetor. Se nenhum trecho for encontrado, considera-se que a seqüência não possui trechos de vetor.

3.2.4 Detecção de trechos de adaptadores

Para realizar a detecção de adaptadores, utilizamos o programa swat [44]. Os parâmetros passados para o programa são os seguintes:

swat [fasta_adaptadores] [fasta_seqüência] -gap_init -5 -gap_ext -5 -ins_gap_ext -5 -del_gap_ext -5 -end_gap -5 -raw -minscore 4 -M [matriz_de_pontuação]

A matriz de pontuação utilizada é exibida na Tabela 3.1.

	A	C	G	T	N	X
A	1	-2	-2	-2	0	-3
C	-2	1	-2	-2	0	-3
G	-2	-2	1	-2	0	-3
T	-2	-2	-2	1	0	-3
N	0	0	0	0	0	0
X	-3	-3	-3	-3	0	-3

Tabela 3.1: Matriz de pontuação utilizada com o swat.

A saída do programa é processada em busca de regiões da seqüência que apresentaram alinhamento com a seqüência do adaptador. Toda região que apresenta alinhamento de tamanho maior ou igual a $t - 4$, sendo t o tamanho do adaptador, é marcada como artefato. Caso existam duas regiões, aquela que apresentar maior pontuação será considerada artefato. Em caso de novo empate, a região mais próxima a extremidade é escolhida.

3.2.5 Detecção de caudas Poli-A/T

Nesta etapa nós também utilizamos o programa swat. Os parâmetros utilizados foram os mesmos do passo anterior. Porém, neste caso, o primeiro arquivo fasta contém a seqüência a ser analisada e o segundo arquivo contém uma seqüência modelo formada por 500 As (ou Ts, conforme o tipo de cauda que está sendo detectada).

Todos os alinhamentos que apresentam pontuação mínima de 10 são marcados como artefatos poli-A (ou poli-T).

3.2.6 Remoção de artefatos e identificação da seqüência de boa qualidade

Após a realização de todos os passos acima, esta última etapa realiza o mascaramento de todos os artefatos detectados na seqüência analisada.

Feito isso, todos os trechos não mascarados são verificados. Aqueles que apresentarem tamanho menor do que 100 são descartados por serem muito curtos.

As seqüências não mascaradas com tamanho maior do que 100 devem apresentar pelo menos 50 bases com qualidade maior ou igual a 20, caso contrário, elas também são descartadas.

Caso exista mais de uma região não mascarada que atenda aos critérios acima, apenas a maior será preservada. Se houver empate no tamanho, a região que apresentar maior soma de qualidade é considerada.

3.3 Aplicação dos métodos às seqüências do projeto Cattle EST

Para avaliar o conjunto de procedimentos descrito na seção anterior, nós realizamos o processamento de ESTs provenientes do projeto Cattle EST [8, 23, 64] que seqüenciou cDNAs do boi (*Bos taurus*).

A Seção 3.3.1 apresentará os dados utilizados nos testes. A Seção 3.3.2 mostrará os resultados obtidos pelos métodos. Finalmente, na Seção 3.3.3 faremos uma avaliação da nova estratégia de detecção de artefatos.

3.3.1 Dados utilizados nos testes

Nós decidimos realizar os testes utilizando dados de projetos de seqüenciamentos com o objetivo de submeter os métodos aos problemas encontrados em um projeto real. Nós não tínhamos disponíveis ainda as seqüências do projeto SUCEST, que foram utilizadas nos estudos dos próximos capítulos e, por isso, iniciamos uma pesquisa em busca de cromatogramas.

Apesar de existirem diversos projetos, poucos disponibilizam os cromatogramas das seqüências. Normalmente os projetos tornam públicos apenas os resultados já processados.

Nós encontramos os cromatogramas necessários ao nosso estudo na página do projeto Cattle EST [23]. Conseguimos obter dois arquivos contendo os cromatogramas produzidos a partir do seqüenciamento dos cDNAs de duas bibliotecas de tecidos de orgãos do boi (*Bos taurus*). Nós também coletamos dados processados pelo projeto e armazenados em um base de dados que pode ser acessada através do programa ESTIMA [59].

Os principais objetivos do Cattle EST Project foram a construção de um mapa comparativo dos genomas bovino e humano, a produção de dados para identificação de genes importantes economicamente, a análise evolucionária dos cromossomos dos mamíferos e a análise funcional do genoma bovino.

As duas bibliotecas seqüenciadas foram produzidas a partir de tecidos da placenta e do baço. A biblioteca produzida a partir da placenta (BP) deu origem a 174 placas contendo 96 poços cada. No entanto, apenas 12.620 cromatogramas (75, 55% do total), correspondentes às seqüências não descartadas durante o processo de detecção e remoção de artefatos do projeto, estão disponíveis para *download*. A biblioteca de baço (BS) deu origem a 63 placas de 96 poços e a página do projeto disponibiliza 5.090 cromatogramas (84, 16% do total) para download.

O processo de detecção e remoção de artefatos do projeto realizou o descarte de seqüências contendo trechos repetitivos e trechos de RNA mitocondrial e ribossomal. Seqüências que, após o processo de remoção de artefato, tornaram-se muito pequenas também foram descartadas.

Todas as seqüências dos cromatogramas obtidos foram submetidas pelo projeto ao dbEST. Através do código de identificação das seqüências no dbEST foi possível obter no NCBI dados necessários ao processo de detecção e remoção de artefatos. A lista das informações obtidas para as duas bibliotecas podem ser visualizadas na Tabela 3.2.

Biblioteca	**Placenta (BP)**	**Baço (BS)**
Nome	Soares normalized bovine placenta	Subtracted Lewin Cattle Spleen
Placas	174	63
Seqüências	12.620	5.090
Variedade	n/a	Angus
Vetor	pT7T3Pac	pBluescript SK+
Adaptador	5′-AATTCGGCACGAGG-3′	5′-AATTCGGCACGAGG-3′
Sítio de restrição 1	EcoRI	EcoRI
Sítio de restrição 2	NotI	XhoI

Tabela 3.2: Informações obtidas no site do NCBI sobre as bibliotecas de cDNA de placenta e de baço do projeto Cattle EST.

Além dos cromatogramas, nós obtivemos, para cada EST contido nas bibliotecas, as seqüências de bases originais e as seqüências de bases após o processamento delas pelos métodos de limpeza do projeto. Seqüências de qualidade não estavam disponíveis.

As seqüências de bases originais apresentaram um tamanho médio de $821,93 \pm 67,39$ bp, enquanto as seqüências processadas apresentaram tamanho médio de $465,93 \pm 101,49$ bp. As seqüências originais foram obtidas através da utilização de programas do pacote

que acompanham as máquinas de seqüenciamento ABI utilizadas. Os cromatogramas foram manualmente processados com o programa SeqEd.

3.3.2 Resultados obtidos pelo conjunto de procedimentos

Para realização dos testes, decidimos trabalhar com as seqüências da biblioteca de placenta devido à maior quantidade de seqüências que ela possui. Porém, antes de executar os métodos foi necessário obter as seqüências de bases e de qualidade dos ESTs. Para isso, processamos os 12.620 cromatogramas utilizando o programa de base-calling phred [44]. As seqüências apresentaram tamanho médio de $820,79 \pm 66,65$ bp e qualidade média $35,37 \pm 17,70$. A razão da diferença apresentada entre os tamanhos médios obtidos pelo phred e os tamanhos das seqüências obtidas pelo projeto é explicada pela utilização de dois algoritmos distintos de base-calling (phred x ABI).

Para a fase de detecção de artefatos ribossomais, montamos um banco composto por seqüências ribossomais de organismos filogeneticamente próximos ao boi. Utilizamos duas seqüências de dois mamíferos diferentes: sub-unidade ribossomal 28S do camundongo (*Mus musculus*) e sub-unidade ribossomal 18S do porco (*Sus sucrofa*). Estas seqüências foram obtidas no NCBI e possuem, respectivamente os identificadores `gi:53988` e `gi:37956930`.

As informações que coletamos sobre o projeto diziam que seu processo de limpeza de seqüências realizara, entre outras coisas, a remoção de seqüências ribossomais. Assim, acreditávamos que não iríamos encontrar nenhuma ocorrência deste tipo de artefato pois as seqüências obtidas foram utilizadas na construção do clustering do projeto. Contudo, nós identificamos 100 seqüências com conteúdo ribossomal. Destas seqüências, 98 foram identificadas com a sub-unidade 28S e 2 com a sub-unidade 18S.

A fase de detecção de baixa qualidade encontrou artefatos em todas as seqüências em ambas as extremidades. No entanto, não ocorreu nenhum caso em que o tamanho da região de boa qualidade fosse menor do que 100 bases.

Artefatos de vetor foram encontrados em 12.461 seqüências ($99,53\%$ das 12.520 seqüências que não foram descartadas na fase de detecção de conteúdo ribossomal).

A etapa de busca de trechos de adaptadores encontrou artefatos em 12.311 seqüências ($98,33\%$).

Como o adaptador é uma seqüência relativamente pequena de DNA, nós analisamos a distribuição dos artefatos encontrados em função de seus tamanhos. Esta distribuição é exibida na Tabela 3.3.

Nós analisamos manualmente todas as 56 seqüências que apresentaram artefato de adaptador com tamanho menor do que 14 (tamanho do adaptador original). Esta análise mostrou que em todos os casos, os artefatos estavam próximos a regiões de vetor, o que

Tamanho do artefato	*Número de seqüências*
10	3
11	12
12	17
13	24
14	12.255
Total	*12.311*

Tabela 3.3: Distribuição dos artefatos de adaptador segundo os seus tamanhos. Note que o tamanho do adaptador utilizado nas construção da biblioteca BP é 14.

evidencia que eles, realmente, indicam posições de adaptador.

Nosso procedimento de detecção de caudas poli-A encontrou este tipo de artefato em 1.957 seqüências (15, 63%). Caudas poli-T foram encontradas em 955 seqüências (7, 63%).

A fase final de combinação de artefatos não removeu nenhuma seqüência pois, mesmo após o mascaramento de todos os artefatos, as regiões não mascaradas de todas as 12.520 sem conteúdo ribossomal atendiam aos critérios de tamanho e qualidade mínimos impostos.

Este conjunto final de seqüências apresentou tamanho médio de $560,06 \pm 114,88$ bp e qualidade média $43,15 \pm 21,06$.

Para avaliar a qualidade dos resultados produzidos pelos processos de detecção e remoção de artefatos, decidimos fazer comparação entre os clusterings produzidos com as seqüências processadas pelo projeto e as seqüências processadas pelos nossos métodos.

Três clustering foram construídos com a utilização do programa CAP3 configurado com seus parâmetros padrão. O primeiro clustering, denominado `Clustering I` foi criado com as seqüências processadas pelo projeto. Os clusterings denominados `Clustering II` e `Clustering III` foram construídos com as seqüências processadas pelos nossos métodos, sendo que o primeiro não utilizou as seqüências de qualidades. O `Clustering II` foi criado para que nós pudéssemos comparar as nossas seqüências com as do projetos em iguais condições e o `Clustering III` foi criado para que nós pudéssemos visualizar como o clustering realmente seria construído em condições normais.

O `Clustering I` apresentou um total de 7.179 clusters, sendo 4.681 singletons (clusters de tamanho 1) e 2.498 contigs (clusters de tamanho maior do que 1). Os singletons apresentaram tamanho médio de $462,61 \pm 104,49$ bp. Os contigs agruparam 7.939 seqüências e seus consensos apresentaram tamanho médio de $655,38 \pm 210,02$ bp.

O segundo clustering produziu um total de 7.045 clusters. Os 4.504 singletons apresentaram tamanho médio de $556,86 \pm 119,48$ bp. Os 2.541 contigs agruparam 8.016

seqüências e os seus consensos mostraram tamanho médio de $763,13 \pm 239,61$ bp.

Finalmente, o `Clustering III` produziu 7.089 clusters dividido em 4.539 singletons e 2.550 contigs. Os singletons mostraram tamanho médio de $557,20 \pm 119,29$. Os contigs agruparam 7.981 seqüências e seus consensos apresentaram tamanho médio de $763,22 \pm 239,46$

A Tabela 3.4 exibe a distribuição dos clusters em função de seus tamanhos para os três clusterings produzidos.

Um fato interessante em relação ao `Clustering I` é a presença de um cluster composto por 93 seqüências. Estas seqüências fazem parte da lista de 100 seqüências que nosso método apontou como sendo ribossomal. Contudo, devemos observar que as sete seqüências restantes se agruparam com seqüências que não são ribossomais em outros clusters, algo que não é desejado.

3.3.3 Avaliação da nova estratégia de detecção de artefatos

Os resultados exibidos até aqui mostram o comportamento global do procedimento que desenvolvemos, mas pouco revelam sobre as vantagens da nova estratégia adotada. Assim, realizamos uma análise sobre todos os artefatos identificados nas seqüências em busca de situações em que eles seriam perdidos devido à detecção de um determinado tipo de artefato.

A análise consistiu em verificar para cada sequência a sua lista de artefatos e identificar situações onde os artefatos de vetor, poli-A, poli-T e adaptador possuíam intersecção com os artefatos de baixa qualidade de modo que a região não coberta pela baixa qualidade tivesse tamanho menor que o mínimo necessário para identificação do artefato. Segundo os métodos utilizados neste capítulo, os tamanhos mínimos necessários para os artefatos serem identificados são 10 bases para adaptador, poli-A e poli-T e 20 para vetor.

Esta análise simula procedimentos de detecção de artefatos que realizam duas etapas. A primeira etapa é a de detecção de baixa qualidade. A segunda etapa é realizada em cima do resultado da primeira para busca de um dos outros tipos de artefatos.

Identificamos a ocorrência de 309 intersecções de artefatos de baixa qualidade com artefatos de adaptador, poli-A e vetor que seriam perdidos devido ao tamanho da região não contida na intersecção. Nenhuma ocorrência com poli-T foi observada. Estas intersecções ocorreram em 308 seqüências. Apenas uma seqüência apresentou duas instersecções: vetor na extremidade $5'$ e poli-A na extremidade $3'$. A Tabela 3.5 mostra o número de artefatos que apresentaram intersecção com baixa qualidade, os seus tamanhos médios e os tamanhos médios das regiões não cobertas pela baixa qualidade conforme os tipos de artefatos.

As 308 seqüências que apresentaram este tipo de problema equivalem a $2,46\%$ do

Tamanho	`Clustering I`	`Clustering II`	`Clustering III`
1	4.681	4.504	4.539
2	1.364	1.366	1.385
3	565	583	584
4	230	247	244
5	128	132	130
6	70	73	71
7	43	39	38
8	38	38	37
9	10	12	14
10	10	13	11
11	7	6	4
12	3	4	4
13	7	7	7
14	5	2	2
15	1	3	3
16	2	3	3
17	5	5	5
18	1	0	0
20	1	1	1
21	1	0	0
22	0	1	1
23	1	2	2
24	1	0	0
25	1	1	1
27	1	1	1
28	1	0	0
33	1	1	1
47	0	0	1
50	1	1	0
93	1	0	0
Total	*7.179*	*7.045*	*7.089*

Tabela 3.4: Distribuição dos clusters em função de seus tamanhos. `Clustering I` foi produzido pelo projeto Cattle EST, `Clustering II` e `Clustering III` foram produzidos com as seqüências processadas pelos nossos métodos de detecção e remoção de artefatos, sendo que o primeiro não utilizou informações de qualidades das seqüências.

Artefato	Baixa qualidade	Ocorrências	Tamanho médio do artefato	Tamanho médio da região não coberta por baixa qualidade
adaptador	$5'$	36	$13,83 \pm 0,69$	$6,64 \pm 1,11$
	$3'$	0	$0,00 \pm 0,00$	$0,00 \pm 0,00$
	$5' + 3'$	36	$13,83 \pm 0,69$	$6,64 \pm 1,11$
poli-A	$5'$	0	$0,00 \pm 0,00$	$0,00 \pm 0,00$
	$3'$	74	$34,01 \pm 20,71$	$4,50 \pm 2,44$
	$5' + 3'$	74	$34,01 \pm 20,71$	$4,50 \pm 2,44$
vetor	$5'$	114	$54,20 \pm 12,30$	$9,04 \pm 5,60$
	$3'$	85	$86,93 \pm 55,63$	$7,32 \pm 6,22$
	$5' + 3'$	199	$68,18 \pm 40,87$	$8,31 \pm 5,94$
todos	$5'$	150	$44,51 \pm 20,30$	$8,47 \pm 5,02$
	$3'$	159	$62,30 \pm 50,50$	$6,01 \pm 5,04$
	$5' + 3'$	309	$53,67 \pm 39,90$	$7,20 \pm 5,18$

Tabela 3.5: Número de ocorrências, tamanho médio dos artefatos e tamanho médio da região não coberta pela baixa qualidade dos artefatos que apresentaram intersecção com as regiões de baixa qualidade e cujos tamanhos da região não coberta eram menores que o mínimo necessário para identificação dos artefatos (10 para poli-A e adaptador, e 20 para vetor).

conjunto de 12.520 sequências que foram analisadas em buscas destes tipos de artefatos. Esta porcentagem é bastante significativa. Ela indica que um grande número de trechos de artefatos, existentes em seqüências processadas por procedimentos que possuem etapas dependentes entre si, podem ser encaminhados para a fase de clusterização, prejudicando, desta maneira, a análise final dos dados produzidos no projeto.

3.4 Discussão dos resultados

O conjunto de procedimentos proposto neste capítulo mostrou bons resultados. Apesar de sua nova abordagem e da simplicidade de algumas etapas, ele foi capaz de identificar todos os tipos de artefatos.

O teste do procedimento de remoção de seqüências ribossomais mostrou que o método desenvolvido por Telles e da Silva é aplicável também em mamíferos.

As seqüências que nós utilizamos são seqüências que haviam sido selecionadas para clusterização. Isso significa que elas já haviam sido aprovadas no processo de limpeza de seqüências do projeto e que, portanto, deviam possuir boa qualidade.

De fato, as seqüências posssuíam boa qualidade e isso não permitiu avaliar a eficiência do método de detecção de artefatos de baixa qualidade no que se refere à eliminação de

seqüências. No entanto, observamos que artefatos deste tipo foram encontrados em ambas as extremidades de todas as seqüências.

As etapas de detecção de vetores e caudas poli-A/T apesar de simples, mostraram capacidade de encontrar com precisão estes tipos de artefatos.

A análise mais detalhada dos resultados da etapa de identificação de adaptadores mostrou que o método é correto. Porém, é importante verificar que o que o critério *tamanho do adaptador* − 4 não é fixo. Se o adaptador for menor, deve-se determinar um número menor do que 4 para a subtração para diminuir a ocorrência de falsos positivos. Da mesma forma, em casos de adaptadores maiores, o número poderá ser aumentado para evitar a ocorrência de falsos negativos.

Comparando o conjunto final de seqüências processadas pelos nossos métodos e o conjunto de seqüências processadas pelo projeto podemos ver que o tamanho médio de nossas seqüências são maiores. O efeito desta diferença de tamanho pode ser visualizado nos clusterings. Tanto o `Clustering II` como o `Clustering III` agrupam mais seqüências que o `Clustering I`, mesmo trabalhando com um conjunto menor de seqüências. Como as seqüências processadas pelos nossos métodos são maiores, a probabilidade de ocorrência de sobreposições aumenta, promovendo um maior índice de clusterização.

A diferença observada entre os `Clustering II` e `Clustering III` é pequena. Como o `Clustering III` foi criado com dados de qualidades das seqüências e a média delas é alta, o CAP3 não permite que algumas seqüências sejam agrupadas da mesma maneira que no `Clustering II`.

Uma característica positiva de nosso método foi a capacidade de remover as seqüências ribossomais e evitar o agrupamento delas com outras seqüências não ribossomais como ocorreu no `Clustering I`.

O desempenho da nova estratégia, objetivo principal do estudo deste capítulo, foi comprovado através da simulação de procedimentos de detecção de artefatos que realizam a remoção de baixa qualidade antes da identificação de outros tipos de artefatos. Na análise, nós observamos que em $2,46\%$ das seqüências existiam trechos de artefatos que não seriam detectados devido à intersecção com a baixa qualidade. No caso do nosso procedimento, este problema não ocorre.

Uma observação importante a ser feita sobre a análise de desempenho da nova estratégia, é que ela foi feita em cima de artefatos identificados por etapas desenvolvidas para serem executadas independentemente. Contudo, muitos procedimentos de detecção de artefatos operam de forma a levar em consideração o resultado da etapa anterior, o que pode aumentar o porcentual de falsos negativos. Por exemplo, pode-se exigir uma distância máxima da extremidade da seqüência para detecção de um poli-A após a identificação do vetor. Neste caso, se o vetor não for encontrado completamente, a distância do poli-A até a extremidade, ou seja, até o vetor, será maior que a máxima e o poli-A

não será descartado.

Como mencionado anteriormente, as etapas foram desenvolvidas para serem simples de forma a permitirem a identificação dos tipos de artefatos que necessitam uma maior especialização no método. Todos as etapas foram capazes de identificar os seus artefatos com precisão neste conjunto de testes. Contudo, isto não significa que estas etapas não necessitem de mais estudos para aperfeiçoamento.

A detecção de baixa qualidade, por exemplo, não pode ser avaliada adequadamente pois as seqüências utilizadas eram todas de boa qualidade. Assim, decidimos dedicar um estudo mais aprofundado sobre este tipo de artefato. Este estudo está descrito no Capítulo 5.

Um artefato não foi estudado neste capítulo. Trata-se da derrapagem que, inclusive, é pouco mencionada na literatura. Assim, com o objetivo de completar o nosso conjunto de procedimentos e de aprofundar o conhecimento de métodos para detecção de derrapagem, decidimos realizar um estudo mais detalhado, que será descrito no Capítulo 4.

O estudo realizado neste capítulo utilizou um pequeno conjunto de ESTs. Porém, para a real validação dos métodos é importante submetê-los a um volume maior de dados. Os estudos dos próximos capítulos utilizam os dados do projeto SUCEST, composto por 291.689 seqüências. No Capítulo 6, estes dados da cana-de-açúcar são utilizados para validar o nosso conjunto final de procedimentos fruto da combinação dos resultados deste e dos próximos capítulos.

Capítulo 4

Derrapagem

Derrapagem é um tipo de artefato que pode ocorrer em seqüências ESTs. Este artefato se caracteriza por trechos de seqüências que possuem bases ecoadas (repetidas). Estas bases são resultados da leitura dos sinais, em um cromatograma, que indicam diversos picos para um único nucleotídeo.

No caso dos cDNAs, a origem da derrapagem está geralmente associada à presença de longas cadeias de As ou Ts que apresentam problemas durante as reações de seqüenciamento de DNA [6]. As longas cadeias de As ou Ts podem não parear corretamente durante a reação de polimerização e isso gera fragmentos homopoliméricos de diferentes tamanhos que são seguidos pela mesma seqüência de bases.

O ruído de fundo existente nos sinais dos trechos derrapados é grande, no entanto, os sinais das bases são fortes o suficiente para que programas de base-calling atribuam a elas altos valores de qualidade. Tal característica impede que estas regiões sejam identificadas por métodos de detecção de baixa-qualidade.

Devido a estas características, realizamos um estudo e propomos diferentes métodos para remoção deste tipo de artefato. O resultado deste trabalho foi apresentado como pôster no congresso "X-Meeting 2005", realizado em Outubro de 2005, em Caxambu – MG, sob o título "Analysis of slipped sequences in EST projects" e posteriormente aceito como artigo de mesmo título para publicação na revista "Genetics and Molecular Research" [12].

A Seção 4.1 apresentará os métodos de detecção e remoção de derrapagem conhecidos. A Seção 4.2 apresentará os métodos que nós propomos e avaliamos através dos testes descritos na Seção 4.3. Finalmente, a Seção 4.4 discutirá os resultados observados nos testes.

4.1 Métodos de detecção e remoção de derrapagem existentes

Durante a nossa pesquisa, observamos que apenas o trabalho de Telles e da Silva trata de artefatos de derrapagem [104].

O método proposto por eles define uma região ecoada com sendo uma região composta por pelo menos 5 bases consecutivas idênticas. Com base nos tamanhos de todas as regiões ecoadas econtradas na seqüência, o método calcula o produto dos tamanhos da seguinte forma: se o tamanho da região for maior ou igual a 10, ela contribui com 10 no produto, caso contrário, ela contribui com o seu próprio tamanho. O método considerará a seqüência derrapada se a soma dos tamanhos das regiões ecoadas for maior ou igual a 20% do tamanho da seqüência e se o produto calculado for maior ou igual a 10^8.

Uma vez que a seqüência é selecionada com os critérios acima, este método realiza um passo adicional para determinar qual porção da seqüência será removida. Este passo consiste na busca por poli-T ou poli-A. Se um poli-T foi encontrado, toda seqüência é removida, já que o poli-T costuma ser encontrado na extremidade $5'$ e toda a seqüência pode estar afetada. Se um poli-A é encontrado, ele é removido junto com a porção final da seqüência, pois o poli-A costuma ser encontrado na extermidade $3'$ e, provavelmente, apenas esta região está afetada pela derrapagem. Se nenhuma cauda poli-A/T é encontrada, toda a seqüência é removida.

O método acima impõem uma cobertura mínima da seqüência de 20% e isto pode ser um problema conforme os tamanhos das seqüências aumentam. Por exemplo, se uma seqüência possuir um tamanho de 600 bases e a região derrapada tiver tamanho menor que 120 bases, a identificação deste artefato não acontecerá segundo estes critérios. Além disso, observamos que o método não exige proximidade entre as regiões ecoadas.

4.2 Métodos de deteção e remoção de derrapagem propostos

Com o objetivo de melhorar a detecção e remoção de regiões derrapadas, desenvolvemos três novos métodos. Estes métodos são simples e possuem duas estratégias para a busca de trechos derrapados nas seqüências:

Sufixo Esta estratégia baseia-se na idéia de que a derrapagem tende a comprometer, a partir de seu ponto de início, toda a porção final da seqüência. O processamento é feito a partir do final da seqüência e trabalha de forma a identificar o maior sufixo que possua uma pontuação que seja maior ou igual ao valor de corte estipulado para o método.

Subseqüência Ao contrário da estratégia anterior, esta considera que a derrapagem possui pontos de início e fim bem definidos e que, portanto, é possível identificá-la completamente, permitindo que o restante da seqüência seja preservado para outras análises. A estratégia aqui é identificar todas as subseqüências que possuam pontuação maior ou igual ao valor de corte definido para o método. Caso duas subseqüências que se sobrepõem possuam pontuações que atendam ao valor de corte definido, a seqüência formada pela união das duas será considerada derrapada mesmo que a pontuação dela seja menor que a do valor de corte.

Além das duas estratégias na forma de processamento, os métodos possuem dois parâmetros em comum. Seja grupo um conjunto de uma ou mais bases idênticas consecutivas, definimos os seguintes parâmetros:

minimum_echo_size Define o tamanho mínimo que o grupo deve ter para ser considerado um grupo ecoado.

minimum_number_of_echoes Define o número mínimo de grupos ecoados que devem estar presentes na região avaliada para que ela seja considerada na análise.

Outro detalhe importante a ser destacado é que todos os métodos consideram como grupos ecoados válidos aqueles que são compostos pelas bases A, T, C ou G. Grupos compostos por Ns não são considerados como ecoados por que são artefatos de baixa qualidade e as suas presenças podem influenciar negativamente os resultados dos cálculos, de modo a acusarem artefatos de derrapagem onde não existem.

4.2.1 Método 1 - Média Aritmética

Este método realiza o cálculo da razão existente entre a soma dos tamanhos dos grupos ecoados e o número de grupos existentes dentro de uma região.

Supondo que a seqüência esteja sendo processada segundo a estratégia *sufixo* e que os parâmetros `minimum_echo_size` e `minimum_number_of_echoes` possuem os valores 4 e 3, respectivamente, a seqüência

A	T	C	G	TTTTTT	AAAAA	CCC	GGGGG	TT	CCC	AAAA	TT
1	1	1	1	6	5	3	5	2	3	4	2

terá os seguintes sufixos

AAAAACCCGGGGGTTCCCAAAATT	$(5+5+4)/7 = 2,00$
TTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6+5+5+4)/8 = 2,50$
GTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6+5+5+4)/9 = 2,22$
CGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6+5+5+4)/10 = 2,00$
TCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6+5+5+4)/11 = 1,81$
ATCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6+5+5+4)/12 = 1,67$

Neste caso, o melhor sufixo apresentou o valor $2,50$. Se esta mesma seqüência for analisada pela estratégia *subseqüência*, a região

$$\text{TTTTTTAAAAACCCGGGGGTTCCCAAAA } (6+5+5+4)/7 = 2,86$$

seria identificada como a melhor subseqüência.

4.2.2 Método 2 - Média Geométrica

O método da média geométrica é muito semelhante ao anterior. A diferença está no fato que o cálculo é feito com base no produto dos tamanhos dos grupos ecoados elevado ao inverso do número de grupos existentes.

Assim, os sufixos calculados, para a mesma seqüência do exemplo acima, seriam

AAAAACCCGGGGGTTCCCAAAATT	$(5*5*4)^{1/7} = 1,93$
TTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6*5*5*4)^{1/8} = 2,22$
GTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6*5*5*4)^{1/9} = 2,04$
CGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6*5*5*4)^{1/10} = 1,90$
TCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6*5*5*4)^{1/11} = 1,79$
ATCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	$(6*5*5*4)^{1/12} = 1,70$

4.2.3 Método 3 - Cobertura por ecos

Este método realiza uma transformação na seqüência para avaliar a cobertura dos grupos derrapados em relação ao trecho da seqüência que está sendo analisado.

A transformação da seqüência é feita de forma que cada grupo não ecoado é substituído por um 0 e cada grupo ecoado é substituído por um 1. Assim, a seqüência utilizada nos exemplos anteriores seria transformada para

$$000011010010.$$

A partir da transformação, faz-se a razão entre o tamanho do trecho analisado e o número de 1s contidos dentro dele. Portanto, os sufixos avaliados seriam

AAAAACCCGGGGGTTCCCAAAATT	1010010	$3/7 = 0,43$
TTTTTTAAAAACCCGGGGGTTCCCAAAATT	11010010	$4/8 = 0,50$
GTTTTTTAAAAACCCGGGGGTTCCCAAAATT	011010010	$4/9 = 0,44$
CGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	0011010010	$4/10 = 0,40$
TCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	00011010010	$4/11 = 0,36$
ATCGTTTTTTAAAAACCCGGGGGTTCCCAAAATT	000011010010	$4/12 = 0,33$

4.3 Definição dos Valores de Corte

Identificar o sufixo ou subseqüência com maior pontuação existente na seqüência não é, necessariamente, identificar todo o trecho derrapado pois, conforme a distribuição dos grupos ecoados, o valor calculado para a região derrapada pode ser menor que o de um trecho contido nela.

Assim, cada método precisa de um valor de corte que deve ser utilizado na análise de forma a identificar a maior região que tenha pontuação maior ou igual a esse valor. Este valor deve ser tal que minimize o número de falsos positivos e de falsos negativos.

Para definição dos valores de corte, diversos testes foram realizados. Estes testes utilizaram o conjunto de todos ESTs produzidos no projeto de seqüenciamento de ESTs da cana-de-açúcar - SUCEST [100]. Este conjunto de dados é exatamente o mesmo utilizado no trabalho de Telles e da Silva e é composto por 291.689 ESTs. As seqüências deste conjunto possuem tamanho médio de $829,44 \pm 182,60$ bases e qualidade média de $23,15 \pm 15,71$.

4.3.1 Teste 1 - Tamanhos dos Grupos Ecoados

O primeiro passo de nossos testes foi realizar a execução de cada uma das duas estratégias dos três métodos utilizando valores para o parâmetro `minimum_echo_size` variando no intervalo $\{1, 2, \ldots, 10\}$. O valor do parâmetro `minimum_number_of_echoes` foi fixado em 8 para facilitar a análise. Além disso, este é o número minimo de grupos ecoados que uma seqüência deve ter para que seja possível atingir o valor 10^8 no método proposto por Telles e da Silva. Acreditamos que este valor é o suficiente para evitar que sufixos muito pequenos prejudiquem a avaliação e evitar que sufixos relevantes sejam perdidos.

Para cada seqüência do conjunto foram executados todos os pares método/estratégia. Os valores máximos para os sufixos e as subseqüências foram anotados para cada uma delas, formando uma lista de valores para cada par método/estratégia.

Devido ao grande volume de dados, cada lista de valores de cada execução foi ordenada em ordem crescente e dividida em intervalos de 100 seqüências. Para cada intervalo foi calculada a média dos valores. As Figuras 4.1, 4.2 e 4.3 ilustram os gráficos produzidos com os resultados da estratégia *sufixo* e as Figuras 4.4, 4.5 e 4.6 ilustram os gráficos produzidos com os resultados da estratégia *subseqüência*.

Como podemos observar nos gráficos das Figuras 4.1, 4.2, 4.3, 4.4, 4.5 e 4.6 os resultados dos métodos produzidos com a estratégia *sufixo* são muito semelhantes aos produzidos com a estratégia *subseqüência* apesar dos valores de pontuação apresentados para esta última serem maiores em função da diminuição do comprimento das regiões identificadas.

Podemos ver nestes gráficos que todos possuem comportamento semelhante conforme os valores do parâmetro `minimum_echo_size` são alterados. Para cada valor utilizado, a

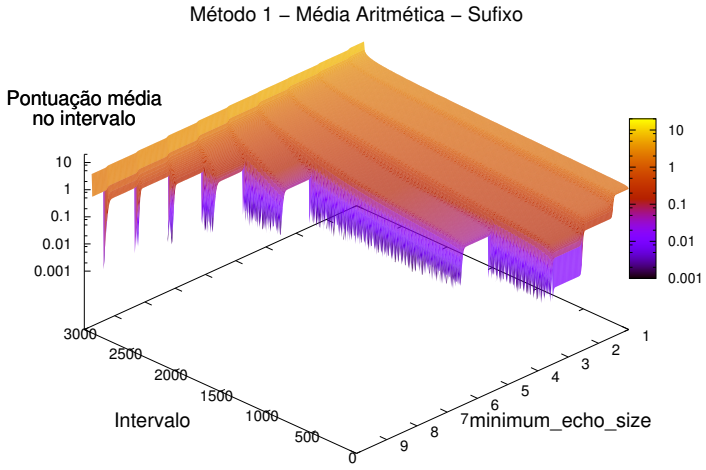

Figura 4.1: Método 1 executado com a utilização da estratégia *sufixo* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

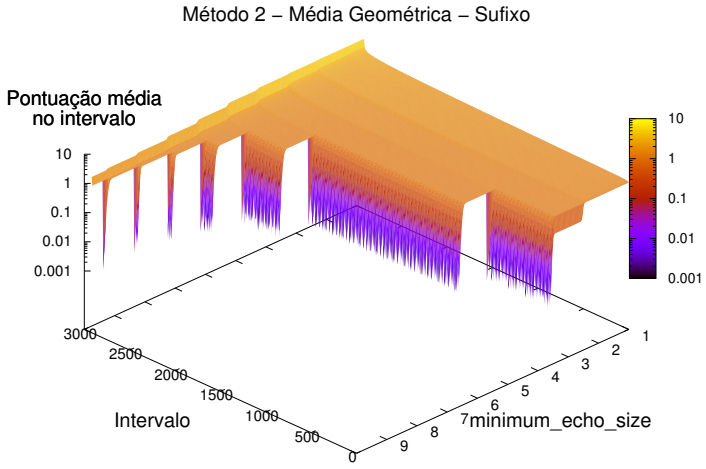

Figura 4.2: Método 2 executado com a utilização da estratégia *sufixo* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

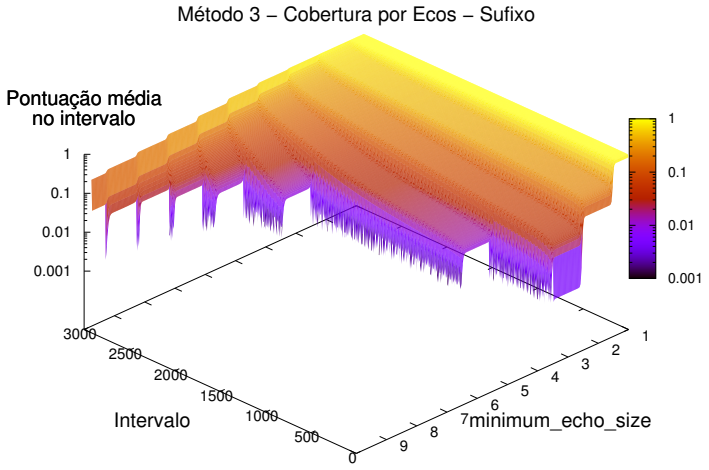

Figura 4.3: Método 3 executado com a utilização da estratégia *sufixo* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

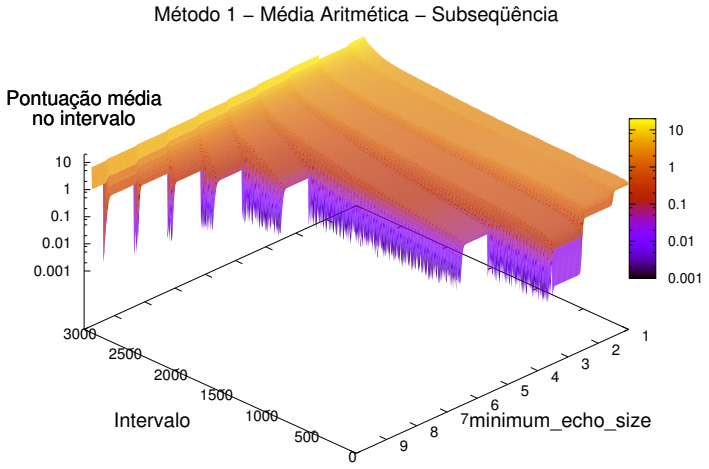

Figura 4.4: Método 1 executado com o uso da estratégia *subseqüência* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

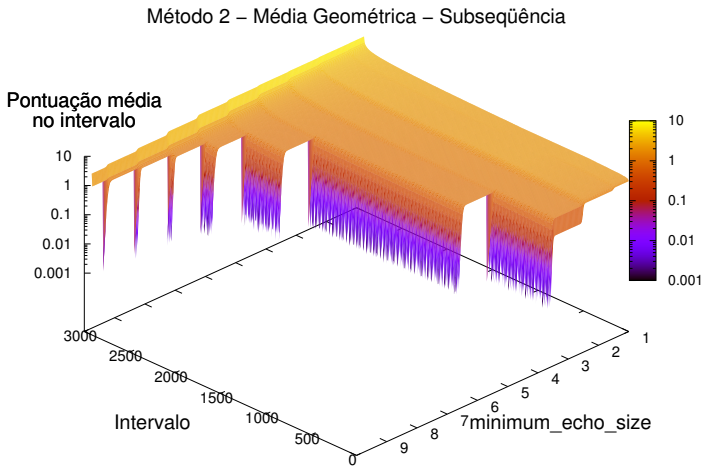

Figura 4.5: Método 2 executado com o uso da estratégia *subseqüência* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

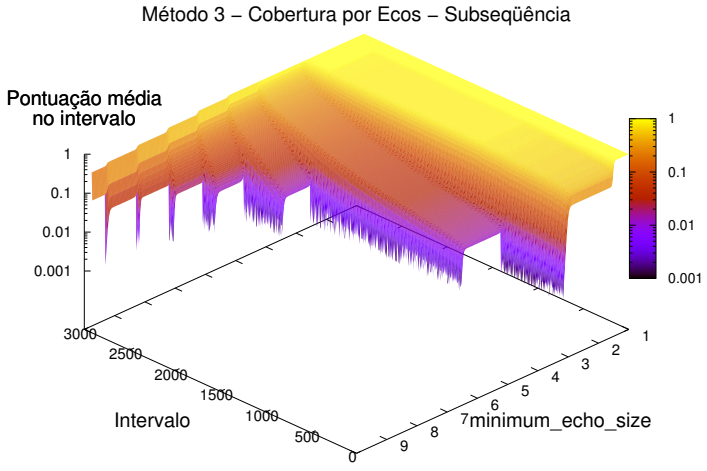

Figura 4.6: Método 3 executado com o uso da estratégia *subseqüência* e parâmetros $minimum\_number\_of\_echoes = 8$ e $minimum\_echo\_size = \{1, 2, \ldots, 10\}$. Os resultados de cada execução foram ordenados em ordem crescente e divididos em intervalos de 100 seqüências. Cada intervalo teve o valor de pontuação média calculado e este gráfico mostra o comportamento destes intervalos em cada uma das execuções. Note que o eixo das pontuações médias está em escala logarítmica.

faixa onde os intervalos possuem pontuação média igual a zero é a mesma. Uma pontuação média igual a zero significa que dentro do intervalo não existe nenhuma seqüência que possua o número mínimo de grupos ecoados (no caso 8) com tamanho determinado pelo parâmetro *minimum_echo_size*. Se uma seqüências não possui um sufixo que tenha o número mínimo de grupos ecoados, ela também não tem uma subseqüência, e vice-versa. Por exemplo, se observamos os resultados produzidos pelo método 3 com `minimum_echo_size` = 5, veremos que apenas 618 intervalos possuem pontuação média maior que zero.

Pontuações mais altas indicam maior probabilidade da seqüência estar derrapada. Ao analisarmos os comportamentos dos valores calculados ao longo dos intervalos, podemos ver que todos possuem uma pequeno número de intervalos que se destacam pelos altos valores atingidos.

O comportamento dos valores ao longo dos intervalos tende a ser mais conservado no método 2. O gráfico exibe pouca variação denotando que este método pode ser mais difícil de calibrar pois pequenas mudanças no valor de corte podem resultar na inclusão ou eliminação de uma grande quantidade de seqüências. Os outros dois métodos possuem uma variação maior ao longo dos intervalos.

Para confirmar estas observações, nós fixamos o valor `minimum_echo_size` em 5 e escolhemos valores de cortes diferentes para cada um dos pares método/estratégia. Os valores foram escolhidos de maneira que o número de seqüências com regiões que tivessem pontuações maiores que a definida pelo valor de corte fosse aproximadamente o mesmo para todos os pares (em torno de 7.000 seqüências). Estes valores de corte foram definidos como valores base de cada par método/estratégia.

Nós definimos um novo conjunto de valores de corte, adicionando a cada valor base -15%, -10%, -5%, -2%, -1%, $+1\%$, $+2\%$, $+5\%$, $+10\%$ e $+15\%$ de seu valor. Feito isso nós contamos o número de seqüências com pontuação maior que os valores de corte para cada par método/estratégia e construímos o gráfico da Figura 4.7.

No gráfico da Figura 4.7, podemos observar que a medida que se diminui o valor de corte, todos os pares método/estratégia mostram um aumento do número de seqüências derrapadas. Contudo, observamos que o método 2, independente da estratégia utilizada, é o que apresenta maior variação entre faixas de valores diferentes, confirmando que este método pode ser mais difícil de calibrar.

Observando os métodos 1 e 3, notamos um comportamento bem similar, sendo que o método 1 apresenta menor variação entre valores de cortes diferentes, o que indica que ele é o menos sensível à mudança realizadas nos valores de corte.

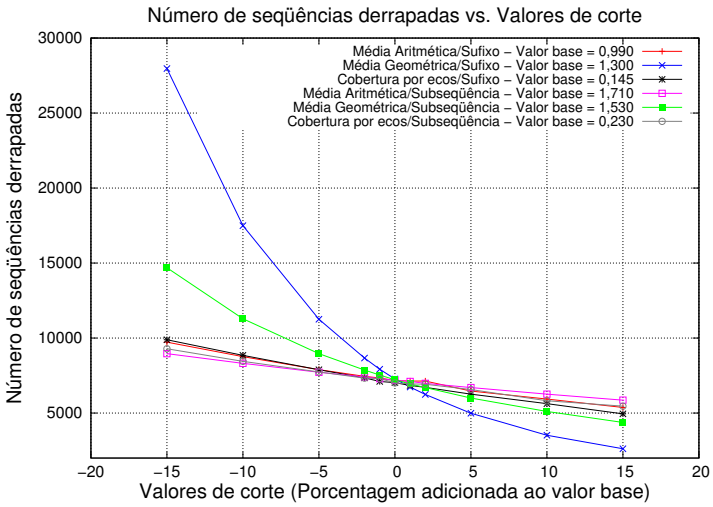

Figura 4.7: Número de seqüências derrapadas detectadas por cada par método/estratégia conforme variação dos valores de corte. Um valor de corte base foi definido para cada par de maneira que todos apontassem, aproximadamente, o mesmo número de seqüências derrapadas (em torno de 7.000 seqüências). A estes valores de corte foram adicionados -15%, -10%, -5%, -2%, -1%, $+1\%$, $+2\%$, $+5\%$, $+10\%$ e $+15\%$ de seus valores e o número de seqüências indicadas como derrapadas por cada um dos pares, nestes valores, foram plotados no gráfico.

4.3.2 Teste 2 - Comparações entre Métodos

O segundo teste que realizamos foi a comparação dos resultados produzidos pelos métodos propostos e pelo Método de Telles e da Silva com o objetivo de se avaliar a capacidade de detecção de seqüências derrapadas de cada um deles.

Implementamos o método de Telles e da Silva (método 4) segundo a descrição contida no trabalho escrito por eles. Esta implementação foi utilizada para processamento do mesmo conjunto de dados e 7.213 seqüências foram marcadas como derrapadas. Note que o processamento foi feito com seqüências completas e não com seqüências que já haviam passado por processos de deteção e remoção de vetor e de qualidade.

Para realizar a comparação decidimos utilizar os resultados de todos os métodos executados com o valor 5 para o parâmetro `minimum_echo_size`. Este foi o mesmo valor adotado por Telles e da Silva e parece ser mais o indicado pois não restringe a detecção de derrapagens que não possuam grupos ecoados muito grandes.

Para cada uma das duas estratégias dos métodos propostos, foram separadas as 7.213 seqüências que apresentaram regiões com maiores pontuações. Esta operação seria o equivalente a definir para os métodos 1, 2 e 3 os valores de corte $0,9860$, $1,3010$ e $0,1429$ para a estratégia *sufixo* e $1,7070$, $1,5306$ e $0,2286$ para a estratégia *subseqüência*.

Os diagramas de Venn-Euler exibidos nas Figuras 4.8 e 4.9 foram construídos e indicam as intersecções entre os conjuntos das seqüências listadas por cada método e cada uma das estratégias.

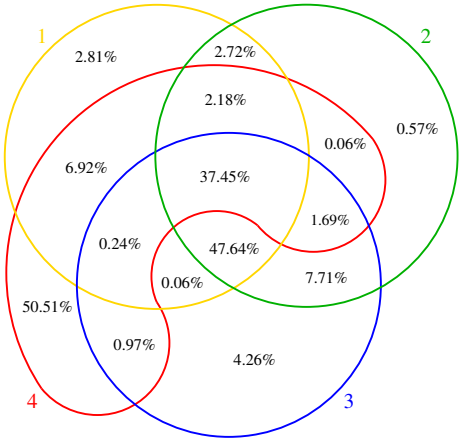

Figura 4.8: Diagrama de Venn-Euler das intersecções entre os conjuntos de seqüências produzidas pelos métodos 1, 2 e 3 estratégia *sufixo* e o método 4.

Analisando os diagramas de Venn-Euler exibidos nas Figuras 4.8 e 4.9 é possível observar que a intersecção dos três métodos propostos é maior na estratégia *sufixo* ($85,09\%$) do que na estratégia *subseqüência* ($81,96\%$). Isso acontece porque a estratégia *sufixo* se

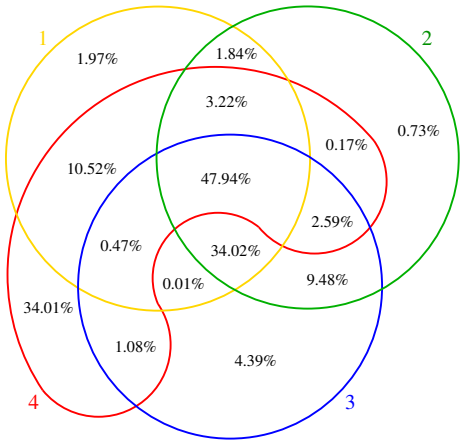

Figura 4.9: Diagrama de Venn-Euler das intersecções entre os conjuntos de seqüências produzidas pelos métodos 1, 2 e 3 estratégia *subseqüência* e o método 4.

mostra incapaz de encontrar derrapagens no início da seqüência pois, nestes casos, os métodos produzem baixa pontuação devido à extremidade final não derrapada. Assim, os três métodos acabam por ficar ancorados nos mesmos sufixos.

Outro fato interessante, quando comparamos os três métodos nestes diagramas, é que o método 2 é praticamente todo envolvido pela união dos conjuntos dos métodos 1 e 3. Menos de $1,00\%$ das seqüências de seu conjunto foram identificadas somente por ele. Este subconjunto é muito menor do que aqueles identificados unicamente pelos métodos 1 e 3, respectivamente, em torno de $11,00\%$ e $5,00\%$.

A comparação entre os resultados das duas estratégias dos três métodos e os resultados do método de Telles e da Silva mostra que a intersecção dos quatro conjuntos é menor na estratégia *sufixo* $(37,45\%)$ que na estratégia *subseqüência* $(47,49\%)$. O motivo desta maior aproximação da estratégia *subseqüência* em relação ao método 4 está no fato de ele não ser ancorado ao fim da seqüência como ocorre com a estratégia *sufixo*. Como a estratégia *subseqüência* tem mais flexiblidade de encontrar as regiões derrapadas, é natural que ela apresente mais coincidências com o método 4 que também não está preso ao sufixo da seqüência.

As intersecções dos conjuntos de resultados das estratégias *sufixo* e *subseqüência*, para cada um dos métodos, envolvem $68,99\%$, $68,89\%$ e $68,24\%$ das seqüências, respectivamente, identificadas pelos métodos 1, 2 e 3. Assim, temos que aproximadamente $31,00\%$ das seqüências dos resultados de uma estratégia não estão presentes nos resultados da outra, e vice-versa.

4.3.3 Teste 3 - Comparação entre as estratégias *sufixo* e *subseqüência*

As listas das 7.213 melhores seqüências de cada estratégia também foram utilizadas no terceiro passo dos testes. Neste passo foi feita a comparação de resultados entre os pares de estratégias de cada método proposto. As intersecções entres os resultados dos pares de estratégias dos métodos 1, 2 e 3 tinham tamanhos 4.976, 4.969 e 4.922, respectivamente.

Além disso, para cada uma das listas ordenamos os resultados em ordem decrescente de pontuação e dividimos em intervalos de tamanho 200. Para cada um dos intervalos, da lista de resultados de uma estratégia de um método, contamos o número de seqüências dentro dele que não estavam presentes na lista completa da outra estratégia. O gráfico da Figura 4.10 foi construído a partir destes dados.

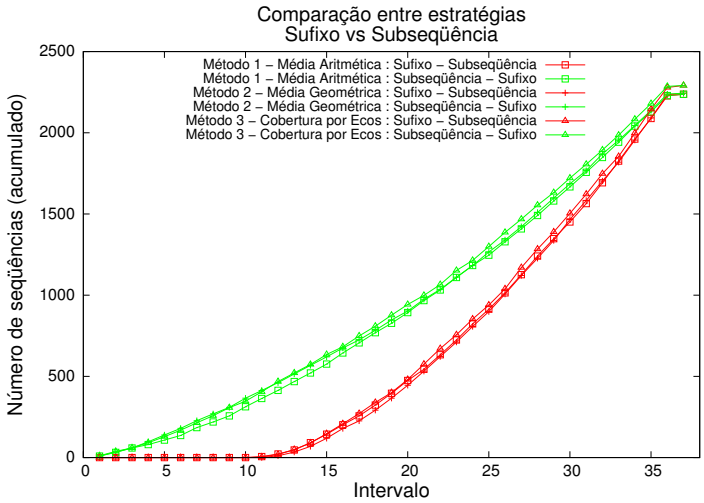

Figura 4.10: Gráfico de comparação entre os pares de estratégias dos métodos 1, 2 e 3. Cada lista de resultado foi ordenada em ordem decrescente, assim, os primeiros intervalos possuem as melhores seqüências de cada método/estratégia. As linhas verdes representam o número acumulado de seqüências que estão dentro do intervalo dos resultados da estratégia *subseqüência* e não estão no conjunto de resultados da estratégia *sufixo*. As linhas vermelhas representam o número acumulado de seqüências que estão dentro do intervalo dos resultados da estratégia *sufixo* e não estão no conjunto de resultados da estratégia *subseqüência*.

A análise do gráfico da Figura 4.10 nos mostra que a estratégia *subseqüência* tende a manter os melhores resultados obtidos pelo estratégia *sufixo*. É possível ver no gráfico que as primeiras 2.000 seqüências com maior pontuação segundo a estratégia *sufixo* são encontradas pela estratégia *subseqüência* (Sufixo - Subseqüência), enquanto na situação

oposta isso não é observado (Subseqüência - Sufixo). Isso ocorre em razão dos mesmos motivos apresentados anteriormente: os melhores sufixos podem ser identificados pela estratégia *subseqüência*, mas boas subseqüências no começo da seqüência quase sempre são perdidas pelo método de sufixo.

4.3.4 Teste 4 - BLAST das seqüências derrapadas

O último teste que realizamos foi a comparação das seqüências apontadas como derrapadas, pelas estratégias dos métodos propostos e pelo método de Telles e da Silva, com as seqüências de um banco público para verificar a influência da remoção de derrapagem na detecção de genes. Neste teste utilizamos a versão 2.2.11 do software BLAST e a versão 46.6 do banco swissprot. Este banco foi utilizado por ser um banco muito bem curado.

A comparação foi feita com a utilização dos conjuntos de 7.213 seqüências apontadas como derrapadas por cada um dos métodos. Estes conjuntos foram testados de três modos diferentes:

i) Seqüências completas, sem qualquer mascaramento.

ii) Seqüências completas, contendo os trechos de vetor mascarados, para avaliação do número de seqüências que, no modo i, apresentaram hits devido à presença de vetor.

iii) Apenas a maior subseqüência que não tenha sido mascarada devido à presença de derrapagem ou de vetor. Seqüências com tamanho menor do que 100 foram descartadas.

O mascaramento de vetor foi feito com a utilização do software cross_match (versão 0.990319) usando os parâmetros `-minmatch` 12, `-minscore` 20 e `-penalty` -2.

Os trechos derrapados utilizados no mascaramento das seqüências no modo iii foram determinados segundo os valores de corte utilizados para criação das listas de seqüências derrapadas (Subseção 4.3.2). Note que as derrapagens identificadas aqui não são as regiões com maiores pontuações, mas sim as regiões que entraram dentro do valor de corte, conforme indicado nas descrições das duas estratégias de processamento no início da Seção 4.2.

A Tabela 4.1 lista o número de seqüências que exibiram pelo menos um hit com e-value $\leq 10^{-5}$ para cada um dos pares método/estratégia e para o método 4. Ela também mostra a porcentagem de perdas de hits quando comparamos os resultados do modo ii com o modo i e do modo iii com o modo ii.

A execução do BLAST com as seqüências completas (modo i), foi feita com o objetivo de avaliar a quantidade de seqüências detectadas como derrapadas que não apresentavam semelhanças com nenhuma seqüência presente no swissprot. Acreditávamos inicialmente

Método	Estratégia	i	ii	iii	$(1-(ii/i))$	$(1-(iii/ii))$
1	*sufixo*	2.532	2.034	1.946	$19,67\%$	$4,33\%$
2		2.811	2.225	2.166	$20,85\%$	$2,65\%$
3		2.856	2.211	2.157	$22,58\%$	$2,89\%$
1	*subseqüência*	1.938	1.590	1.516	$17,96\%$	$4,65\%$
2		2.198	1.781	1.716	$18,97\%$	$3,65\%$
3		2.287	1.831	1.765	$19,94\%$	$3,60\%$
4	–	1.443	710	85	$50,08\%$	$88,03\%$

Tabela 4.1: Número de seqüências com pelo menos um hit com e-value $\leq 10^{-5}$ contra o banco swissprot em cada um dos conjuntos de 7.213 seqüências marcadas como derrapadas pelos pares método/estratégia propostos e pelo método 4. Cada conjunto foi submetido ao BLAST de três modos diferentes: i - seqüência completa, ii - seqüência com vetor mascarado, e iii - maior subseqüência sem trechos de vetor ou derrapagem. As últimas duas colunas indicam a porcentagem de perda de hits quando comparamos os modos ii e i e os modos iii e ii.

que um menor número de hits permitiria a identificação do método que produz o menor número de falsos positivos, ou seja, o menor número de seqüências que são, de fato, genes e não seqüências derrapadas. Contudo, observamos que uma quantidade considerável de hits apareciam devido aos trechos de vetores presentes nas seqüências.

Diante dos resultados apresentados pelo modo i, decidimos mascarar as seqüências de vetores e executar o BLAST novamente (modo ii).

Observamos que em torno de $20,00\%$ dos hits apresentados nas seqüências marcadas como derrapadas por nossos métodos foram causados por trechos de vetor. No caso do método 4, esta porcentagem girou em torno de $50,00\%$.

Com a estimativa do número de seqüências que apresentavam hits, executamos o BLAST das seqüências preparadas como descrito no modo iii.

Os resultados mostraram que os métodos propostos perderam de $2,65\%$ a $4,65\%$ de hits.

O método 4, por sua vez, perdeu $88,03\%$ dos hits. Isso ocorre devido ao critério de descarte dos trechos derrapados imposto pelo método. Se a seqüência derrapada apresentar uma cauda poli-T ou nenhum tipo de cauda, ela é descartada completamente. Somente se a seqüência apresentar uma cauda poli-A, ela terá algum trecho preservado.

Comparando os nossos pares método/estratégia, é possível ver que o método 1 tende apresentar um número menor de seqüências com hits. Ao mesmo tempo, este método tende a perder mais hits após a realização da remoção da derrapagem. Os métodos 2 e 3 também apresentam estes efeitos, contudo eles são mais semelhantes entre si. Ambos

tendem a apresentar seqüências com mais hits de BLAST e menor perda de hits após a remoção da derrapagem que o método 1.

Comparando as duas estratégias, é possível notar que a estratégia *sufixo* apresenta um número maior de seqüências com hits e que ela tende a perder menos hits após a remoção dos artefatos de derrapagem. Isto pode ser explicado pela natureza dos artefatos de derrapagem produzidos nesta estratégia. Como os artefatos de derrapagem são removidos da extremidade final, eles não produzem fragmentação da seqüências. No caso da estratégia *subseqüência*, os artefatos podem estar no meio da seqüência e no momento da remoção, a maior subseqüência restante pode não conter o hit original.

4.4 Discussão dos Resultados dos Testes

Os resultados produzidos nos permitem concluir que a estratégia *subseqüência* tende a ser melhor que a *sufixo*. A estratégia escolhida encontra as regiões derrapadas com eficiência maior, sendo capaz de encontrar artefatos inclusive na porção inicial da seqüência. Dessa maneira, definimos para esta estratégia os valores de corte $1,90$, $1,60$ e $0,25$ para os métodos 1, 2 e 3, respectivamente.

Estes valores de corte são mais restritivos e diminuem o conjunto de seqüências derrapadas anterior em cerca de 15%. Durante a definição destes valores de corte verificamos que o observado nos gráficos de superfície se confirmou. O método 1 é o mais fácil e o método 2 é o mais difícil de se calibrar.

A escolha de um valor mais restritivo deve-se principalmente a estratégia de detecção e remoção de artefatos de nosso estudo como um todo. A nossa abordagem se preocupa em identificar os artefatos independetemente uns dos outros, podendo haver inclusive sobreposições. Isto gera fragmentação da seqüência, o que não é de fato um problema porque apenas a maior subseqüência será mantida ao final do processo de remoção de artefatos.

Acreditamos, portanto, que a combinação dos artefatos será suficiente para produzir uma boa limpeza das seqüências e que o valor mais restritivo evitará a geração de falsos positivos na detecção de derrapagem.

O método 3 apresentou resultados que indicam que sua metodologia parece ser capaz de delimitar a região derrapada com mais precisão do que as outras estratégias.

O método 2, como já dito anteriormente, tende a deixar os valores muito próximos devido à fórmula de cálculo dos valores e, portanto, é mais difícil de ser calibrado.

Apesar do método 3 ser considerado melhor, o método 1 ainda merece atenção. Neste tipo de estratégia talvez seja viável a utilização de valores menores para o parâmetro `minimum_echo_size`.

Além disso, a variação do parâmetro `minimum_number_of_echoes` pode ser melhor

estudada. Nós utilizamos o valor 8 para obter uma melhor comparação com o método de Telles e da Silva, utilizado no projeto SUCEST. Contudo, isso não indica que este é necessariamente o melhor valor. Esta análise poderá ser executada em um trabalho futuro.

Outra possibilidade interessante, para a execução de um trabalho futuro, é o desenvolvimento de métodos que realizem a verificação dos artefatos identificados e confirmem ou não a ocorrência de derrapagem. Através da atribuição de pontuações que indiquem o grau de confiabilidade da identificação, este métodos refinariam o processo de detecção e remoção de trechos derrapados.

Finalmente, outra possível extensão deste trabalho é a aplicação dos métodos desenvolvidos em seqüências de outros organismos para validação e refinamento dos procedimentos aqui propostos.

Capítulo 5

Baixa Qualidade

Neste capítulo apresentaremos o estudo realizado para definição de um método de identificação de trechos de baixa qualidade mais adequado ao nosso conjunto de procedimentos de deteção e remoção de artefatos.

Nosso estudo se concentrou em dois algoritmos diferentes, que tiveram seus parâmetros variados para que pudéssemos encontrar a combinação que produz melhores resultados.

A Seção 5.1 apresentará o algoritmo de janela deslizante. A Seção 5.2 apresentará o algoritmo de subseqüência máxima. A Seção 5.3 apresentará os procedimentos de deteção de baixa qualidade executados pelo prgograma LUCY. Finalmente, a Seção 5.4 apresentará e discutirá os testes que realizamos para determinar qual a melhor opção entre os dois algoritmos.

Resultados obtidos neste capítulo foram apresentados no congresso "X-Meeting 2006", realizado em Agosto de 2006, em Fortaleza – CE, através de um pôster entitulado "Low quality trimming on SUCEST ESTs".

5.1 Janela deslizante

O algoritmo de janela deslizante é utilizado em muitos projetos de seqüenciamento de ESTs.

Este algoritmo é linear no tamanho da seqüência e consiste na utilização de uma janela que cobre um determinado número de bases e percorre a seqüência de uma extremidade a outra, base a base, até encontrar um conjunto de nucleotídeos que atendam a um determinado critério de qualidade. Quando este conjunto é encontrado, todas as bases anteriores a este conjunto (bases que foram percorridas e não se encontram dentro da janela) são descartadas como artefatos de baixa qualidade.

Este algoritmo é utilizado por muitos projetos devido a sua simplicidade. O tamanho da janela e os critérios de qualidade podem variar muito. Além disso, existem projetos

que analisam as seqüências nas duas direções e outros que analisam apenas a extremidade $3'$, que normalmente é a que possui mais erros.

Para esse trabalho, implementamos a versão do algoritmo utilizada no projeto SUCEST [104]. Esta versão percorre a seqüência nas duas direções e possui três parâmetros:

- `window_size`: Define o tamanho da janela em número de bases.
- `quality_threshold`: Define o valor mínimo de qualidade que uma base deve ter para ser considerada de boa qualidade.
- `bad_bases_threshold`: Define o número máximo de bases, com qualidade menor do que a definida no parâmetro `quality_threshold`, que podem existir dentro da janela.

Durante o desenvolvimento do conjunto de procedimentos de deteção e remoção de artefatos do projeto SUCEST, Telles e da Silva testaram algumas combinações de parâmetros. Após algumas análises, decidiram optar pela utilização dos valores 20, 10 e 12 para os parâmetros `window_size`, `quality_threshold` e `bad_bases_threshold`, respectivamente.

5.2 Subseqüência máxima

O problema da subseqüência máxima consiste em identificar dentro de um seqüência de números reais a subseqüência (subcadeia contígua) que possui soma máxima dentre todas as subseqüências possíveis. Existem algoritmos lineares no tamanho da seqüência para a resolução deste problema.

No caso da detecção de baixa qualidade, este algoritmo pode ser utilizado para maximizar a qualidade e, desta forma, minimizar a probabilidade de erro das bases.

O software de base-calling phred oferece a opção de executar a identificação das regiões de baixa qualidade das seqüências que ele processa utilizando este algoritmo.

Para isso, o phred transforma a seqüência de qualidades das bases em uma seqüência de valores reais v_i segundo a fórmula $v_i = \texttt{trimming\_cutoff} - \texttt{error\_probability}_i$, onde $0 \leq \texttt{trimming\_cutoff} \leq 1$. O valor $\texttt{error\_probability}_i$ é calculado segundo a fórmula $\texttt{error\_probability}_i = 10^{(Q_i/-10)}$, onde Q_i é o valor da qualidade da base i. O parâmetro `trimming_cutoff` indica o valor máximo da probabilidade de erro que uma base deve ter para ser considerada boa.

Nesta transformação, todas as bases que possuem probabilidades de erro maiores do que a representada pelo valor `trimming_cutoff` recebem valores negativos, o que torna possível a utilização do algoritmo de subseqüência máxima.

O phred utiliza o valor $0,05$ para o parâmetro `trimming_cutoff`. Este valor é equivalente, aproximadamente, ao valor de qualidade 13.

5.3 LUCY

LUCY [26] é um programa utilizado pelo grupo TIGR para a realização de limpeza de seqüências. Ele realiza um procedimento mais elaborado para detecção e remoção de baixa qualidade. Diversos passos são executados para que, em cada um deles, sejam eliminados trechos de baixa qualidade com características específicas.

O primeiro passo do algoritmo é a remoção das pontas de baixa qualidade com a utilização do algoritmo de janela deslizante. O software processa a seqüência a partir de uma extremidade em busca da primeira janela que possua tamanho `window_size` e probabilidade de erro média de no máximo max_avg_error. Os valores padrão são 10 e $0,02$, respectivamente, para os parâmetros `window_size` e `max_avg_error`. O processo é repetido na direção oposta, a partir da outra extremidade.

A subseqüência situada entre as pontas de baixa qualidade, encontradas no passo acima, é submetida ao próximo passo, que tem a função de remover regiões com taxas de erros inaceitáveis.

Neste passo, a seqüência pode ser analisada diversas vezes por janelas com diferentes pares de parâmetros $\texttt{window\_size}_i$ e $\texttt{max\_avg\_error}_i$. No caso da configuração padrão do LUCY, são dois pares: $\texttt{window\_size}_1 = 50$ e $\texttt{max\_avg\_error}_1 = 0,08$, e $\texttt{window\_size}_2 = 10$ e $\texttt{max\_avg\_error}_2 = 0,30$. O objetivo do primeiro par é excluir grandes regiões com baixa qualidade e o do segundo par é excluir pequenas regiões com qualidade muito baixa que "escaparam" do processsamento realizado pela primeiro par.

O programa percorre a subseqüência, identificada no primeiro passo, com a janela configurada com o primeiro par de parâmetros. Cada janela que se encaixa no critério é considerada como parte de uma região apta a continuar no processo. Quando a janela encontra uma janela que não atende aos critérios, LUCY separa a região de boa qualidade identificada e continua percorrendo a subseqüência até encontrar uma nova janela que atenda ao critério para que ele possa iniciar uma nova região de boa qualidade.

Feito isso, o programa processa cada uma das regiões identificadas com janelas, configuradas com o segundo par de parâmetros, executando o mesmo processo e produzindo novas regiões de boa qualidade.

Toda região, identificada no processo acima, que tiver tamanho menor que o definido no parâmetro `minimum` (100 é o valor padrão), é eliminada.

Finalmente, cada região de boa qualidade restante será avaliada para verificação dos seguintes critérios: probabilidade média de erro na seqüência toda de no máximo $\texttt{max\_avg\_error}_f$ e probabilidade de erro das duas primeiras bases de cada extremidade de no máximo `max_error_at_ends`. A maior região que atender estes critérios será o resultado final do processo de remoção de baixa qualidade do programa LUCY. Os valores padrão para estes dois parâmetros são $0,025$ e $0,020$, respectivamente.

5.4 Análise dos algoritmos de detecção e remoção de baixa qualidade

Para desenvolver o nosso algoritmo de deteção e remoção de baixa qualidade, decidimos avaliar diversas combinações de parâmetros dos algoritmos "Janela deslizante" e "Subseqüência máxima". Além disso, decidimos utilizar o LUCY, com seus parâmetros padrões, para a realização de comparações, pois ele é o algoritmo utilizado pelo TIGR.

5.4.1 Conjunto de dados utilizados nos testes

Como conjunto de dados para testes, utilizamos os 291.689 ESTs produzidos no projeto SUCEST. Estas seqüências possuem tamanho médio de $829,44 \pm 182,60$ bases. A qualidade média é de $23,15 \pm 15,71$.

O gráfico da Figura 5.1 nos mostra a distribuição de qualidade média ao longo das 1.000 primeiras posições dos ESTs da cana-de-açúcar e o gráfico da Figura 5.2 nos mostra a distribuição da qualidade média ao longo do comprimento das seqüências (em porcentagem).

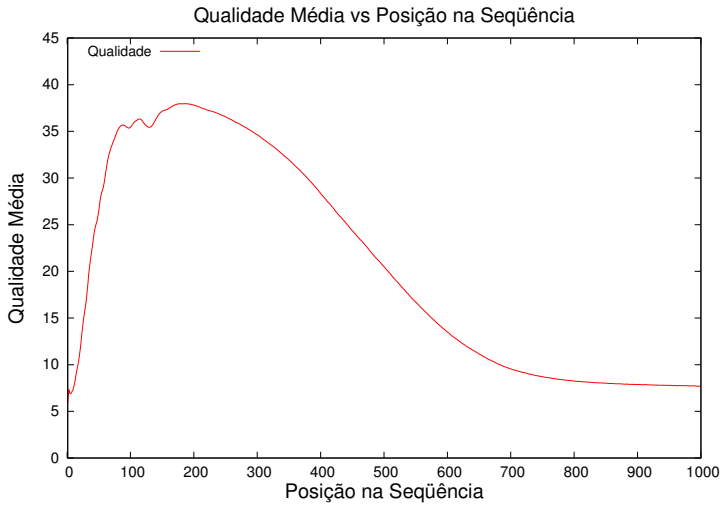

Figura 5.1: Distribuição da qualidade média ao longo das 1.000 primeiras posições dos ESTs do projeto SUCEST.

Analisando os dois gráficos, podemos ver claramente uma característica muito comum em leituras de máquinas de seqüenciamento. A seqüência possui uma pequena região de baixa qualidade no início da seqüência. Após esta região, a qualidade média cresce bastante e depois de um determinado trecho a qualidade cai gerando uma grande extremidade $3'$ de baixa qualidade.

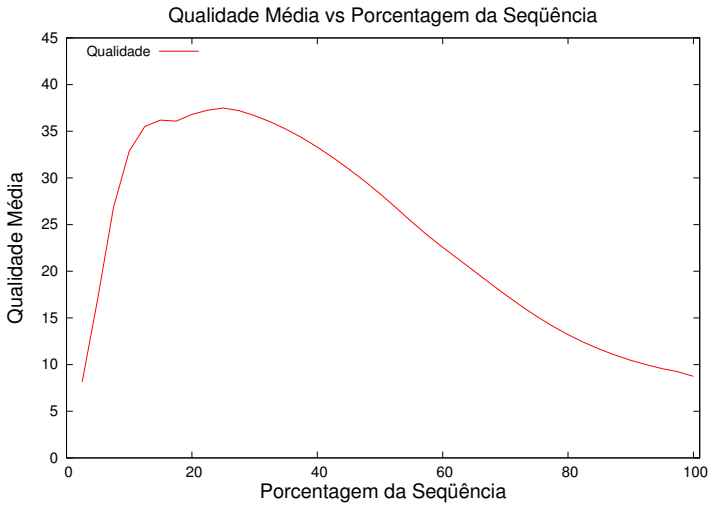

Figura 5.2: Distribuição da qualidade média ao longo do comprimento (em porcentagem) dos ESTs do projeto SUCEST.

5.4.2 Janela deslizante

O algoritmo de janela deslizante foi implementado como descrito anteriormente e utilizado com diversas combinações de parâmetros para processar o conjunto de testes.

Testamos seis tamanhos diferentes de janela variando o parâmetro `window_size` dentro do conjunto $\{5, 10, 15, 20, 25, 30\}$.

Para cada um dos diferentes tamanhos de janela o parâmetro `bad_bases_threshold` recebeu valores dentro do intervalo $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$.

Finalmente, o parâmetro `quality_threshold` foi variado dentro do intervalo $[1, 30]$.

Analisando os dados das diferentes execuções observamos um comportamento similar quando comparamos os diferentes tamanhos de janela. Os gráficos da Figura 5.3 ilustram este comportamente. Podemos ver que a medida que o valor do parâmetro `quality_threshold` cresce, o algoritmo, como era de se esperar, produz maiores artefatos de baixa qualidade, diminuindo, portanto, o tamanho da seqüência de boa qualidade. Olhando para o parâmetro `bad_bases_threshold`, nota-se que quanto menor o seu valor, maiores serão os artefatos de baixa qualidade.

5.4.3 Subseqüência máxima

O algoritmo de subseqüência máxima foi implementado nos mesmos moldes do utilizado pelo software phred. Contudo, para tornar as análises mais intuitivas, transformamos o parâmetro `trimming_cutoff` no parâmetro `minimum_quality` que indica o va-

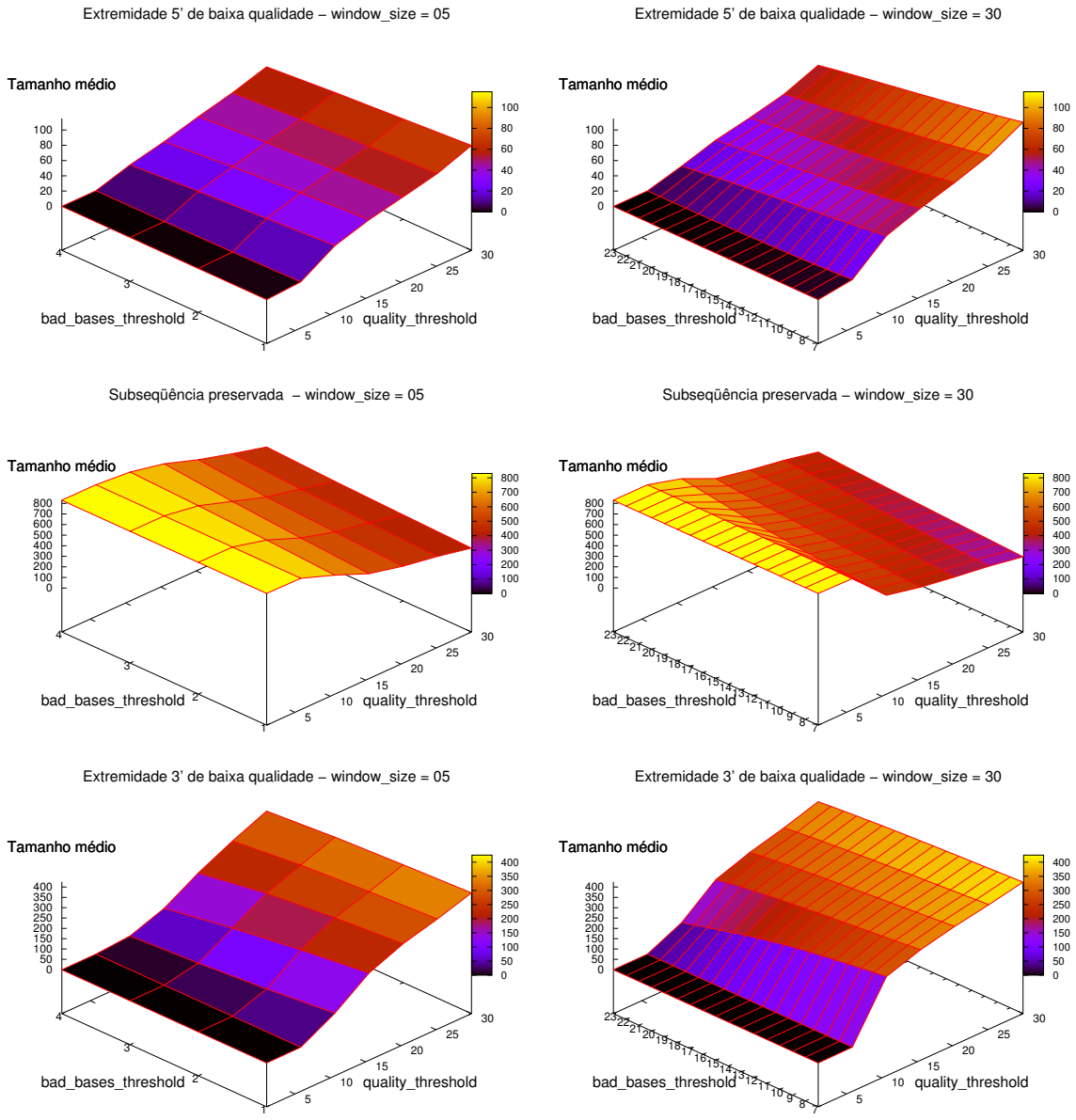

Figura 5.3: Gráficos de superfícies que representam os tamanhos médios dos artefatos de baixa qualidade das extremidades 5′ e 3′ e da subseqüência de boa qualidade em cada uma das execuções do algoritmo de janela deslizante utilizando os valores 5 e 30 no parâmetro `window_size` (colunas esquerda e direita, respectivamente).

lor mínimo de qualidade que uma base deve ter para ser considerada boa. Para obter o valor `trimming_cutoff` utilizado no algoritmo, basta aplicar a seguinte fórmula $\texttt{trimming\_cutoff} = 10^{(\texttt{minimum\_quality}/-10)}$.

O algoritmo foi testado através do processamento de todas as seqüências do conjunto de testes com variação do parâmetro `minimum_quality` dentro do intervalo $[1, 30]$.

Avaliando os resultados de cada execução, produzimos uma série de gráficos que ilustram o funcionamento do algoritmo conforme se altera o parâmetro `minimum_quality`.

A Figura 5.4 exibe gráficos que mostram a distribuição do número de ESTs conforme o tamanho das subseqüências identificadas pelo algoritmo como sendo de boa qualidade. Os dois gráficos são idênticos com exceção da direção do eixo do tamanho da subseqüência. Podemos ver que quando o valor de `minimum_quality` é muito baixo, as subseqüências de boa qualidade são praticamente dos mesmos tamanhos das seqüências originais. À medida que o valor do parâmetro aumenta, a remoção de baixa qualidade é mais efetiva até que, em valores mais altos, descarta completamente a maioria das seqüências.

Na Figura 5.5 observamos a comparação entre os tamanhos médios dos artefatos de baixa qualidade $5'$ e $3'$ e da subseqüência de boa qualidade. Neste gráfico podemos ver, de forma clara, a faixa de valores do parâmetro `minimum_quality` em que a remoção de baixa qualidade realmente começa a ser realizada. Podemos ver também que o maior salto entre as diferenças de tamanhos dos artefatos de execuções consecutivas ($3'$ principalmente) ocorre no intervalo $7 \leq \texttt{minimum\_quality} \leq 9$. Após este intervalo, o algoritmo reduz o tamanho da seqüência original numa taxa bem menor.

O gráfico da Figura 5.6 nos mostra as curvas das médias e das medianas dos tamanhos das subseqüências de boa qualidade, das diversas execuções. Ele reforça a observação feita sobre o gráfico anterior, pois é dentro do intervalo $7 \leq \textit{minimum\_quality} \leq 9$ que as curvas se cruzam, indicando a faixa em que a remoção de baixa qualidade se torna mais agressiva.

5.4.4 LUCY

O LUCY foi utilizado com os seus parâmetros padrões para processar o conjunto de seqüências do SUCEST.

A partir dos resultados que ele produziu, verificamos que as médias dos tamanhos dos artefatos de baixa qualidade $5'$ e $3'$ são de $83,92 \pm 142,11$ e $297,56 \pm 195,87$ bases respectivamente. A média dos tamanhos das subseqüências de boa qualidade é de $447,97 \pm 191,08$ bases. A mediana do tamanho das subseqüências de boa qualidade é de 509 bases.

Considerando que o tamanho médio das seqüências originais é de $829,44 \pm 182,60$, podemos ver que o software é bem agressivo, descartando um número considerável de bases.

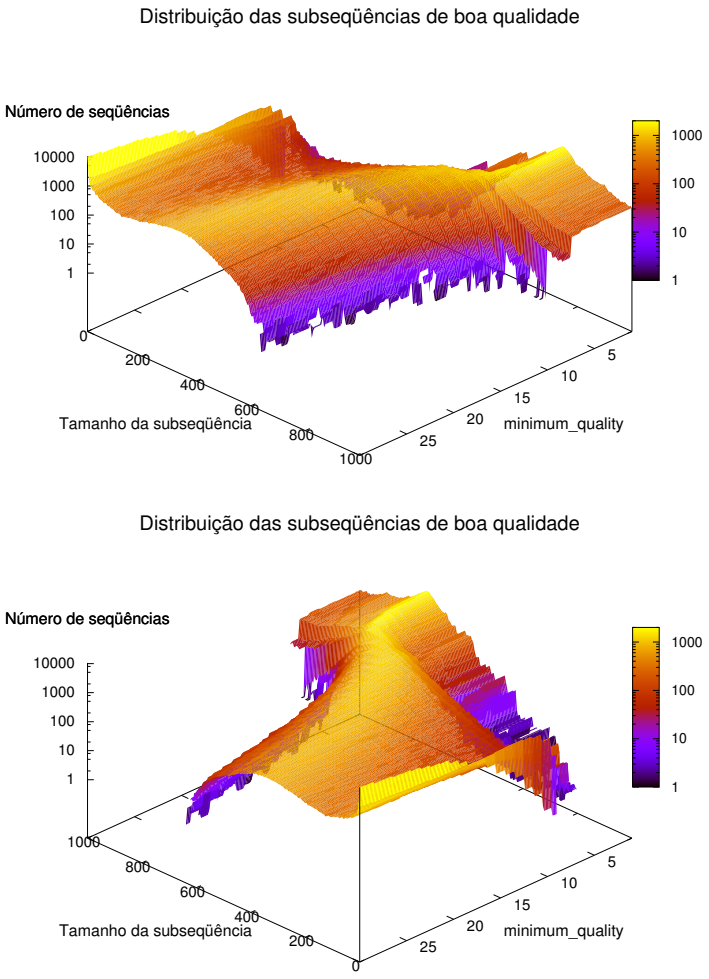

Figura 5.4: Distribuição dos tamanhos das subseqüências de boa qualidade das seqüências processadas pelo algoritmo de subseqüência máxima com o parâmetro `minimum_quality` variando no intervalo $[1, 30]$.

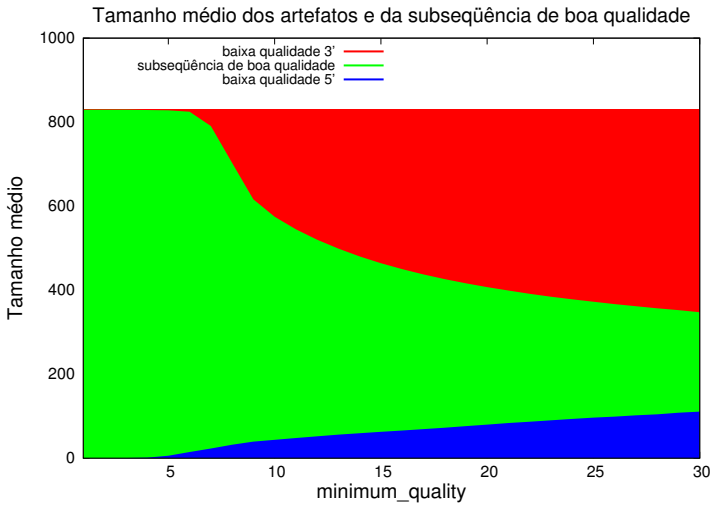

Figura 5.5: Tamanho médio dos artefatos de baixa qualidade (extremidades 5′ e 3′) e da seqüência de boa qualidade das seqüências processadas pelo algoritmo de subseqüência máxima com o parâmetro `minimum_quality` variando no intervalo $[1, 30]$.

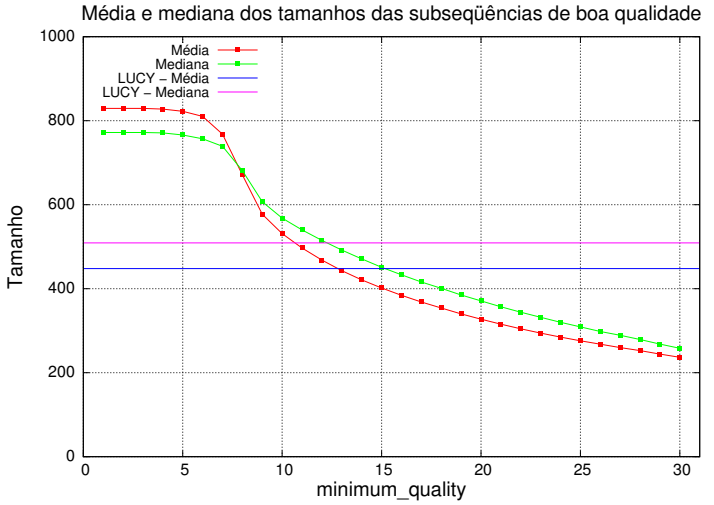

Figura 5.6: Comparação entre média e mediana dos tamanhos das subseqüências de boa qualidade conforme o valor utilizado para o parâmetro `minimum_quality`.

5.4.5 Médias das probabilidades de erro nas extremidades

Um fato que nos chamou atenção em relação ao programa LUCY é a sua preocupação em verificar a existência de bases de baixa qualidade nas extremidades das seqüências.

No algoritmo de janela deslizante, a quantidade de bases de baixa qualidade nas extremidades está diretamente relacionada ao parâmetro *bad_bases_threshold*. Se o valor deste parâmetro é muito alto, a janela aceitará muitas bases de baixa qualidade.

No caso do algoritmo de subseqüência máxima, a qualidade da primeira e da última base da seqüência de boa qualidade são diretamente relacionadas ao valor do parâmetro *minimum_quality*. Sabemos que na extremidade $5'$, o valor que inicia a seqüência é maior que o mínimo, pois caso contrário, a seqüência de boa qualidade não seria iniciada. Na extremidade $3'$ o mesmo ocorre, pois se uma base abaixo do valor mínimo fosse adicionada à seqüência, a soma seria menor do que a soma sem ela e, portanto, a seqüência não poderia ser a subseqüência máxima.

Para avaliar a qualidade das extremidades das seqüências decidimos colher dados sobre as médias das probabilidades de erros das primeiras 25 bases e das últimas 25 bases da seqüência de boa qualidade definida por cada algoritmo. Nós realizamos esta operação sobre 9.600 seqüências selecionadas aleatoriamente dentro do conjunto de 291.689 seqüências do projeto SUCEST. Nós separamos o equivalente a 100 placas compostas por 96 poços. Isso foi feito tendo em vista a complexidade dos testes feitos nesta seção e na próxima, onde utilizaremos o programa BLAST.

No caso do LUCY, verificamos que a média da probabilidade de erro das primeiras 25 bases é de $0,94\%$, enquanto a média das últimas 25 bases é de $3,86\%$.

Os gráficos da Figura 5.7 mostram o comportamento das médias das probabilidades de erros das primeiras 25 bases da seqüência de boa qualidade, determinadas pelo algoritmo de janela deslizante em cada um dos seis tamanhos de janelas testados. Cada uma das curvas representa um valor para o parâmetro *quality_threshold* que, nos gráficos, varia dentro do intervalo $[10, 30]$ (Os valores dentro do intervalo $[1, 9]$ não foram colocados no gráfico para facilitar a visualização e por apresentarem médias de erro muito altas). A linha horizontal vermelha representa o valor de probabilidade de erro média encontrada nas primeiras 25 bases das seqüências trimadas pelo LUCY. Os gráficos da Figura 5.8 representam os mesmos gráficos para a extremidade oposta da seqüência.

A Figura 5.9 mostra as médias de probabilidade de erro das primeiras/últimas 25 bases das seqüências de boa qualidade identificadas nas execuções do algoritmo de subseqüência máxima. As linhas horizontais azul e magenta representam os valores do LUCY.

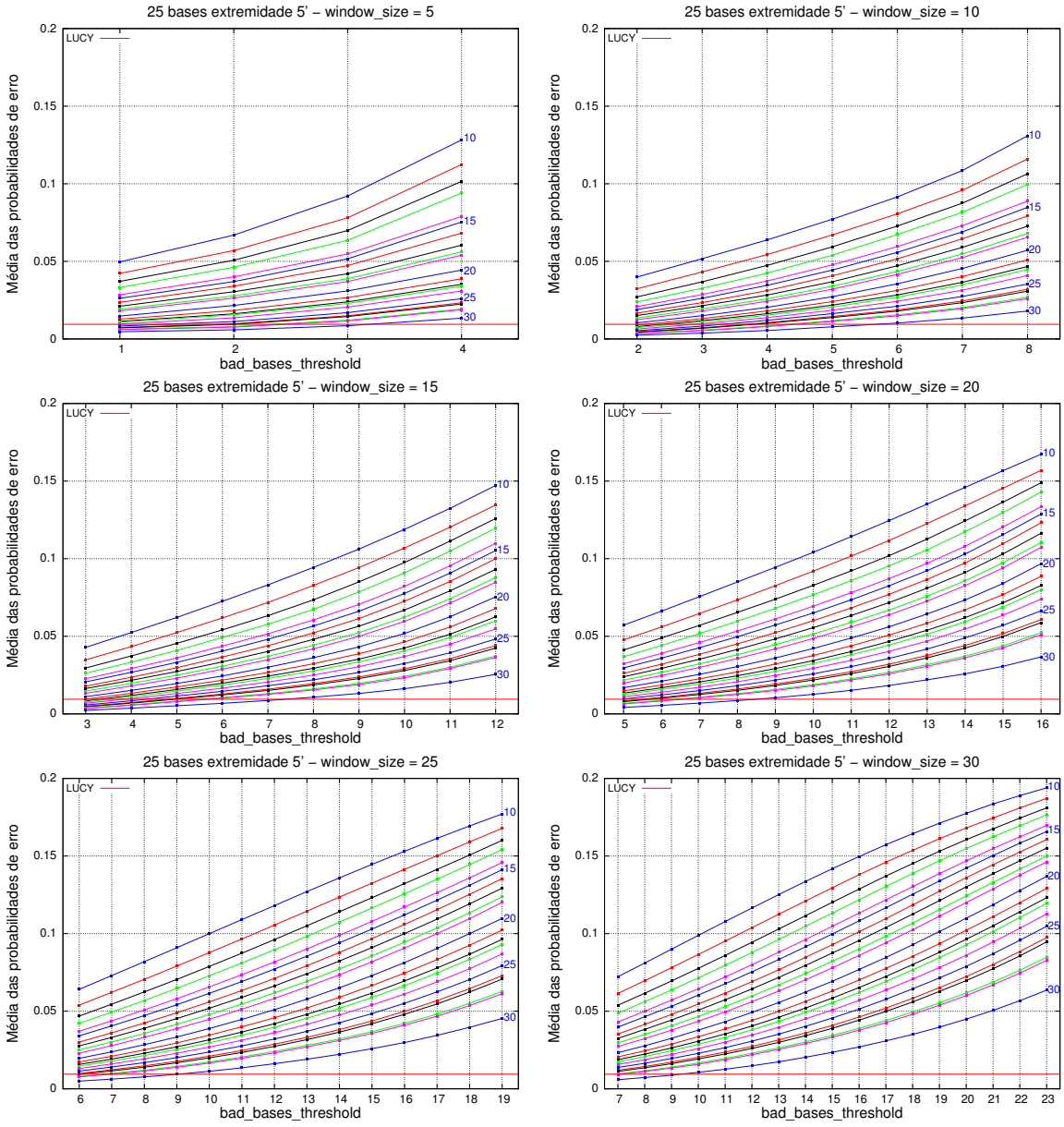

Figura 5.7: Gráficos de curvas que representam as médias das probabilidades de erro das primeiras 25 bases das seqüências de boa qualidade determinadas por cada uma das execuções do algoritmo de janela deslizante com seis tamanhos de janela ($\texttt{window\_size} = \{5, 10, 15, 20, 25, 30\}$) e variação dos parâmetros `bad_bases_threshold` e `quality_threshold` dentro dos intervalos $[10, 30]$ e $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$, respectivamente. Estes dados foram produzidos sobre 9.600 ESTs selecionados aleatoriamente dentro do conjunto de todos os ESTs da cana-de-açúcar. A linha horizontal vermelha indica o valor obtido pelo LUCY.

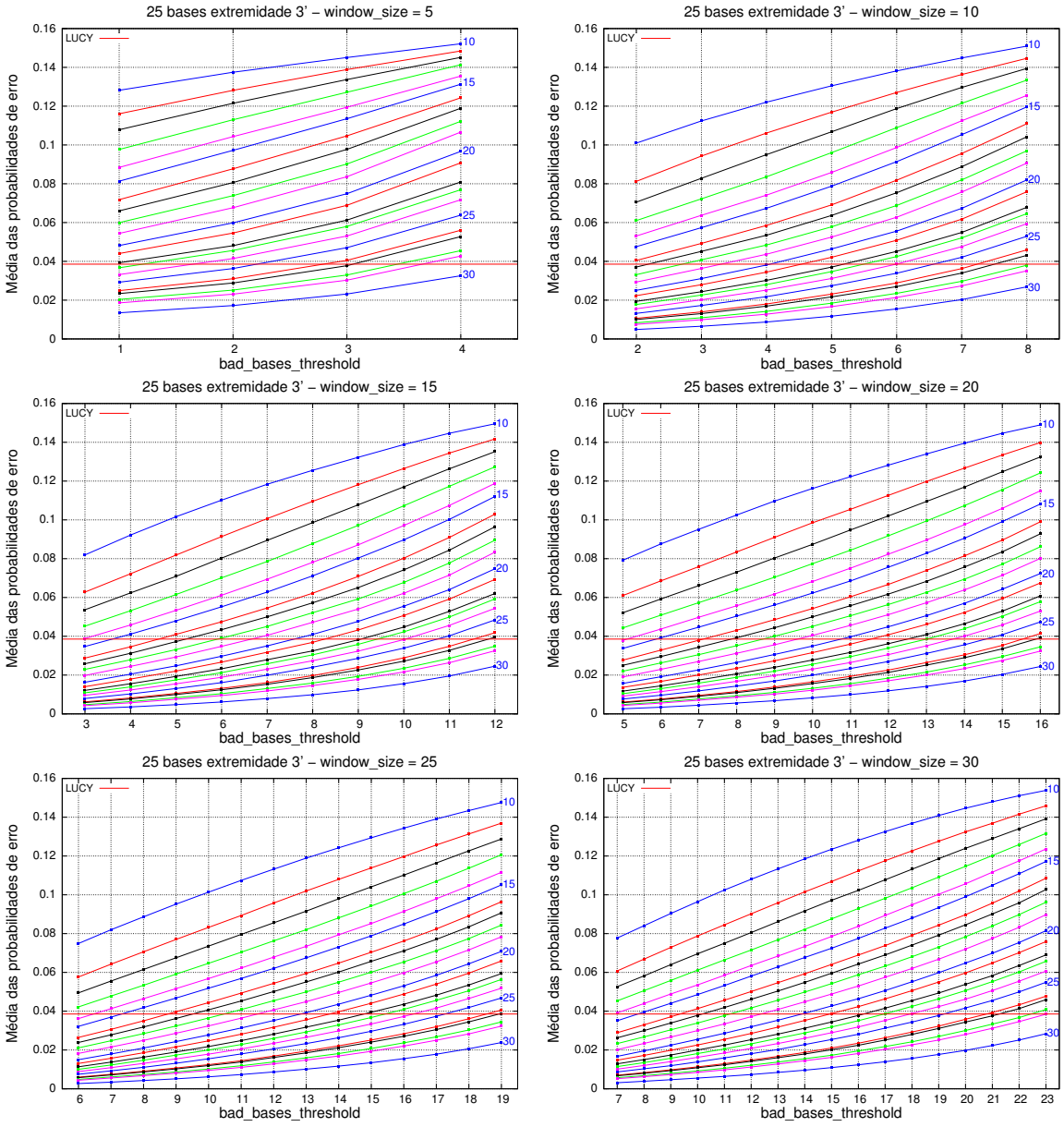

Figura 5.8: Gráficos de curvas que representam as médias das probabilidades de erro das últimas 25 bases das seqüências de boa qualidade determinadas por cada uma das execuções do algoritmo de janela deslizante com seis tamanhos de janela (`window_size` $= \{5, 10, 15, 20, 25, 30\}$) e variação dos parâmetros `bad_bases_threshold` e `quality_threshold` dentro dos intervalos $[10, 30]$ e $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$, respectivamente. Estes dados foram produzidos sobre 9.600 ESTs selecionados aleatoriamente dentro do conjunto de todos os ESTs da cana-de-açúcar. A linha horizontal vermelha indica o valor obtido pelo LUCY.

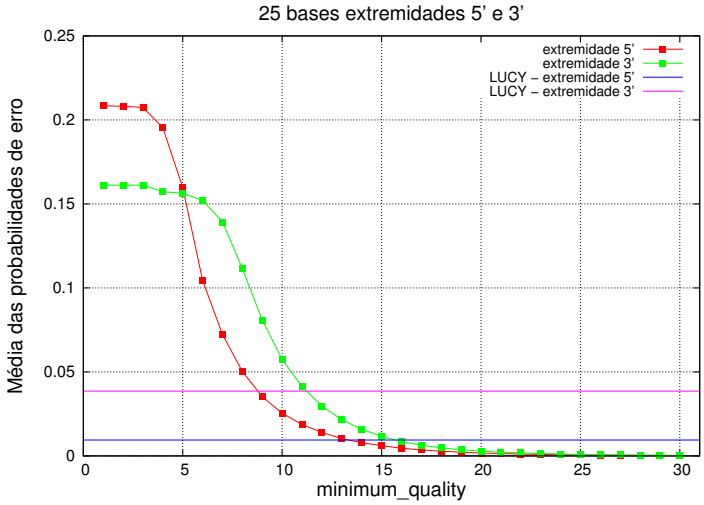

Figura 5.9: Média das probabilidades de erro das primeiras/últimas 25 bases das seqüências de boa qualidade determinadas pelas execuções do algoritmo de subseqüência máxima com o parâmetro `minimum_quality` variando dentro do intervalo $[1, 30]$. Estes dados foram produzidos sobre 9.600 ESTs selecionados aleatoriamente dentro do conjunto de todos os ESTs da cana-de-açúcar. As linhas horizontais azul e magenta indicam os valores obtidos pelo LUCY.

5.4.6 BLAST

A análise da probabilidade de erro é importante, contundo, não podemos nos esquecer de um dos principais objetivos dos projetos ESTs, que é a descoberta de genes.

Poderíamos, por exemplo, escolher o valor 25 para o parâmetro `minimum_quality` no algoritmo de subseqüência máxima, já que os gráficos mostram que nesta região a média da probabilidade de erro das extremidades é próxima de zero. Contundo, se observarmos os gráficos dos tamanhos dos artefatos de baixa qualidade para esta faixa, veremos que eles são muito grandes e que as seqüências de boa qualidade são reduzidas a menos da metade de seus tamanhos originais em média. Esse efeito seria terrível para a clusterização das seqüências, pois a possibilidade de sobreposições entre as seqüências seria reduzida, principalmente entre ESTs de extremidades opostas de um mesmo gene.

Para tentar avaliar o efeito da remoção de baixa qualidade sobre a identificação de genes, decidimos observar a distância média entre as extremidades das seqüências de boa qualidade e seu BLAST hit.

Para isso utilizamos o mesmo conjunto de 9.600 seqüências selecionado anteriormente e utilizamos o software cross_match com os parâmetros `-minmatch` 12 `-minscore` 20 `-penalty` -2 `-screen` para realizar o mascaramento do vetor, utilizado na clonagem,

para que ele não influenciasse o resultado do BLAST.

Após o mascaramento dos trechos de vetor, nós executamos o BLAST destas seqüências contra o banco swissprot. Como o banco swissprot é muito bem curado, selecionamos os melhores hits de cada seqüência que tivessem e-value menor ou igual a 10^{-5}. Neste processo, foram identificadas 4.119 seqüências com hits que atenderam a este critério. Para cada uma delas, anotamos os pontos de início e fim do hit na seqüência.

Os pontos de início e fim foram utilizados para cálculo das distâncias entre o início do hit até a extremidade final do artefato de baixa qualidade $5'$ e entre a extremidade inicial do artefato de baixa qualidade $3'$ e o final do hit. Esta operação foi realizada para todas as execuções dos algortimos de janela deslizante e de subseqüência máxima. Esta análise também foi feita para o LUCY.

A média de distância da entre o início do hit e a extremidade final do artefato de baixa qualidade $5'$, apresentada pelo LUCY, foi de $139,25$ bp. Já a média de distância entre o início do artefato de baixa qualidade $3'$ e o final do hit foi de $-59,46$ bp. A média das porcentagens preservadas dos hits (trecho do hit que não foi cortado pela remoção de baixa qualidade) foi de $74,92\%$.

Os gráficos da Figura 5.10 mostram os resultados obtidos para as médias de distâncias entre o início do hit e a extremidade final do arterfato de baixa qualidade $5'$. A linha horizontal vermelha indica o valor encontrado para o LUCY. Os gráficos da Figura 5.11 representam os resultados encontrados para os artefatos de baixa qualidade $3'$.

Os gráficos da Figura 5.12 exibem as curvas das médias das porcentagens preservadas dos tamanhos dos hits em cada uma das execuções do algoritmo de janela deslizante.

A Figura 5.13 exibe o gráfico contendo os resultados da análise de ambos os artefatos $5'$ e $3'$ para as execuções do algoritmo de subseqüência máxima. As linhas horizontais azul e magenta indicam o valor obtido pelo LUCY.

Na Figura 5.14 podemos observar a curva das médias das porcentagens dos tamanhos dos hits que foram preservadas pela remoção de baixa qualidade em cada uma das execuções do algoritmo de subseqüência máxima. A linha horizontal vermelha indica o valor obtido pelo LUCY.

5.4.7 Escolha do melhor método

A análise da distância média dos artefatos de baixa qualidade ao hit da seqüência apontado pelo BLAST deve ser feita com bastante cuidado. Devemos observar que distâncias negativas não são desejadas pois isso significa que o hit está sendo cortado e, portanto, correrá o risco de não ser identificado pelo BLAST se ele receber como entrada a seqüência processada e não a original.

Contudo, devemos nos lembrar dos resultados apresentados na Seção 5.4.5 que mos-

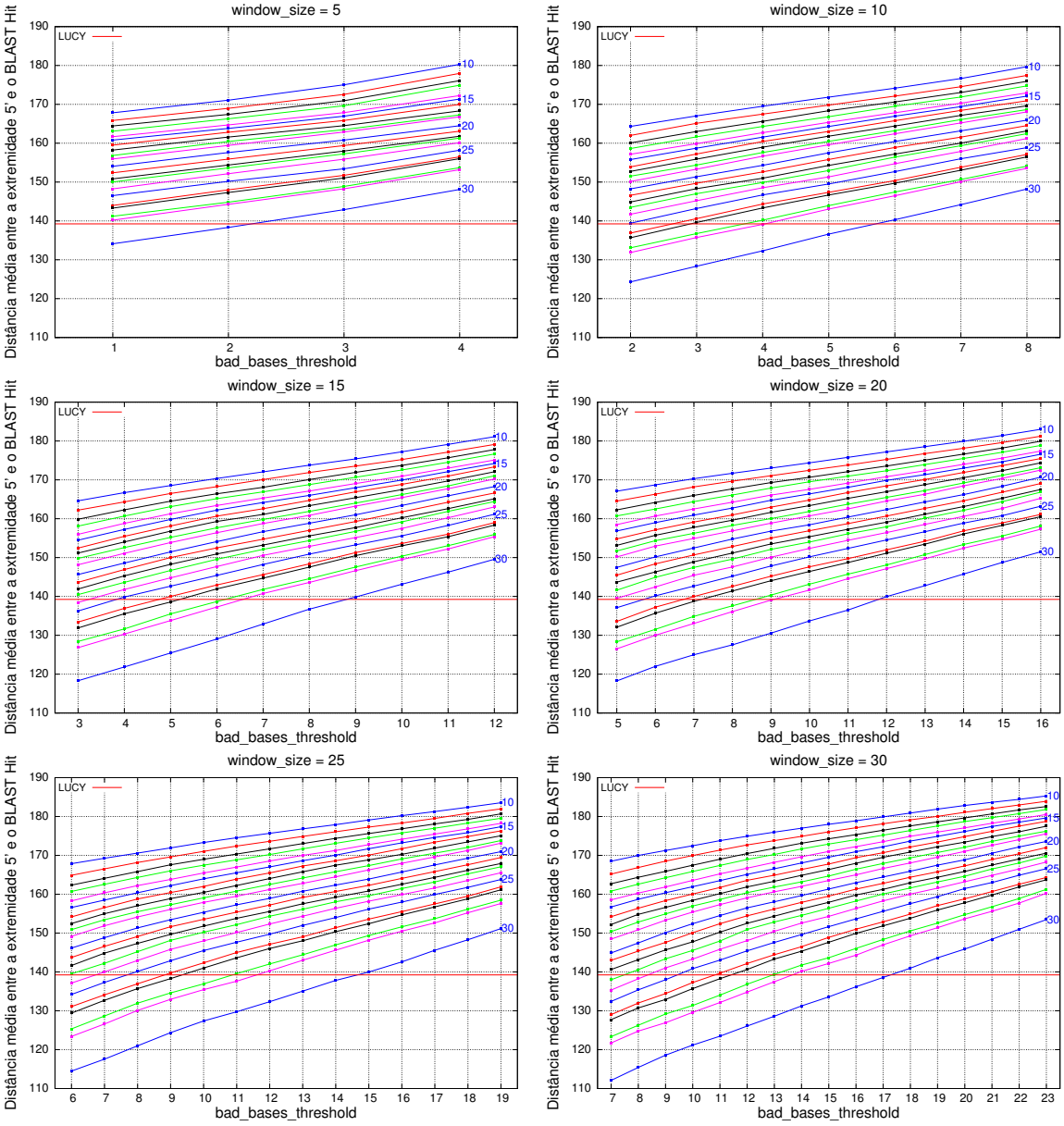

Figura 5.10: Gráficos de curvas que representam as médias das distâncias entre a extremidade final do artefato de baixa qualidade $5'$ e a extremidade inicial do BLAST hit nas seqüências processadas pelo algoritmo de janela deslizante com seis tamanhos de janela (`window_size` $= \{5, 10, 15, 20, 25, 30\}$) e variação dos parâmetros `bad_bases_threshold` e `quality_threshold` dentro dos intervalos $[10, 30]$ e $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$, respectivamente. As curvas foram produzidas a partir do processamento de 4.119 seqüências que apresentaram hit contra o banco swissprot com e-value $\leq 10^{-5}$. A linha horizontal vermelha indica os valores obtidos pelo LUCY.

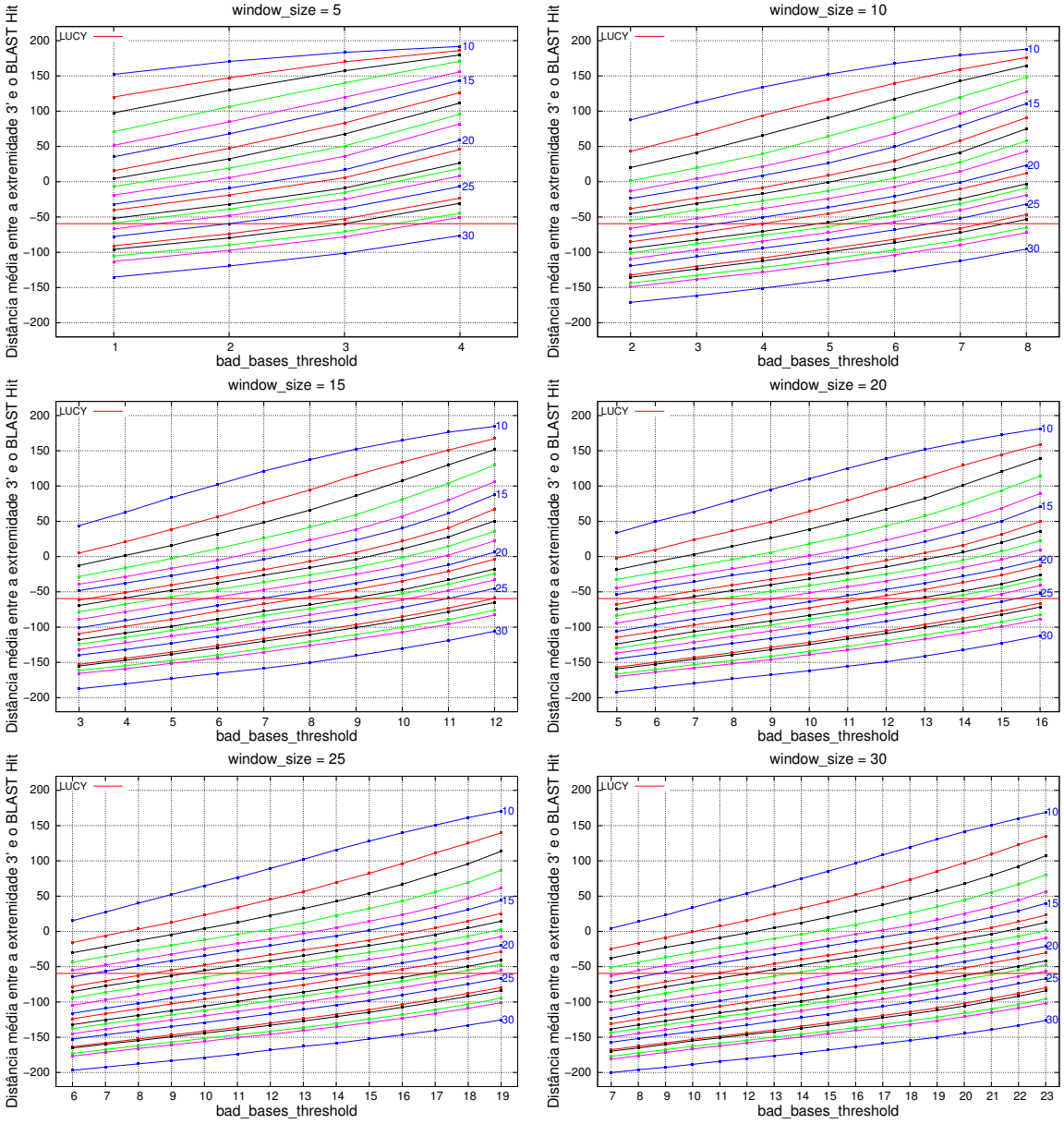

Figura 5.11: Gráficos de curvas que representam as médias das distâncias entre a extremidade inicial do artefato de baixa qualidade $3'$ e a extremidade final do BLAST hit nas seqüências processadas pelo algoritmo de janela deslizante com seis tamanhos de janela (`window_size` $= \{5, 10, 15, 20, 25, 30\}$) e variação dos parâmetros `bad_bases_threshold` e `quality_threshold` dentro dos intervalos $[10, 30]$ e $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$, respectivamente. As curvas foram produzidas a partir do processamento de 4.119 seqüências que apresentaram hit contra o banco swissprot com e-value $\leq 10^{-5}$. A linha horizontal vermelha indica o valor obtido pelo LUCY.

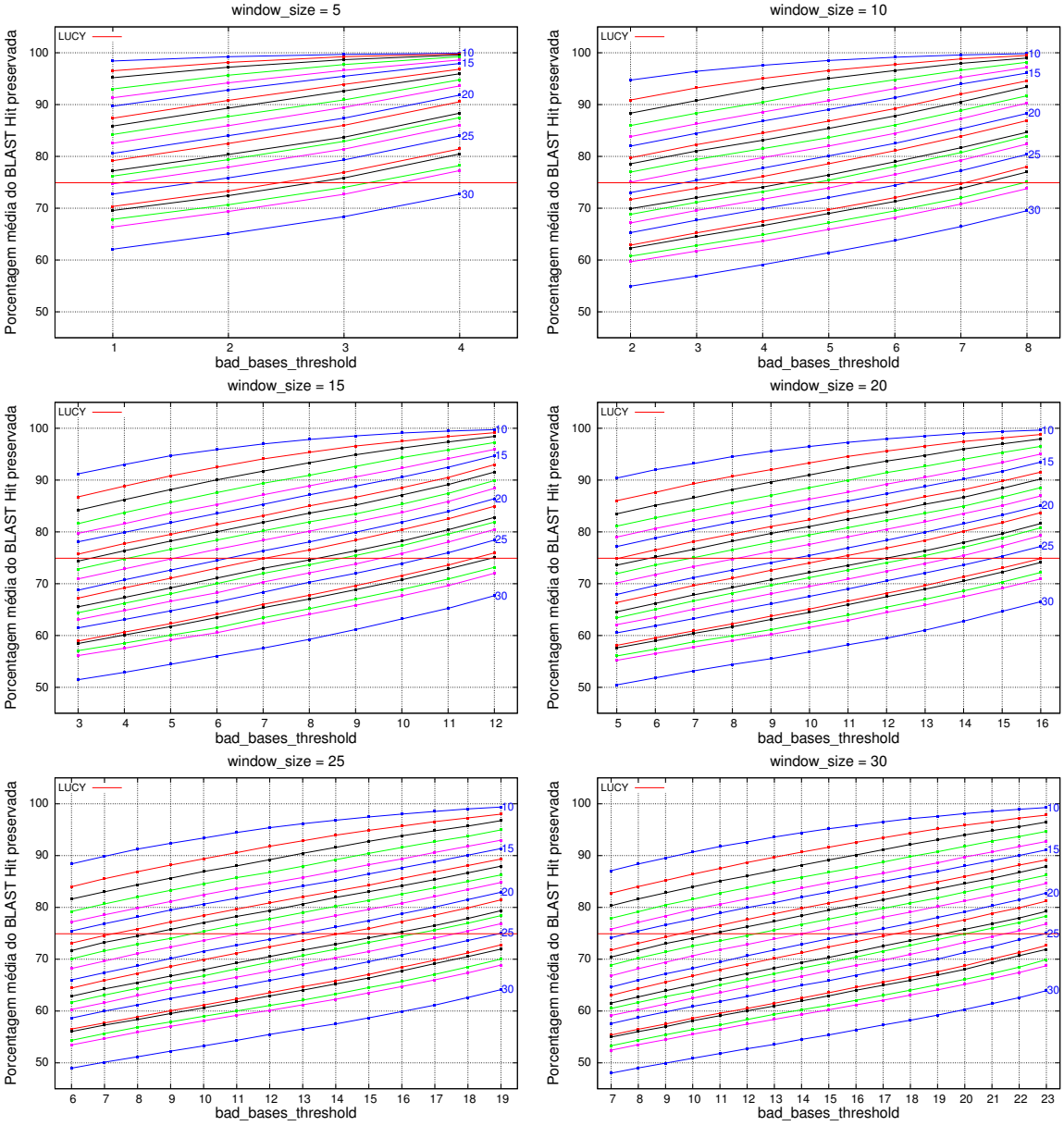

Figura 5.12: Gráficos de curvas que representam a média da porcentagem preservada dos BLAST hits nas seqüências processadas pelo algoritmo de janela deslizante com seis tamanhos de janela (`window_size` $= \{5, 10, 15, 20, 25, 30\}$) e variação dos parâmetros `bad_bases_threshold` e `quality_threshold` dentro dos intervalos $[10, 30]$ e $[\lfloor window\_size/4 \rfloor, \lceil (3 \times window\_size)/4 \rceil]$, respectivamente. As curvas foram produzidas a partir do processamento de 4.119 seqüências que apresentaram hit contra o banco swissprot com e-value $\leq 10^{-5}$. A linha horizontal vermelha indica o valor obtido pelo LUCY.

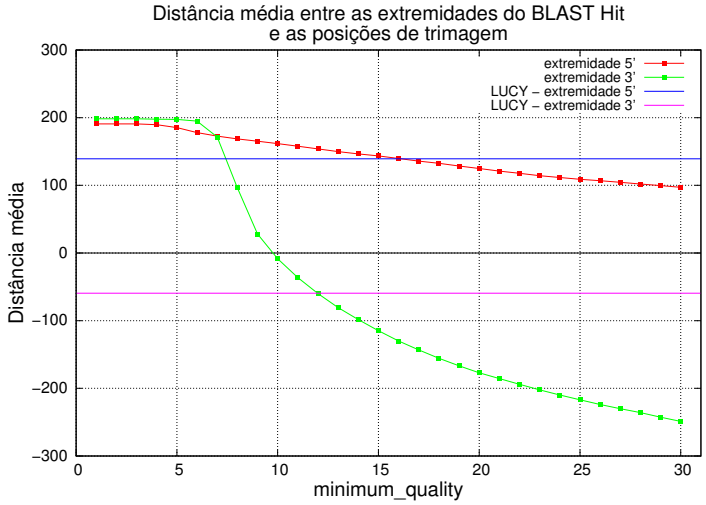

Figura 5.13: Distância média entre a extremidade do BLAST hit e a extremidade do artefato de baixa qualidade nas seqüências processadas pelo algoritmo de subseqüência máxima com variações do parâmetro `minimum_quality` dentro do intervalo $[1, 30]$. As curvas foram produzidas a partir do processamento de 4.119 seqüências que apresentaram hit contra o banco swissprot com e-value $\leq 10^{-5}$. As linhas horizontais azul e magenta indicam o valor obtido pelo LUCY.

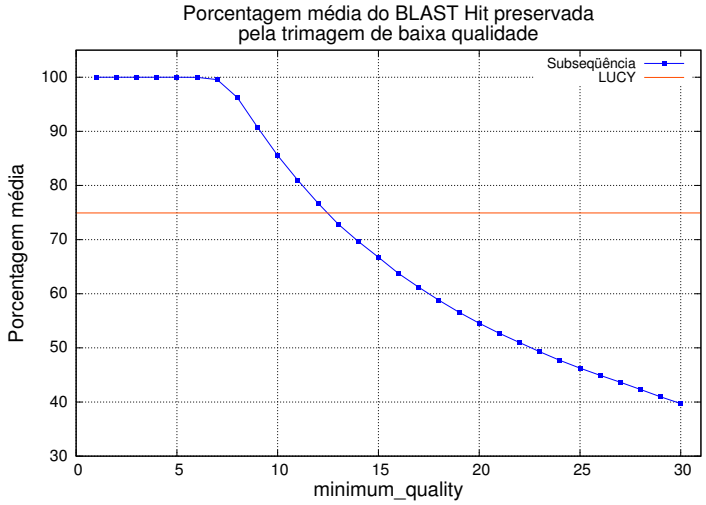

Figura 5.14: Média da porcentagem preservada dos BLAST hits nas seqüências processadas pelo algoritmo de subseqüência máxima com variações do parâmetro `minimum_quality` dentro do intervalo $[1, 30]$. As curvas foram produzidas a partir do processamento de 4.119 seqüências que apresentaram hit contra o banco swissprot com e-value $\leq 10^{-5}$. A linha horizontal vermelha indica o valor obtido pelo LUCY.

tram que se os parâmetros são afrouxados demais, as seqüências processadas passam a apresentar probabilidades de erro médias muito elevadas.

A escolha pelo melhor método baseada nos dados e gráficos produzidos é uma tarefa difícil pois, como foi possível observar, o comportamento dos algoritmos (principalmente o de janela deslizante) costumam ser bem semelhante entre execuções que possuem poucas diferenças entre os valores dos parâmetros utilizados. Além disso, somando todas as execuções diferentes dos dois algoritmos, temos um conjunto de 1.950 resultados.

Assim, decidimos definir uma série de requisitos mínimos para reduzir a gama de opções e nos concentrarmos nas execuções que apresentaram melhores resultados. Para isso, criamos um banco de dados com todas as informações e realizamos buscas de modo a filtrar opções que não eram promissoras.

Os dois primeiros requisitos que definimos estão relacionados às médias das probabilidades de erros das 25 bases de cada uma das extremidades das seqüências de boa qualidade definidas pelos algoritmos. Definimos que a média das probabilidades de erro das primeiras 25 bases (extremidade $5'$) deveria se de no máximo $2,5\%$. Para as últimas 25 bases (extremidade $3'$) definimos que a média deveria ser de no máximo $5,0\%$.

Esta diferença de valores deve-se às características da distribuição da qualidade ao longo da seqüência que foi possível observar no gráfico da Figura 5.1.

Como a qualidade média das bases cresce rapidamente na extremidade $5'$, a definição de um valor mais restritivo tenta evitar que bases de boa qualidade, inseridas num grupo em que a maioria é de baixa qualidade, bloqueiem a remoção deste grupo.

Por outro lado, esta abordagem não se aplica na extremidade oposta. A qualidade das bases da extremidade $3'$ decrescem de maneria mais lenta e a definição de um valor muito restritivo pode acarretar em algoritmos que eliminam uma quantidade de bases maior do que a necessária.

A definição destes dois requisitos permitiu que o número de elementos do conjunto de possibilidades caísse de 1.950 para 359, um número grande ainda.

Tendo em vista um dos principais objetivos de um projeto EST, que é a identificação de genes, decidimos definir um valor mínimo para a média das porcentagens dos tamanhos preservados dos hits. Definimos que em média, 80% do hit deveria ser preservado pelo processo de remoção de baixa qualidade.

Esta exigência reduziu o grupo de 359 possibilidades para apenas seis possibilidades. Destas, cinco eram execuções do algoritmo de janela deslizante e uma era do algoritmo de subseqüência máxima.

A Tabela 5.1 exibe a média das probabilidades de erros das primeiras/últimas 25 bases das seqüência de boa qualidade definidas pelos algoritmos selecionados no conjunto de 9.600 seqüências de cana-de-açúcar escolhidas aleatoriamente. Ela também lista a porcentagem média preservada dos BLAST hits identificados em 4.119 das 9.600 seqüências

Algoritmo	Média de probabilidade de erro nas 25 bases da		Porcentagem média preservada BLAST hit
	extremidade $5'$	extremidade $3'$	
SM [11]	$1,87\%$	$4,12\%$	$80,97\%$
JD $[05, 20, 01]$	$1,48\%$	$4,82\%$	$80,58\%$
JD $[05, 22, 02]$	$1,62\%$	$4,81\%$	$80,37\%$
JD $[10, 15, 02]$	$1,89\%$	$4,76\%$	$81,98\%$
JD $[10, 16, 03]$	$2,34\%$	$4,93\%$	$82,28\%$
JD $[10, 17, 03]$	$2,14\%$	$4,52\%$	$81,04\%$
SUCEST 1 $[20, 15, 08]$	$4,97\%$	$5,04\%$	$81,86\%$
SUCEST 2 $[20, 10, 12]$	$12,46\%$	$12,82\%$	$97,95\%$
LUCY [padrões]	$0,94\%$	$3,86\%$	$74,92\%$

Tabela 5.1: Valores de médias de probabilidade de erro nas primeiras/últimas 25 bases da seqüência de boa qualidade e de porcentagem média preservada do BLAST hit nas seqüências processadas pelas seis execuções selecionadas. A Tabela reúne também os valores obtidas pelas duas configurações utilizadas no projeto SUCEST e pelo LUCY. Entre colchetes são listados os valores utilizados para os parâmetros: `minimum_quality` para o método de subseqüência máxima (SM) e `window_size`, `quality_threshold` e `bad_bases_threshold` para os métodos que utilizam janelas deslizantes (JD, SUCEST1 e SUCEST2).

deste conjunto. Além dos resultados das seis execuções selecionadas, a tabela mostra os valores obtidos pelo LUCY e pelas duas configurações do algoritmo de janela deslizante utilizadas no projeto SUCEST (SUCEST 1 - configuração utilizada antes do trabalho de Telles e da Silva e SUCEST 2 - configuração realmente utilizada no projeto, definida por Telles e da Silva).

A Tabela 5.2 exibe o número de seqüências descartadas (tamanho menor do que 100) e o tamanho médio das seqüências de boa qualidade obtidas por cada um das configurações listadas na Tabela 5.1 sobre o conjunto de todas as seqüências do projeto SUCEST.

Observando as duas tabelas podemos comprovar que o programa LUCY é o mais exigente quanto a remoção de baixa qualidade. Ele possui as menores probabilidades médias de erro, mas também é o que mais sacrifica os hits de BLAST. Em média ele preserva menos de 75% dos hits.

A versão antiga do procedimento de remoção de baixa de qualidade utilizada pelo projeto SUCEST (SUCEST 1) se aproxima bastante dos resultados obtidos pelas configurações que selecionamos. Apesar de ela preservar em média mais de 80% do BLAST hit, as probabilidades médias de erro são maiores que a que nós definimos como limite,

Algoritmo	Número de seqüências descartadas	Tamanho médio das seqüências de boa qualidade
SM [11]	16.272 (5, 58%)	524, 50
JD [05, 20, 01]	14.288 (4, 90%)	525, 39
JD [05, 22, 02]	14.607 (5, 01%)	526, 47
JD [10, 15, 02]	14.379 (4, 93%)	534, 82
JD [10, 16, 03]	13.615 (4, 67%)	536, 19
JD [10, 17, 03]	15.215 (5, 22%)	527, 52
SUCEST 1 [20, 15, 08]	15.470 (5, 30%)	537, 85
SUCEST 2 [20, 10, 12]	437 (0, 15%)	739, 99
LUCY [padrões]	30.018 (10, 29%)	499, 35

Tabela 5.2: Número de seqüências descartadas e tamanho médio das seqüências processadas pelas seis execuções selecionadas. A Tabela reúne também os valores obtidas pelas duas configurações utilizadas no projeto SUCEST e pelo LUCY. Entre colchetes são listados os valores utilizados para os parâmetros: `minimum_quality` para o método de subseqüência máxima (SM) e `window_size`, `quality_threshold` e `bad_bases_threshold` para os métodos que utilizam janelas deslizantes (JD, SUCEST1 e SUCEST2).

especialmente na extremidade $5'$.

A versão mais nova (SUCEST 2), por outro lado, se distancia bastante dos resultados que acreditamos serem ideais. Apesar de ela ser capaz de preservar em média a maior parte do BLAST hit, ela aceita médias de probabilidades de erros nas extremidades muito altas.

Esta característica apresentada pelo algoritmo utilizado no projeto SUCEST se deve a abordagem utilizada em sua definição. Telles e da Silva, em seu estudo, realizaram a análise sobre seqüências com hits "full-length". Nestas seqüências os hits de BLAST possuíam e-value menor ou igual a 10^{-20}, cobriam praticamente toda a extensão da seqüência, mas não alcançavam a sua extremidade final.

Telles e da Silva avaliaram a distância entre a extremidade final do hit e o ponto de corte da baixa qualidade na extremidade $3'$ da seqüência para várias execuções utilizando janela de tamanho 20. Após os testes eles escolheram os valores parâmetros 20, 10 e 12 para os parâmetros `window_size`, `quality_threshold` e `bad_bases_threshold`, respectivamente.

Esta abordagem tenta priorizar a identificação de genes completos. Contudo, ela pode prejudicar o processo de clusterização devido à média mais alta de probabilidade de erro. Por exemplo, duas seqüências, que juntas formariam um cluster que tornasse possível a

identificação de um gene completo, poderiam ser separadas devido aos erros existentes nas extremidades.

Olhando mais atentamente para os dados da Tabela 5.2 vemos que o LUCY descarta praticamente o dobro do número de seqüências que são descartadas pelas nossas opções de algoritmo. Além disso, os tamanhos médios das seqüências são, aproximadamente, 5% menores.

A configuração SUCEST 2, por outro lado, descarta um número muito pequeno de seqüências e os tamanhos médios das seqüências são aproximadamente 41% maiores. Porém, devemos nos lembrar que no esquema de detecção e remoção de artefatos proposto por Telles e da Silva, a detecção de artefatos era feita passo a passo. A remoção de baixa qualidade era efetuada após a remoção de vetores e assim, as seqüências já possuíam um tamanho reduzido em relação a seqüência original. Isso significa que no esquema original, o número de seqüências descartadas seria maior e os tamanho médios seriam menores.

Com base nos dados exibidos pelas duas tabelas decidimos selecionar duas opções de algorimo para continuar o nosso estudo.

A primeira opção foi o subseqüência máxima utilizando o valor 11 para o parâmetro `minimum_quality`. Esta escolha foi feita baseada no menor valor de probabilidade de erro apresentado na extremidade $3'$.

A segunda opção foi pela configuração $[10, 16, 3]$ do algoritmo de janela deslizante. Nesta opção decidimos optar pelo número menor de seqüências descartadas e, principalmente pela maior preservação dos hits de BLAST, reflexo da maior média de tamanho das seqüências de boa qualidade entre as seis execuções selecionadas.

5.4.8 Ilhas de baixa qualidade

As duas escolhas feitas na seção anterior foram utilizadas no estudo de detecção de ilhas de baixa qualidade.

O segundo passo no algoritmo do LUCY é a identificação de pequenas regiões com qualidade muito baixa que permaneceram após o processamento realizado no primeiro passo.

Esta solução é interessante porque detecta regiões que podem prejudicar a montagem de clusters. Se a região de baixa qualidade é muito pequena (algumas bases), o software de clusterização consegue compensar a existência delas e unir as seqüências. No entanto, a medida que esta "ilha de baixa qualidade" cresce, o programa de clusterização começa a separá-las.

A nossa intenção aqui é desenvolver um método capaz de achar regiões de baixíssima qualidade existentes no meio das seqüências.

O procedimento de detecção de ilhas de baixa qualidade utiliza uma janela deslizante que percorre a seqüência base a base até encontrar uma região que possua probabilidade de erro média acima de um valor de corte. Neste momento, ela inicia a construção da ilha de baixa qualidade, que cresce até o momento em que a janela encontra uma região que volte a ter uma probabilidade de erro média abaixo do valor de corte.

Para uma avaliação inicial, decidimos utilizar as seqüências processadas pelo método de subseqüência máxima utilizando o valor 11 para o parâmetro `minimum_quality`. Nós utilizamos janelas de tamanhos 10, 15 e 20 e valores de corte para probabilidade de erro médio de 10%, 15% e 20%.

Executamos as análises, sob o conjunto de todas as seqüências do projeto SUCEST, com cada uma das três janelas combinadas com cada uma das três probabilidades mínimas. Para cada análise, contamos o número de seqüências que apresentavam ilhas de baixa qualidade. Estes números pode ser vistos na Tabela 5.3.

Tamanho da janela	Média de probabilidade de erro		
	10%	15%	20%
10	183.872 (63, 04%)	53.290 (18, 27%)	9.922 (3, 40%)
15	123.774 (42, 43%)	16.674 (5, 72%)	2.006 (0, 69%)
20	80.954 (27, 75%)	5.941 (2, 04%)	559 (0, 19%)

Tabela 5.3: Número de seqüências que apresentaram ilhas de baixa qualidade conforme a combinação Tamanho da janela x Média de probabilidade de erro. Esta avaliação foi feita sobre as todas as seqüências do SUCEST após remoção de baixa qualidade realizada com o algoritmo de subseqüência máxima utilizando o valor 11 para o parâmetro `minimum_quality`.

A Tabela 5.3 nos mostra que quanto menor o tamanho da janela utilizada, maior o número de seqüências que apresentam ilhas de baixa qualidade. Isto é natural, pois quanto maior for o tamanho da janela, maior a chance de existirem bases de boa qualidade que diluem o efeito das bases de baixa qualidade.

Em relação à média de probabilidade de erro, podemos ver que a medida que diminuímos a média de probabilidade de erro mínima, exigida na janela para considerar a região como sendo de baixa qualidade, maior é o número de seqüências com ilhas de baixa qualidade.

O nosso objetivo é encontrar ilhas com qualidades muito baixas. Se considerarmos que as seqüências já foram processadas em um primeiro passo, estas ilhas não deveriam ser tão comuns. Isso indica que utilizar valores muito baixos para a probabilidade de erro mínima poderá gerar um número muito grande de ilhas de baixa qualidade.

Com base nos dados, definimos que a melhor combinação a ser utilizada seria a janela de tamanho 10 com uma média mínima de probabilidade de erro de 20% para considerar a janela como sendo de baixíssima qualidade.

Após a definição das ilhas de baixa qualidade, nosso algoritmo realiza a identificação do ponto de menor qualidade dentro da ilha e o utiliza como ponto de corte na seqüência (em caso de empate, o primeiro é escolhido). Esta operação irá dividir a seqüência em $n + 1$ fragmentos, sendo n o número de ilhas encontrado.

Cada um desses fragmentos é processado pelo algoritmo de detecção e remoção de baixa qualidade utilizado no primeiro passo. Por exemplo, se a seqüência original foi processada com o algoritmo de subseqüência máxima antes da detecção de ilhas, os fragmentos serão processados com este mesmo algoritmo configurado com os mesmos parâmetros.

Dentre todos os fragmentos processados, escolhe-se aquele que tiver a maior soma de qualidade. Caso ocorra empate, seleciona-se o de maior tamanho. No caso de novo empate, seleciona-se a região mais próxima da extremidade $5'$.

Adaptamos as implementações dos nossos dois algoritmos de detecção e remoção de baixa qualidade para incorporar o procedimento descrito acima.

Nós executamos os dois algoritmos modificados utilizando os valores selecionados na seção anterior para processar todos os ESTs da cana-de-açúcar.

A Tabela 5.4 exibe a média das probabilidades de erro ao longo das primeiras/últimas 25 bases das seqüências de boa qualidade definidas por cada execução dos algoritmos originais e usando detecção de ilhas. Ela também lista a média das probabilidade de erro de todas as bases das seqüências de boa qualidade. A Tabela 5.5 mostra o número de seqüências descartadas (tamanho menor do que 100) e o tamanho médio das seqüências de boa qualidade para este mesmo conjunto de execuções.

Com o objetivo de analisar o efeito da introdução desta modificação nos algoritmos sob o aspecto da detecção de genes, nós realizamos o BLAST das seqüências processadas por eles nas diferentes configurações analisadas nesta seção.

Nós executamos os algoritmos modificados sobre as 4.119 seqüências (dentre as 9.600 selecionadas aleatoriamente), que possuíam BLAST hit, para obter as seqüências de boa qualidade definidas por cada um deles. Feito isso, nós realizamos o mascaramento das regiões de vetores e executamos o BLAST contra o banco swissprot. Novamente, utilizamos o valor 10^{-5} como o máximo para seleção dos melhores hits de cada seqüência. A Tabela 5.6 lista o número de seqüências deste conjunto que foram descartadas pelos algoritmos e o número de seqüências que apresentaram hits com e-value dentro do critério.

Analisando as Tabelas 5.4, 5.5 e 5.6 é possível observar que as versões orginais e modificadas dos dois algoritmos possuem resultados muito próximos entre si.

No caso do algoritmo de subseqüência máxima, a proximidade dos resultados é maior do que a apresentada entre as versões original e modificada do algoritmo de janela des-

Algoritmo	Média da probabilidade de erro		
	nas 25 bases da		na seqüência
	extremidade 5′	extremidade 3′	completa
SM v1 [11]	1,88%	4,12%	1,55%
SM v2 [11]	1,87%	4,08%	1,51%
JD v1 [10, 16, 03]	2,34%	4,97%	1,95%
JD v2 [10, 16, 03]	2,56%	4,51%	1,71%
SUCEST 1 [20, 15, 08]	4,95%	5,09%	1,94%
SUCEST 2 [20, 10, 12]	12,47%	12,96%	5,34%
LUCY [padrões]	0,94%	3,84%	1,05%

Tabela 5.4: Médias de probabilidade de erros observadas nas primeiras/últimas 25 bases e na seqüência de boa qualidade completa nas execuções dos algoritmos originais (v1) e usando detecção de ilhas de baixa qualidade (v2) com os 2 conjutos de parâmetros selecionados. A Tabela exibe também os valores obtidos pelas duas configurações utilizadas no projeto SUCEST e pelo LUCY. Entre colchetes são listados os valores utilizados para os parâmetros: `minimum_quality` para o método de subseqüência máxima (SM) e `window_size`, `quality_threshold` e `bad_bases_threshold` para os métodos que utilizam janelas deslizantes (JD, SUCEST1 e SUCEST2).

Algoritmo	Número de seqüências descartadas	Tamanho médio das seqüências de boa qualidade
SM v1 [11]	16.272 (5,58%)	524,50
SM v2 [11]	16.403 (5,62%)	520,91
JD v1 [10, 16, 03]	13.615 (4,67%)	536,19
JD v2 [10, 16, 03]	15.401 (5,28%)	521,01
SUCEST 1 [20, 15, 08]	15.470 (5,30%)	537,85
SUCEST 2 [20, 10, 12]	437 (0,15%)	739,99
LUCY [padrões]	30.018 (10,29%)	499,35

Tabela 5.5: Número de seqüências descartadas e tamanho médio das seqüências de boa qualidade observados nas execuções dos algoritmos originais (v1) e usando detecção de ilhas de baixa qualidade (v2) com os 2 conjutos de parâmetros selecionados. A Tabela exibe também os valores obtidos pelas duas configurações utilizadas no projeto SUCEST e pelo LUCY. Entre colchetes são listados os valores utilizados para os parâmetros: `minimum_quality` para o método de subseqüência máxima (SM) e `window_size`, `quality_threshold` e `bad_bases_threshold` para o método de janela deslizante (JD, SUCEST1 e SUCEST2).

Algoritmo	Número de seqüências descartadas	Número de seqüências com BLAST hit com e-value $\leq 10^{-5}$
SM v1 [11]	39 (0,95%)	3.722 (90,36%)
SM v2 [11]	40 (0,97%)	3.711 (90,09%)
JD v1 [10, 16, 03]	20 (0,49%)	3.755 (91,16%)
JD v2 [10, 16, 03]	36 (0,87%)	3.708 (90,02%)
SUCEST 1 [20, 15, 08]	32 (0,78%)	3.734 (90,65%)
SUCEST 2 [20, 10, 12]	0 (0,00%)	4.078 (99,00%)
LUCY [padrões]	146 (3,54%)	3.527 (85,63%)

Tabela 5.6: Número de seqüências descartadas e número de seqüências com BLAST hit e-value $\leq 10^{-5}$ no conjunto de 4.119 seqüências processadas pelos algoritmos originais (v1) e usando detecção de ilhas de baixa qualidade (v2) com os 2 conjutos de parâmetros selecionados. A Tabela exibe também os valores obtidos pelas duas configurações utilizadas no projeto SUCEST e pelo LUCY. Entre colchetes são listados os valores utilizados para os parâmetros: `minimum_quality` para o método de subseqüência máxima (SM) e `window_size`, `quality_threshold` e `bad_bases_threshold` para os métodos que utilizam janelas deslizantes (JD, SUCEST1 e SUCEST2).

lizante. Isso indica que o algoritmo de subseqüência máxima, em sua versão original, já produz uma boa remoção de baixa qualidade. Contudo, acreditamos que a utilização da versão modificada para detecção de ilhas de baixa qualidade é uma boa opção devido ao refinamento que ela produz nos resultados.

As maiores distâncias entre os resultados das versões original e modificada do algoritmo de janela deslizante mostram que o efeito da adoção da modificação é muito mais efetiva neste algoritmo. Isso ocorre devido à própria natureza do algoritmo. Uma janela percorre a seqüência construindo o trecho de baixa qualidade enquanto não for encontrada nenhuma janela que possua os critérios mínimos de boa qualidade. Uma vez que esta janela de boa qualidade é encontrada, a análise é interrompida. Isso significa que o algoritmo não realiza a análise da seqüência toda, como no caso do algoritmo de subseqüência máxima. Como a seqüência não é toda analisada, eventualmente existem regiões internas de baixa qualidade que são descartadas pelo algoritmo de subseqüência máxima, mas não são descartadas pelo algoritmo de janela deslizante.

As Tabelas 5.4, 5.5 e 5.6 mostram, assim como todos os testes anteriores, que o LUCY possui o método mais exigente quanto a remoção de baixa qualidade. Suas seqüências possuem médias de probabilidade de erro menores, assim como a média de seus tamanhos. Além disso, o LUCY descarta mais seqüências e perde mais hits de BLAST.

Por outro lado, o método adotado por Telles e da Silva é o menos exigente. O número de seqüências descartadas é mínimo e a probabilidade média de erro são as mais altas. De fato, o método de Telles e da Silva supõe que as seqüências deveriam ter sido processadas por dois passos anteriores: remoção de seqüências ribossomais e remoção de vetor e de poli-A. Contudo, observando os resultados apresentados no trabalho deles, ainda é possível concluir que o método é bastante tolerante quanto a baixa qualidade. Após a realização das duas etapas iniciais (remoção de seqüências ribossomais e remoção de vetor e de poli-A), existiam 275.436 seqüências. Feita a análise de baixa qualidade, apenas 1.708 seqüências foram descartadas, o equivalente a $0,62\%$ deste conjunto ou $0,59\%$ de todas as seqüências.

No entanto, mesmo tendo as piores médias de probabilidade de erros, o método adotado no projeto SUCEST apresentou o melhor aproveitamento quanto a preservação de BLAST hits.

Baseado nos resultados acima, elegemos o algoritmo de subseqüência máxima, modificado para detecção de ilhas de baixa qualidade, como o ideal para o nosso conjunto de procedimentos de detecção e remoção.

Nosso objetivo principal é a construção de bons clusters e acreditamos que as menores taxas de erro das seqüências produzida por este algoritmo serão positivas, mesmo considerando o menor tamanho das seqüências.

Comparando este método com o LUCY, que é um software utilizado por um grande grupo como o TIGR, podemos observar que nosso método consegue boas médias de probabilidade de erros sem descartar um número grande de seqüências.

Se, por outro lado, o objetivo do projeto de seqüenciamento for a identificação do maior número de genes com o menor número de seqüências possível, o ideal seria adotar uma configuração do algoritmo de subseqüência máxima, combinado com a detecção de ilhas de baixa qualidade, com parâmetros que produzissem seqüências com tamanho médios maiores, mesmo que a probabilidade de erro média fosse sacrificada.

Capítulo 6

Procedimento completo de detecção e remoção de artefatos

Durante o estudo realizado, nós procuramos criar um conjunto de procedimentos que fosse capaz de realizar a detecção e a remoção de artefatos de forma eficiente.

Utilizando como base o trabalho de Telles e da Silva, desenvolvemos uma estratégia de detecção e remoção de artefatos (Capítulo 3). Este conjunto utilizou um nova estratégia de detecção de artefatos. Ao invés de realizar a remoção de artefatos através de etapas dependentes umas das outras, a nossa estratégia visa a detecção de diferentes tipos de artefatos sem que um influencie na detecção do outro.

Além do desenvolvimento deste conjunto básico, realizamos estudos específicos nas etapas de remoção de artefatos de baixa qualidade (Capítulo 5) e de derrapagem (Capítulo 4).

Este capítulo apresentará o resultado final de nossa pesquisa. A Seção 6.1 apresentará detalhadamente cada uma das etapas do procedimento. A Seção 6.2 mostrará os resultados da aplicação do procedimento sobre as seqüências do projeto SUCEST.

Os resultados produzidos neste capítulo foram apresentados como pôster no congresso "14^{th} Annual International Conference On Intelligent Systems For Molecular Biology (ISMB2006)", realizado em Agosto de 2006, em Fortaleza – CE, sob o título "New EST trimming procedure applied to SUCEST sequences".

6.1 Etapas do procedimento de detecção e remoção de artefatos

Durante o estudo procuramos produzir um conjunto de procedimentos bem simples formado por diversas etapas que analisam a seqüência para remoção de diferentes tipos de artefatos. Estas etapas serão descritas nas seguintes subseções:

6.1.1 Descarte de seqüências com conteúdo ribossomal

6.1.2 Detecção de artefatos de baixa qualidade

6.1.3 Detecção de artefatos de vetor

6.1.4 Detecção de artefatos de adaptadores

6.1.5 Detecção de caudas poli-A e poli-T

6.1.6 Detecção de trechos de derrapagem

6.1.7 Identificação do inserto e remoção de seqüências curtas

A última etapa (6.1.7) não é exatamente um procedimento de detecção e remoção de artefatos. Ela é na verdade a etapa de finalização de todo o processo, quando todos os trechos identificados pelas demais etapas são combinados para que seja possível separar apenas a região que será preservada para o processo de análise das seqüências.

Todas as demais etapas são totalmente independentes e podem ser executadas em qualquer ordem. Contudo, recomendamos a execução da etapa de descarte de seqüências com conteúdo ribossomal antes das outras etapas. A recomendação se deve ao fato de que esta etapa, quando encontra regiões com DNA ribossomal, marca toda a seqüência como sendo um artefato. Como a seqüência inteira é marcada para ser descartada, não é necessária a realização das outras análises, permitindo, assim, uma economia de tempo.

6.1.1 Descarte de seqüências com conteúdo ribossomal

A detecção de seqüências ribossomais é realizada rigorosamente da mesma maneira que a proposta por Telles e da Silva. Ela consiste na execução do BLAST da seqüência a ser analisada contra um banco formado por seqüências ribossomais.

A seqüência será descartada como um artefato ribossomal, se ela apresentar pelo menos um hit com e-value máximo de 10^{-10}.

A construção do banco de seqüências ribossomais é o item mais importante nesta etapa. Ele deverá ser formado por seqüências ribossomais do mesmo organismo que está sendo seqüenciado ou por seqüências ribossomais de qualquer organismo filogeneticamente próximo.

A utilização de seqüências de organismos filogeneticamente próximos é possível devido à grande conservação apresentada pelo DNA ribossomal. Isso foi demonstrado em nossos testes quando realizamos o processamento dos ESTs do boi (*Bos taurus*) contra um banco de seqüências ribossomais do porco (*Sus sucrofa*)(Capítulo 3).

6.1.2 Detecção de artefatos de baixa qualidade

Nós realizamos um amplo estudo sobre a detecção e remoção de baixa qualidade no Capítulo 5.

Após analisar uma série de possibilidades, concluímos que a melhor opção é a utilização do algoritmo de subseqüência máxima com `minimum_quality` = 11, combinado com o procedimento de detecção de ilhas de baixa qualidade usando janela de tamanho 10 e probabilidade média de erro de 20%.

6.1.3 Detecção de artefatos de vetor

A remoção de vetores é feita com a ajuda do software cross_match.

O software é configurado com os parâmetros recomendados na própria documentação do programa: `-minmatch` 12 e `-minscore` 20.

Toda região da seqüência que apresentar alinhamento com a seqüência do vetor é marcada como um artefato.

6.1.4 Detecção de artefatos de adaptadores

O programa swat é utilizado na deteção de adaptadores. Ele realiza o alinhamento da seqüência do adaptador com a seqüências que está sendo analisada.

Para gerar o alinhamento, o programa é configurado com uma matriz de pontuação que atribui 1 ponto para cada coincidência e -2 pontos para cada erro. Além disso, são utilizados os seguintes parâmentros: `-gap_init` -5, `-gap_ext` -5, `-ins_gap_ext` -5, `-del_gap_ext` -5 e `-end_gap` -5, com o objetivo de minimizar a ocorrência de buracos no alinhamento.

A região alinhada que tiver tamanho maior ou igual ao tamanho do adaptador menos quatro é marcada como artefato. Caso exista mais de uma região, a que apresentar maior pontuação é escolhida.

Note que o valor utilizado como limite para o tamanho do alinhamento deve ser analisado caso a caso.

No nosso estudo, trabalhamos com adaptadores de tamanho 11 e 16, e a utilização deste valor limite mostrou bons resultados.

Contudo, para adaptadores menores, o valor a ser subtraído de seus tamanhos deve ser menor. Caso contrário, alinhamentos muito pequenos serão marcados como artefatos, gerando falsos positivos.

Por outro lado, valores maiores podem ser utilizadas em casos de adaptadores maiores, com o objetivo de diminuir a ocorrência de falsos negativos.

6.1.5 Detecção de caudas poli-A e poli-T

A remoção de caudas poli-A e poli-T também utiliza o software swat.

A mesma configuração da etapa anterior é utilizada para alinhar cada seqüência com seqüências formadas por 500 As ou 500 Ts.

Todas as regiões que apresentarem alinhamento com pontuação mínima 10 são marcados como artefatos.

6.1.6 Detecção de trechos de derrapagem

O Capítulo 4 descreve o nosso estudo sobre a detecção e remoção de artefatos de derrapagem.

Neste estudo, nós optamos pela utilização do método chamado "Cobertura por ecos" utilizando o valor 5 para o parâmetro `minimum_echo_size`, o valor 8 para o parâmetro `minimum_number_of_echoes` e a estratégia *subseqüência*. Além disso, adotamos o valor de corte $0,25$ como pontuação mínima que a região deve possuir para ser considerada como um artefato de derrapagem.

6.1.7 Identificação do inserto e remoção de seqüências curtas

Após a realização de todos os passos acima, cada seqüência possui um conjunto de artefatos. Os artefatos são então mascarados na seqüência, e qualquer trecho bom (não mascarado) que tiver tamanho menor do que 100 é descartado.

No caso de existir mais de um trecho bom com tamanho maior do que 100, o primeiro critério de seleção é preservar o trecho com maior soma de qualidade. O objetivo é manter uma boa relação entre tamanho e qualidade da seqüência.

Em caso de empate, o segundo critério de seleção é a região de maior tamanho. Isto visa de privilegiar o procedimento de clusterização devido à chance de produzir maiores sobreposições.

Finalmente, em caso de um novo empate, seleciona-se a região que estiver mais próxima da extremidade inicial da seqüência.

6.2 Avaliação do conjunto de procedimentos de detecção e remoção de artefatos

Para avaliar o conjunto de procedimentos descritos acima, utilizamos as seqüências ESTs produzidas pelo projeto SUCEST.

Todas as 291.689 seqüências foram processadas por todas as etapas e os resultados produzidos em cada etapa serão descritos nas subseções a seguir.

6.2.1 Descarte de seqüências com conteúdo ribossomal

Para realizar esta etapa, utilizamos o mesmo banco de seqüências ribossomais que o empregado no trabalho de Telles e da Silva.

O banco é composto pelas seqüências `GenBank AF168884` (rRNA 18S do organismo *Zea mays*), `GenBank AF162215` (rRNA 5,8S do organismo *Platanus occidentalis*) e `GenBank AF162215` (rRNA 26S do organismo *Lambertua inermis*). Todos estes organismos são filogeneticamente próximos da cana-de-açúcar.

Nesta etapa, descartamos 8.843 seqüências. Este número é um pouco maior do que o número de seqüências descartadas no trabalho de Telles e da Silva (8.473). A razão da diferença nos resultado pode ser explicada pela utilização de versões diferentes do programa BLAST. Nós utilizamos a versão 2.2.11 de 05/07/2005 enquanto a versão utilizada por eles foi a de 31/10/2000.

Todas estas seqüências são automaticamente descartadas do processo. Assim, restaram 282.846 seqüências para serem processadas pelas demais etapas.

6.2.2 Detecção de artefatos de baixa qualidade

O conjunto de 282.846 seqüências foi processado pela etapa de remoção de baixa qualidade.

Artefatos de baixa qualidade $5'$ foram detectados em 280.471 seqüências $(99,16\%)$ e apresentaram tamanho médio de $48,71 \pm 109,29$ bp. Os artefatos de baixa qualidade $3'$ foram identificados em 279.508 seqüências $(98,82\%)$ e mostraram tamanho médio de $288,84 \pm 223,25$ bp.

Apenas nove seqüências $(0,003\%)$ foram descartadas completamente (um único artefato que cobre toda a extensão da seqüência). Em 16.134 seqüências $(5,70\%)$ a porção de boa qualidade atingiu tamanho menor que 100 bp, tamanho que já qualifica o descarte da seqüência.

As 266.703 seqüências $(94,29\%)$, cujas porções de boa qualidade atingiram tamanho maior ou igual a 100 bp, apresentaram tamanho médio de $524,23 \pm 119,66$ bp.

6.2.3 Detecção de artefatos de vetor

O programa cross_match foi utilizado para alinhar o conjunto sem seqüências ribossomais com a seqüências do vetor *pSport1*.

Trechos deste vetor foram encontrados em 215.265 seqüências $(76,11\%)$ e totalizaram 250.705 artefatos. Estes artefatos apresentaram uma média de $76,49 \pm 108,15$ bp.

Apenas 17 seqüências $0,006\%$ foram marcadas inteiramente como um único artefato de vetor e em 7.323 seqüências $(2,59\%)$ os trechos não marcados como vetor possuíam tamanhos menores do que 100 bp.

6.2.4 Detecção de artefatos de adaptadores

Dois adaptadores foram utilizados com o vetor *pSport1* no projeto SUCEST: o *pSport1-1* (`ccacgcgtccg`) e o *pSport1-2* (`tcgacccacgcgtccg`).

Utilizando o programa swat, realizamos os alinhamentos e identificamos a ocorrência do adaptador *pSport1-1* em 253.953 seqüências $(89,78\%)$ e do adaptador *pSport1-2* em 224.579 seqüências $(79,40\%)$. A média de tamanho dos artefatos foi de $10,48 \pm 1,27$ bp e $15,79 \pm 0,77$ bp, respectivamente. Para nenhum dos dois adaptadores nós verificamos a ocorrência de mais de uma região alinhada em uma mesma seqüência.

6.2.5 Detecção de caudas poli-A e poli-T

Artefatos poli-A foram encontrados em 52.050 seqüências $(18,40\%)$ e apresentaram tamanho médio de $31,45 \pm 36,71$ bp. Artefatos poli-T se mostraram um pouco menos freqüentes, aparecendo em 49.130 seqüências $(17,37\%)$ e mostrando tamanho médio de $30,61 \pm 32,80$ bp.

Um total de 47 seqüências $(0,02\%)$ apresentaram regiões não marcadas como poli-A com tamanhos menores que 100 bp. Destas, apenas uma foi marcada como sendo totalmente composta por poli-A.

No caso da cauda poli-T, nenhuma seqüência foi marcada como sendo um único artefato, mas quatro seqüências $(0,001\%)$ apresentaram regiões não marcardas como poli-T menores do que o tamanho mínimo aceitável.

6.2.6 Detecção de trechos de derrapagem

Os artefatos de derrapagem ocorreram em 6.045 seqüências $(2,14\%)$. Um total de 6.986 artefatos apresentaram tamanho médio de $196,35 \pm 139,19$ bp.

Se considerássemos apenas a remoção de derrapagem, apenas 293 seqüências $(0,10\%)$ seriam descartadas por possuírem regiões não marcadas como artefatos com tamanhos menores do que 100 bp. Dentre estas, apenas uma foi marcada como sendo inteiramente formada por derrapagem.

6.2.7 Identificação do inserto e remoção de seqüências curtas

Finalmente, a última etapa tratou de agrupar todos os artefatos das etapas anteriores para produzir o conjunto final de seqüências de boa qualidade. A Tabela 6.1 resume o resultado de todas as etapas anteriores ao mostrar a quantidade e o tamanho médio dos artefatos encontrados.

Após a combinação de todos os artefatos e descarte das seqüências com tamanho menor do que 100 bp, restaram 253.848 seqüências ($87,03\%$ do conjunto inicial de 291.689 ESTs) com tamanho médio de $472,05 \pm 121,68$ bp e qualidade média $33,25 \pm 14,78$.

Em seu trabalho, Telles e da Silva produziram, após a execução de seu conjunto de procedimentos, 237.954 seqüências de boa qualidade ($81,56\%$). Estas seqüências apresentaram tamanho médio de $641,57 \pm 139,79$ bp e qualidade média $27,74 \pm 14,30$.

Como podemos ver, o nosso conjunto de procedimentos descarta um número menor de seqüências. Além disso, as seqüências processadas segundo nossos métodos apresentam, em média, tamanhos menores e qualidades maiores que as apresentadas pelas seqüências processadas pelos métodos utilizados no projeto SUCEST.

Artefato	Número de artefatos	Tamanho médio
Ribossomal	8.843	–
Baixa qualidade $5'$	280.471	$48,71 \pm 109,29$
Baixa qualidade $3'$	279.508	$288,84 \pm 223,25$
Vetor pSport1	250.705	$76,49 \pm 108,15$
Adaptador pSport1-1	253.953	$10,48 \pm 1,27$
Adaptador pSport1-2	224.579	$15,79 \pm 0,77$
Poli-A	52.050	$31,45 \pm 36,71$
Poli-T	49.130	$30,61 \pm 32,80$
Derrapagem	6.986	$196,35 \pm 139,19$

Tabela 6.1: Quantidade e tamanho médio dos artefatos conforme o seus tipos. O tamanho médio do artefato ribossomal não foi calculado porque o método não delimita início e fim do artefato, mas apenas marca a seqüência para descarte.

6.3 Clusterização das seqüências processadas

Para avaliar o efeito das diferenças observadas entre os resultados de nossos métodos e os dos métodos utilizados no projeto da cana-de-açúcar, realizamos a clusterização das seqüências processadas pelo nosso conjunto de procedimentos para comparação com o clustering oficial do projeto SUCEST.

A clusterização, assim como no projeto da cana-de-açúcar, foi feita com o programa CAP3 utilizando seu parâmetros padrões.

Para realizar este procedimento nós tínhamos à disposição uma máquina com dois processadores INTEL Xeon $3,2$GHz, 4GB DDR ECC e quatro discos 320 ULTRA SCSI 133GB e sistema operacional Fedora Core 4. No entanto, apesar de possuir 4GB de memória, o sistema operacional Fedora Core 4 é capaz de endereçar no máximo 2 GB por processo e, por isso, não conseguimos executar o CAP3 utilizando o conjunto completo de seqüências.

Assim, foi necessário a realização do processamento de um conjunto menor de dados. Antes de realizar a seleção nós verificamos que as seqüências do projeto SUCEST tiveram origem em 26 bibliotecas diferentes [113]. Estas bibliotecas agrupavam clones produzidos a partir de tecidos e variedades diferentes de cana-de-açúcar.

A seleção de um conjunto menor de seqüências deveria, idealmente, manter proporções de seqüências de cada biblioteca semelhante às observadas no conjunto original.

Para atingir este objetivo, construímos o novo conjunto de dados de acordo com o seguinte procedimento: nós realizamos a ordenação dos nomes de todas as seqüências em ordem alfabética e selecionamos todas as seqüências que estavam em posições ímpares. Dessa maneira, selecionamos, aproximadamente, metade das seqüências de cada biblioteca produzida no projeto.

Este novo conjunto continha 145.845 seqüências com tamanho médio de $834,64 \pm 182,86$ bp e qualidade média $23,07 \pm 14,98$.

Após o processamento destas seqüências pelos nossos métodos, o conjunto foi reduzido a 126.988 seqüências de boa qualidade ($87,07\%$) com tamanho médio de $473,32 \pm 121,66$ bp e qualidade média $33,25 \pm 14,78$. Já o conjunto processado pelos métodos de Telles e da Silva ficou com 118.991 seqüências ($81,59\%$) com tamanho médio de $643,82 \pm 141,32$ e qualidade média $27,69 \pm 14,30$. Estes dados estão todos agrupados na Tabela 6.2.

	Seqüências	Tamanho médio	Qualidade Média
Seqüências selecionadas	145.845	834.64 ± 182.86	$23,07 \pm 14,98$
Processadas pelo método SUCEST	118.991	$643,82 \pm 141,32$	$27,69 \pm 14,30$
Processadas pelo nosso método	126.988	$473,32 \pm 121,66$	$33,25 \pm 14,78$

Tabela 6.2: Número de seqüências, tamanho médio das seqüências e qualidade médias das seqüências encontradas no conjunto de seqüências selecionadas (formado por aproximadamente metade da seqüências do projeto SUCEST) e nos conjuntos produzidos pelo processamento das seqüências selecionadas pelos métodos de detecção e remoção de artefatos empregados no SUCEST e pelos métodos desenvolvidos neste trabalho.

O clustering produzido com as seqüências analisadas pelos métodos do projeto SUCEST, gerou um total de 36.596 clusters divididos em 20.202 singletons (clusters de tamanho 1) e 16.394 contigs (clusters de tamanho maior do que 1). Os tamanhos médios apresentados pelos singletons e pelos contigs foram de $635,19 \pm 157,27$ e $921,52 \pm 426,17$ bp, respectivamente. Para facilitar a análise, chamaremos este clustering de `TS`.

O clustering produzido com o conjunto de seqüências processadas pelos nossos métodos (clustering `BD`) gerou 39.965 clusters divididos em 22.479 singletons e 17.486 contigs. Os tamanhos médios apresentados pelos singletons e pelos contigs foram de $437,48 \pm 143,11$ e $840,30 \pm 398,91$ bp, respectivamente.

A partir da comparação dos conjuntos de clusters dos dois clusterings, observamos que 16.543 singletons e 9.280 contigs apareceram em ambos com a mesma composição de seqüências. No clustering `TS`, estes singletons apresentaram tamanho médio de $632,55 \pm 156,43$ bp, enquanto no clustering `BD`, eles apresentaram tamanho médio de $465,70 \pm 125,05$ bp. Os contigs comuns aos dois clusterings apresentaram tamanho médio de $804,69 \pm 322,75$ e $780,92 \pm 325,71$, respectivamente nos clusterings `TS` e `BD`.

Como podemos observar nos números acima, os tamanhos médios dos singletons e dos contigs do clustering `BD` são menores do que os do clustering `TS`. Este resultado é reflexo dos menores tamanhos médios apresentados pelas seqüências processadas pelos nossos métodos.

6.3.1 Avaliação dos clusterings

Para avaliar os clusterings e, conseqüentemente, avaliar os conjuntos de métodos de detecção e remoção de artefatos, realizamos um processamento para extrair dados de consistência externa, consistência interna, redundância, número de clusters full-length, número de SNPs e número de INDELs.

Consistência externa

Ao avaliar a consistência externa de um clustering, nós procuramos observar a ocorrência de erros de clusterização em que seqüências, provenientes de um mesmo gene, são agrupadas em clusters diferentes.

Para realizar esta avaliação, realizamos o BLAST da seqüência consenso de cada cluster contra todas as seqüências consensos dos outros clusters do mesmo clustering. Feito isso, nós analisamos os hits encontrados e procuramos por aqueles que apresentassem sobreposições de pelo menos 200 bases e que não se situassem a mais de 10 das extremidades das seqüências. Além disso, a sobreposição deveria apresentar identidade mínima de 75%.

O número de sobreposições encontrado foi então dividido por $n(n-1)/2$, que é o

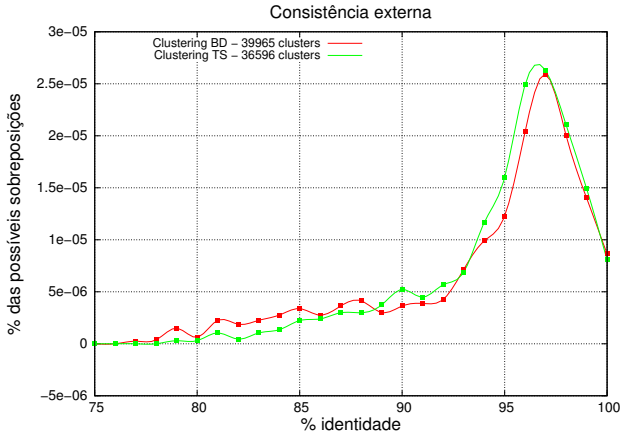

Figura 6.1: Distribuição das sobreposições encontradas no BLAST de "todos contra todos" clusters dos clusterings `TS` e `BD`. As sobreposições deveriam ter tamanho mínimo de 200 bases e estar localizadas no máximo a 10 bases de uma das extremidades das seqüências. O gráfico mostra, para cada um dos clusterings, a distribuição das sobreposições encontradas em função da identidade apresentada na sobreposição. O valor no eixo y indica a porcentagem de sobreposições encontradas dentro do número máximo número de sobreposições possíveis ($n(n-1)/2$, onde n é o número de clusters).

número de máximo de sobreposições para n clusters. Esta divisão é feita para que tenhamos a porcentagem das possíveis sobreposições que realmente ocorreram.

O clustering `TS` apresentou um total de 1.098 sobreposições, ou seja, $1,64 \times 10^{-4}\%$ das possíveis sobreposições. Já o clustering `BD` produziu 1.269 sobreposições, equivalente a $1,59 \times 10^{-4}\%$ das possíveis sobreposições.

A partir das sobreposições encontradas nos construímos o gráfico da Figura 6.1. Cada ponto do gráfico é resultado da função $f(x) = x \times [n(n-1)/2] \times 100$, onde x é o número de sobreposições encontradas com $x\%$ de identidade.

Observando os dados produzidos nesta análise, verifica-se que o clustering `BD` apresenta um número de sobreposições ligeiramente menor do que o apresentado pelo clustering `TS`, considerando o número total de sobreposições possíveis. No gráfico da Figura 6.1 podemos ver que o clustering `TS` apresenta um porcentual ligeiramente maior de sobreposições na faixa que considera as sobreposições com porcentagem de identidade maiores do que 89%. Este resultado pode indicar que os nossos métodos de detecção e remoção de artefatos possuem uma leve tendência de redução das ocorrências de erros de clusterings ocasionados pela quebra de um ou mais clusters em clusters menores.

Consistência interna

A avaliação de consistência interna tem o objetivo de analisar a ocorrência de erros de clusterização que promovem a união, em um mesmo clusters, de seqüências que, na realidade, deveriam estar separadas.

Para realizar esta análise, nós levantamos dados sobre a discrepância existente entre as bases das seqüências que compõem o cluster e as bases da seqüência consenso. Note que apenas clusters com tamanhos maiores ou iguais a 2 são analisados.

Consideramos uma base discrepante quando a sua probabilidade de erro é menor do que $x\%$ e ela difere da base escolhida para o consenso do cluster. A partir desta definição, fizemos duas análises.

A primeira análise, semelhante à realizada por Telles e da Silva, verificou a distribuição das seqüências que compõem os clusters conforme a porcentagem de bases discrepantes existentes dentro delas. A porcentagem foi calculada através da razão da soma do número de bases discrepantes, existentes dentro do cluster, pelo número total de bases (soma dos tamanhos de todas as seqüências que participam do cluster).

A segunda análise verificou a distribuição dos clusters conforme o número de posições do consenso que possuíam pelo menos uma base discrepante no alinhamento. Para cálculo desse número, cada coluna do alinhamento de um cluster era analisada em busca de, pelo menos, uma base discrepante, caso em que o contador de posições discrepantes do cluster era incrementado.

As duas análises foram feitas utilizando os valores de probabilidade de erro menores ou iguais a 2%, 10% e 100% (qualidades maiores ou iguais a 17, 10 e 0).

Os gráficos exibidos na Figura 6.2 foram construídos com os dados da primeira análise. Eles mostram como as seqüências que compõem os clusters se distribuem conforme o valor de porcentagem de posições discrepantes que elas possuem.

Os dados da segunda análise geraram os gráficos exibidos na Figura 6.3. Estes gráficos mostram a distribuição de clusters em função do número de posições discrepantes no alinhamento.

Analisando os gráficos das duas figuras (6.2 e 6.3), podemos observar que quando utilizamos um valor de qualidade maior (17) os dois clusterings possuem comportamento similar, sendo que o clustering `BD` apresenta inconsistência interna levemente maior. No entanto, quando diminuímos o valor da qualidade a ser considerada para a análise de consistência, o quadro se inverte. Essa diferença ocorre devido à menor média de qualidade das seqüências produzidas pelo processo de detecção e remoção de artefatos utilizado no projeto SUCEST.

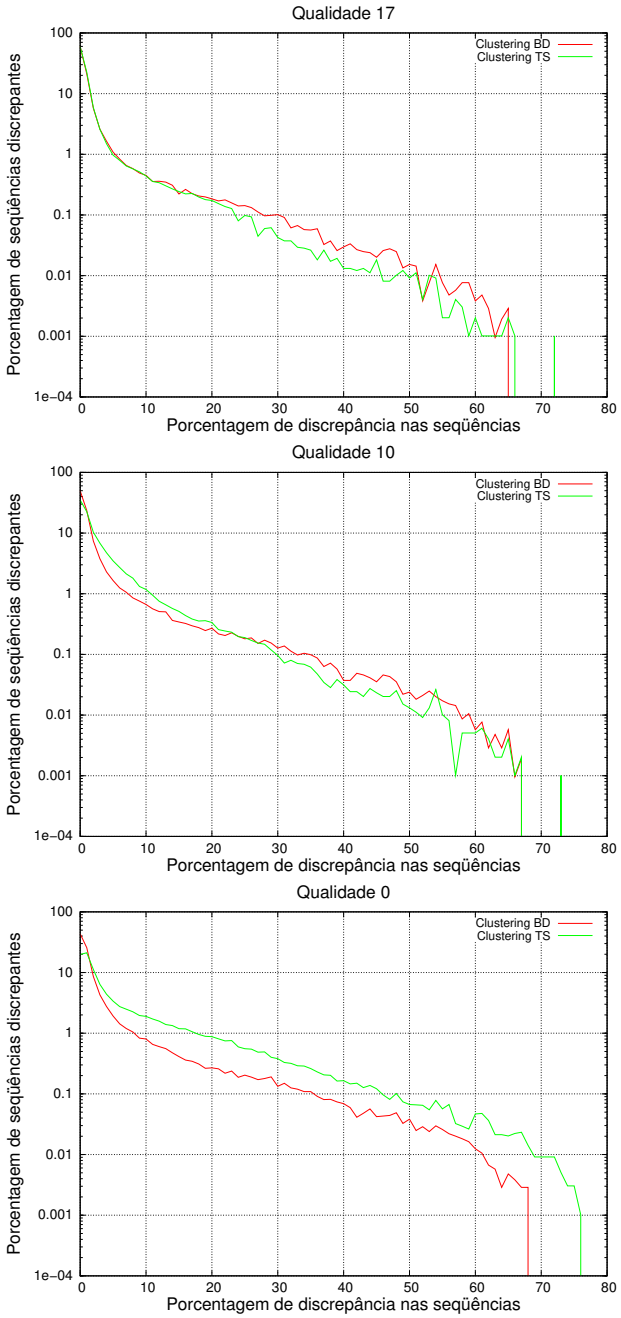

Figura 6.2: Distribuição das seqüências que participaram da montagem de clusters em função da porcentagem de discrepância encontradas nas seqüências. Para cada cluster com tamanho maior ou igual a 2, verificou-se, em cada seqüência, a porcentagem de bases com qualidades maiores ou iguais a 17, 10 e 0 (probabilidades de erros menores ou iguais a 2%, 10% e 100%) que diferiam da base escolhida para o consenso.

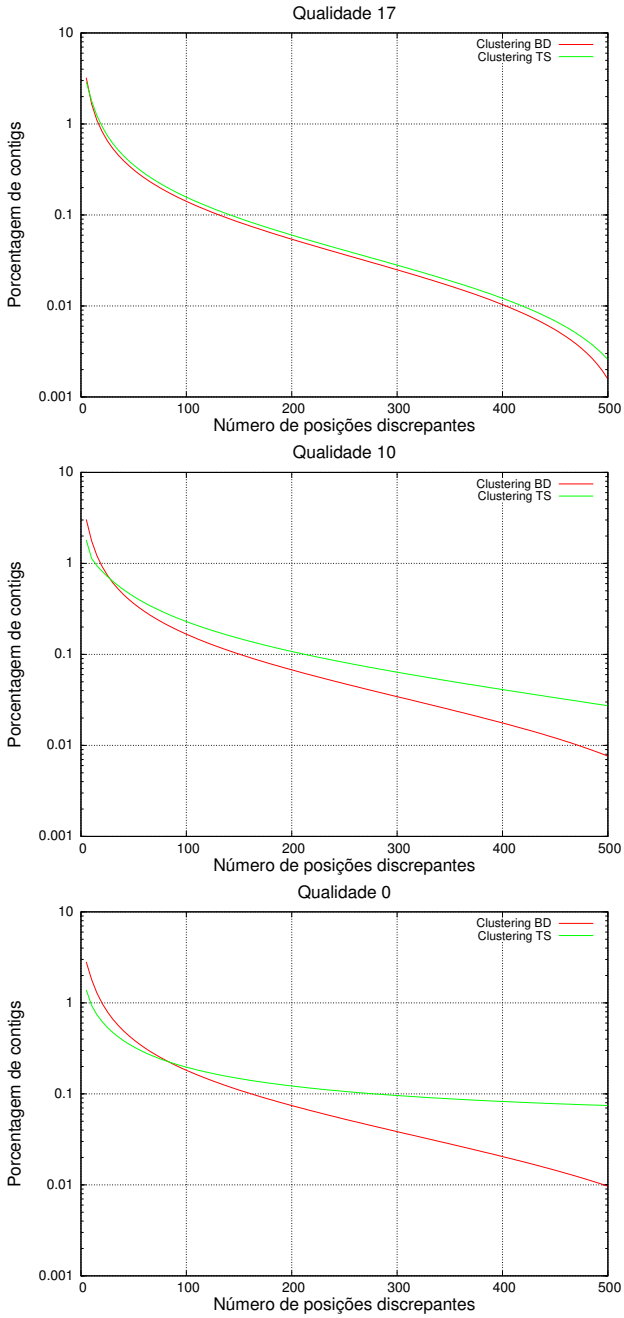

Figura 6.3: Distribuição dos clusters em função do número de posições discrepantes encontradas no alinhamento de suas seqüências. Para cada clusters com tamanho maior ou igual a 2, verificou-se o número de posições no alinhamento que possuíam pelo menos uma base com qualidade maior ou igual a 17, 10 e 0 (probabilidade de erro menor ou igual a 2%, 10% e 100%) que diferiam da base escolhida para o consenso.

Redundância

Outra maneira de avaliar a qualidade da clusterização é verificar a redundância existente entre seus clusters. Um clustering com redundância muito alta pode indicar que o processo de clusterização separou clusters, que na realidade deviam estar juntos, possivelmente, influenciado pelo conjunto de seqüências.

A avaliação da redundância de cada clustering foi feita com um conjunto formado por todas as seqüências dos singletons e dos consensos dos contigs do clustering. Cada seqüência do conjunto foi comparada com todas as outras com utilização do programa cross_match para identificação de quais seqüências seriam agrupadas formando "contigs" e quais permaneceriam sozinhas formando "singletons". Quanto maior o número de "contigs", maior a redundância apresentada pelo clustering.

O cross_match foi executado de duas maneiras diferentes. A primeira é similar à realizada no trabalho de análise dos dados produzidos pelo projeto SUCEST [112] onde o programa é executado com os parâmetros `-penalty` -10, `-minmatch` 32 e `-minscore` 77. Todo par de seqüências que foi apontado pelo cross_match foi considerado na formação dos "contigs", independentemente do tamanho do alinhamento.

A segunda maneira foi a execução do programa cross_match com os parâmetros padrões e realização da filtragem dos resultados produzidos por ele. Os pares de seqüências encontrados só foram considerados para agrupamento em "contigs" quando possuíam alinhamento com pelo menos 98% de identidade ao longo de no mínimo 100 bases.

Segundo a maneira utilizada no trabalho de análise dos dados do SUCEST, o clustering `TS` apresentou uma redundância de $18,56\%$ (25.902 "singletons" e 3.903 "contigs") enquanto o clustering `BD` mostrou uma redundância de $14,51\%$ (30.749 "singletons" e 3.417 "contigs").

De acordo com a segunda maneira, a redundância apresentada foi de $6,10\%$ (32.818 "singletons" e 1.547 "contigs") para o clustering `TS` e $5,97\%$ (35.989 "singletons" e 1.592 "contigs") para o clustering `BD`.

As redundâncias apresentadas por ambos os métodos executados indicam que o clustering `TS` possui redundância maior do que a apresentada pelo clustering `BD`, mesmo possuindo um menor número de seqüências formando os clusters. Isto pode evidenciar que o clustering `TS` pode ter mais erros de clusterização causados pela menor qualidade média das seqüências.

Clusters full-length

Para avaliar o número de clusters full-length apresentado por cada um dos clusterings, nós executamos o BLAST das seqüência dos singletons e dos consensos dos clusters contra o banco `nr`.

Para definir um cluster como sendo full-length, utilizamos o mesmo critério utilizado no trabalho de análise dos dados do SUCEST [112]. Um cluster é considerado full-length quando o seu melhor hit apresenta e-value menor ou igual a 10^{-40} e alinhamento com início nas 15 primeiras posições da seqüência subject.

O BLAST contra o banco `nr` apontou um total de 4.941 ($13,50\%$) clusters full-length para o clustering `TS` e 4.408 ($11,03\%$) para o clustering `BD`.

Decompondo os resultados, temos que dos 4.941 hits full-length obtidos pelo clustering `TS`, 3.702 foram encontrados nos contigs e 1.239 nos singletons. Já no caso do clustering `BD`, foram 3.602 hits encontrados nos contigs e 882 nos singletons.

O menor tamanho das seqüências processadas pelos nossos métodos de detecção e remoção de artefatos resultou na diminuição do tamanho médio dos singletons e das seqüências consensos. Como conseqüência, observou-se uma redução do número de hits full-length encontrados através do BLAST das seqüências dos clusters contra o banco `nr`.

Contagem de SNP e INDEL

O critério para contagem de SNPs e INDELs também foi extraído do trabalho de análise dos dados da cana-de-açúcar.

Um candidato a SNP era identificado nas colunas do alinhamento, do consenso com as seqüências que compõem o contig, que possuíssem pelo menos duas bases idênticas com qualidades maiores ou iguais a 20 e diferentes da base do consenso. Apenas contigs formados por quatro ou mais seqüências foram considerados (7.515 para o clustering `TS` e 7.867 para o clustering `BD`).

Feita a identificação dos candidatos, um processamento foi realizado de modo a impedir a detecção de mais de um SNP dentro de uma janela de 5 bases com alinhamento de qualidade maior ou igual a 20.

A identificação dos eventos de INDELs também foi feita a partir da análise do alinhamento do consenso com as seqüências que compunham o cluster. Os INDELs foram detectados como séries de posições adjacentes de alinhamentos que possuíssem, entre as seqüências participantes, pelo menos duas que estivessem marcadas com gap ao invés de uma base. Além disso, as qualidades das bases das outras seqüências, na mesma posição do alinhamento, deveriam ser maiores ou iguais a 20.

O clustering `TS` apresentou 22.840 SNPs e 1.550 INDELs, enquanto o nosso apresentou 23.901 SNPs e 2.292 INDELs. Os gráficos das Figuras 6.4 e 6.5 mostram a distribuição de contigs por número de SNPs ou INDELs encontrados. As taxas de SNP/contig obtida pelos dois clustering foram idênticas: $3,04$ SNP/contig. As taxas de INDEL/contig apresentaram diferenças: $0,21$ INDEL/contig para o clustering `TS` e $0,28$ INDEL/contig para o clustering `BD` (um aumento de $33,33\%$).

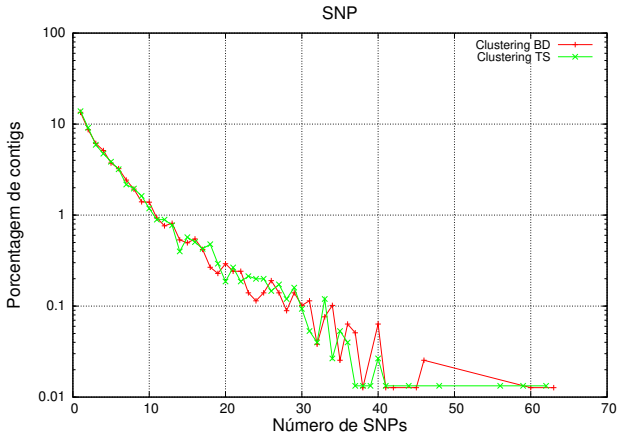

Figura 6.4: Distribuição dos contigs em função do número de SNPs encontrados neles (apenas contigs formados por quatro ou mais seqüências foram considerados). Um SNP foi anotado para toda posição de alinhamento de seqüência do contig que tivesse pelo menos duas bases com qualidades maiores ou iguais a 20 que diferiam da base do consenso. Os número de seqüências alinhadas na posição deveria ser maior ou igual a 4. Não são permitidos mais de um SNP dentro de uma janela de 5 bases.

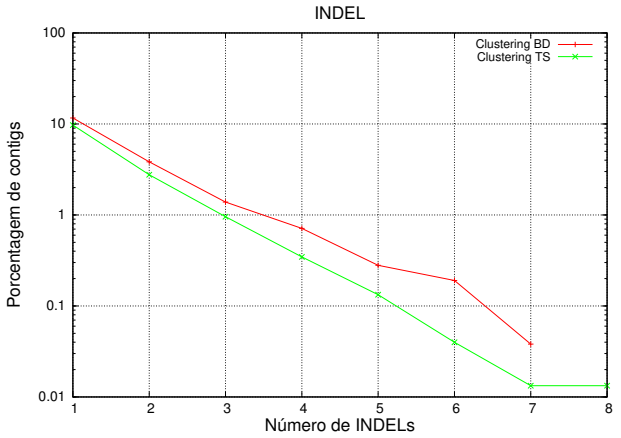

Figura 6.5: Distribuição dos contigs em função do número de INDELS encontrados neles (apenas contigs formados por quatro ou mais seqüências foram considerados). Os INDELs foram detectados como séries de posições adjacentes de alinhamentos que possuíssem, entre as seqüências participantes, pelo menos duas que estivessem marcadas com gap ao invés de uma base. Além disso, as qualidades das bases das outras seqüências, na mesma posição do alinhamento, deveriam ser maiores ou iguais a 20.

Realizamos também a análise de SNPs e INDELs apenas nos contigs que apresentaram composições de seqüências preservadas nos dois clusterings. No total, 3.108 contigs com quatro ou mais seqüências participantes foram analisados em cada clustering. Estes contigs apresentaram tamanhos médios de $1.048,56 \pm 398,25$ bp, no clustering `TS`, e $1.026,77 \pm 401,92$, no clustering `BD`.

Os números totais de SNPs, encontrados nos clusterings `TS` e `BD`, foram de 5.840 ($1,88$ SNP/contig) e 5.371 ($1,73$ SNP/contig), respectivamente. A Tabela 6.3 mostra as distribuições de contigs (quantidade e porcentagem) conforme o número de SNPs contidos neles para os clusterings `TS` e `BD`. Os números de INDELs encontrados foram de 447 ($0,14$ INDEL/contig) para o clustering `TS` e 498 ($0,16$ INDEL/contig) para o clustering `BD`. A distribuição dos contigs (quantidade e porcentagem) conforme o número de INDELs contidos neles pode ser vista na Tabela 6.4.

Quando observamos os dados dos contigs de mesma composição de seqüências nos dois clusterings, podemos ver que no clustering `BD` a taxa de SNP/contig tende a ser menor que no clustering `TS`. A taxa de INDEL/contig, por sua vez, possui tendência contrária. Uma possível causa para este fenômeno pode estar relacionada com a maior qualidade média e com o menor tamanho médio das seqüências. Um programa de clusterização como o CAP3, por exemplo, tende a evitar o alinhamento de duas bases de alta qualidade e nucleotídeos diferentes. Este tipo de alinhamento ocorre somente quando existe uma cobertura de seqüências, na região em torno da posição, suficientemente grande para que o clusterizador considere a união das duas bases como sendo uma possibilidade válida. No caso de nossas seqüências, acreditamos que o tamanho menor diminui a chance de uma posição ser coberta por várias seqüências e que a alta qualidade impede que bases diferentes sejam alinhadas resultando, assim, em um número maior de ocorrências de INDELs e um número menor de ocorrências de SNPs.

6.3.2 Conclusão

Os resultados das análises nos mostram que o nosso método de detecção e remoção de artefatos apresenta vantagens quando comparado com o método utilizado no projeto SUCEST.

O clustering construído com as seqüências processadas pelo nosso método apresenta menos erros de consistência interna e externa. Além disso, os clusters montados parecem ser melhores devido à menor redundância apresentada entre eles.

Contudo, os clusters possuem consensos de tamanhos menores, o que reduz o número de hits full-length. Isso pode gerar a necessidade de um aumento no número de ESTs seqüenciados. Outra alternativa interessante para aumentar o número de clusters full-length é combinar o seqüenciamento de ESTs com a técnica ORESTES [22].

Número de SNPs	Contigs Clustering TS		Contigs Clustering BD	
	Quantidade	Porcentagem	Quantidade	Porcentagem
1	472	15,19%	430	13,84%
2	310	9,97%	281	9,04%
3	153	4,92%	162	5,21%
4	135	4,34%	141	4,54%
5	86	2,77%	84	2,70%
6	81	2,61%	71	2,28%
7	51	1,64%	47	1,51%
8	35	1,13%	34	1,09%
9	27	0,89%	27	0,87%
10	18	0,58%	24	0,77%
11	11	0,35%	9	0,29%
12	13	0,42%	12	0,39%
13	7	0,23%	7	0,23%
14	5	0,16%	4	0,13%
15	10	0,32%	8	0,26%
16	6	0,19%	3	0,10%
17	5	0,16%	5	0,16%
18	3	0,10%	4	0,13%
19	5	0,16%	0	0,00%
20	1	0,03%	4	0,13%
21	3	0,10%	1	0,03%
22	2	0,06%	2	0,06%
23	3	0,10%	1	0,03%
24	2	0,06%	1	0,03%
26	8	0,26%	5	0,16%
27	0	0,00%	2	0,06%
29	1	0,03%	0	0,00%
30	1	0,03%	0	0,00%
31	1	0,03%	1	0,03%
32	1	0,03%	1	0,03%
33	3	0,10%	1	0,03%
34	1	0,03%	2	0,06%
35	1	0,03%	1	0,03%
36	1	0,03%	2	0,06%
37	0	0,00%	1	0,03%
38	1	0,03%	0	0,00%
39	1	0,03%	0	0,00%

Tabela 6.3: Distribuição dos contigs dos clusterings TS e BD conforme o número de SNPs contidos neles. Foram analisados 3.108 contigs que apareceram com mesma lista de seqüências nos dois clusterings e eram compostos por quatro ou mais ESTs.

Número de INDELs	Clustering TS		Clustering BD	
	Contigs		Contigs	
	Quantidade	Porcentagem	Quantidade	Porcentagem
1	226	7,27%	256	8,24%
2	55	1,77%	57	1,83%
3	23	0,74%	19	0,61%
4	5	0,16%	11	0,35%
5	3	0,10%	3	0,10%
6	0	0,00%	2	0,06%
7	1	0,03%	0	0,00%

Tabela 6.4: Distribuição dos contigs dos clusterings TS e BD conforme o número de INDELs contidos neles. Um total de 3.108 contigs, compostos por quatro ou mais seqüências e que apresentaram mesma lista de seqüências participantes nos dois clusterings, foram analisados.

O efeito da diminuição do tamanho das seqüências foi causado principalmente pela etapa de detecção e remoção de baixa qualidade, muito mais exigente que a utilizada no método de Telles e da Silva. Isso levanta a possibilidade da realização de uma análise futura para verificação do efeito da variação do nível de exigência, quanto à qualidade das bases no processo de remoção de baixa qualidade, sob os resultado final do procedimento de detecção e remoção de artefatos.

Capítulo 7

Conclusões e Trabalhos Futuros

Motivados pela importância que a detecção e remoção de artefatos possui na análise de seqüências ESTs, trabalhamos ao longo da dissertação com o objetivo de construir um conjunto eficiente de procedimentos de detecção e remoção de artefatos.

Este conjunto baseou-se em uma nova estratégia, apresentada no Capítulo 3, que determina a identificação dos artefatos em etapas independentes. A independência das etapas visa a diminuição da freqüência de falsos negativos que podem ocorrer quando etapas dependem dos resultados de outras.

O conjunto de procedimentos foi criado a partir da simplificação de várias etapas do conjunto utilizado no projeto SUCEST. Apesar de serem mais simples, as etapas foram capazes de encontrar todos os tipos de artefatos nos testes realizados com seqüências obtidas do projeto Cattle EST. A análise da nova estratégia mostrou que ela possui bons resultados pois, foi capaz de remover corretamente artefatos que seriam perdidos por métodos que exigissem dependência entre etapas.

No Capítulo 4 realizamos um estudo aprofundado sobre artefatos de derrapagem. Propomos três novos algoritmos: Média Aritmética, Média Geométrica e Cobertura por Ecos. Os três algoritmos trabalham de forma a avaliar a ocorrência de grupos ecoados dentro de uma determinada região da seqüência. Nós estudamos duas possibilidades de construção destas regiões: sufixos ou subseqüências.

As análises dos testes feitos com as seqüências do projeto SUCEST nos permitiram concluir que o algoritmo de Cobertura por Ecos trabalhando em busca de subseqüências é o que apresenta melhores resultados.

Concluído o estudo sobre artefatos de derrapagem, nós iniciamos o estudo sobre artefatos de baixa qualidade. Este estudo, que é apresentado no Capítulo 5, analisou duas variações principais de algoritmos: Janela Deslizante e Subseqüência Máxima. Estas duas opções foram testadas a partir da variação dos valores de seus parâmetros. Além de variar os parâmetros, nós implementamos extensões dos algorimos para detecção de "ilhas" de

baixa qualidade existente dentro das seqüências.

A partir da avaliação da qualidade média das seqüências e da preservação de hits de BLAST nós decidimos optar pelo algoritmo de Subseqüência Máxima com a extensão para detecção de ilhas de baixa qualidade.

Finalmente, no Capítulo 6 agrupamos todo o estudo realizado nos capítulos anteriores e implementamos um conjunto completo de procedimentos de detecção e remoção de artefatos. Este conjunto foi testado com as seqüências do projeto SUCEST e seus resultados comparados com os do conjunto utilizado no projeto e desenvolvido por Telles e da Silva.

Os testes mostraram que o conjunto produz seqüências que promovem a criação de clusterings com melhores consistências externas e internas e menor redundância. Contudo, devido à maior exigência de qualidade, as seqüências possuem menores tamanhos médios, o que gera a redução do número de clusters full-length.

Assim, o nosso conjunto é indicado para projetos que exigem clusterings mais confiáveis quanto às consistências externas e internas. Para o aumento da obtenção de clusters full-length, o nosso método, como proposto, exige o seqüenciamento de um número maior de ESTs. Contudo, alguns parâmetros de qualidade podem ser levemente relaxados para permitir o aumento do tamanho médio das seqüências. Outra alternativa é a complementação do seqüenciamento de ESTs com a utilização da técnica ORESTES.

As principais contribuições desta dissertação são:

- Desenvolvimento e validação de uma nova estratégia de detecção de artefatos;
- Proposta e avaliação de novos algoritmos para detecção de artefatos de derrapagem e determinação do método mais adequado;
- Estudo aprofundado sobre a remoção de artefatos de baixa qualidade que determinou a escolha de um algoritmo e de valores de parâmetros adequados a esta finalidade;
- Criação e validação de um conjunto completo de procedimentos para detecção e remoção de artefatos baseados nos resultados obtidos ao longo do desenvolvimento da dissertação.

O trabalho produzido durante a escrita desta dissertação possui várias possibilidades para futuras extensões. Por exemplo, nós estudamos profundamente apenas dois tipos de artefatos. Contudo, isso não significa que as etapas de detecção e remoção estudadas sejam perfeitas. Avaliações mais detalhadas de cada uma delas são pontos importantes para a evolução do conjunto de procedimentos.

O estudo de derrapagem fixou o valor do parâmetro `minimum_number_of_echoes` em 8 e não realizou uma análise mais detalhada do parâmetro `minimum_echo_size`. A variação

destes parâmetros pode ser interessante para a descoberta de uma configuração mais eficiente ou, até mesmo, para a escolha de um algoritmo diferente do Cobertura por Ecos.

Apesar de nosso método de detecção de baixa qualidade ser um pouco menos exigente, em relação às qualidades das bases, que o programa LUCY, ele é muito mais exigente que o algoritmo de janela deslizante utilizado no projeto SUCEST. Esta maior exigência está ligado às imposições que nós fizemos em relação as probabilidades de erro máximas permitidas nas seqüências. Como nós observamos nos resultados, a maior qualidade média das seqüências resulta em menor tamanho médio. Um trabalho interessante seria analisar o efeito da variação da exigência de qualidade sobre o clustering produzido. Este estudo possivelmente mostrará um ponto de equilíbrio entre qualidade, consistência e número de clusters full-length.

Durante os testes, nós utilizamos seqüências de boi e de cana-de-açúcar. No entanto, realização de testes com seqüências de outros organismos serveriam para confirmar a eficiência do método ou detectar situações em que ocorrem falhas.

Finalmente, durante a análise dos clusterings, identificamos que os algoritmos de clusterização utilizados apresentavam problemas para agrupar determinados tipos de seqüências como, por exemplo, as oriundas de genes polimórficos.

Genes polimórficos produzem seqüências de bases que possuem diferenças de uma ou mais bases. Em diversos casos, as seqüências possuem repetições formadas por números diferentes de cópias de pequenos trechos e, devido a esta característica, elas são freqüentemente separadas em clusters diferentes. Seria interessante que o algoritmo de clusterização fosse capaz de agrupar as seqüências de genes polimórficos em um mesmo cluster, indicando inclusive a ocorrência de polimorfismo no cluster, para facilitar a análise dos dados.

Outro problema é tratamento de seqüências quiméricas. Estas seqüências, formadas por trechos de diferentes genes fazem com que o clusterizador agrupe, em um mesmo cluster, seqüências provenientes de dois genes. Assim, uma extensão interessante aos algoritmos de clusterização, seria a capacidade de detecção de tais ocorrências para descarte das quimeras e construção de melhores clusters.

Apêndice A

Revisão Bibliográfica

Nesta revisão bibliográfica iremos extrair os objetivos de cada artigo e citar os principais tópicos de interesse para o nosso trabalho. A escolha dos textos foi feita de modo que eles englobassem diversos aspectos relacionados, direta ou indiretamente, ao nosso estudo, visando, assim, a aquisição de um conhecimento mais amplo sobre o contexto da pesquisa e sobre as soluções existentes para os problemas ligados ao processamento de seqüências genômicas.

A.1 Trimming and clustering sugarcane ESTs [104]

O procedimento de clustering adotado inicialmente pelo projeto SUCEST apresentava diversos problemas como, por exemplo, o número excessivo de clusters e a presença de seqüências provenientes de RNA ribossomal. Isto fez com que o procedimento de clustering fosse refeito, assim como o procedimento de trimagem. O objetivo deste artigo é descrever as novas estratégias de trimagem e clustering adotados no projeto.

O projeto Surgacane EST project (SUCEST) produziu 291.689 ESTs. Para a análise desta grande quantidade de seqüências o processo de clusterização foi importante para que seqüências do mesmo transcrito pudessem ser agrupadas e para que seqüências representativas de cada grupo fossem obtidas, permitindo que a redundância das seqüências fosse avaliada durante e ao final do projeto.

Assim como em outros projetos ESTs, o processo de seqüenciamento produziu seqüências que possuíam trechos indesejados, tais como, poli-A, regiões de baixa qualidade, fragmentos de vetor e adaptadores, e derrapagem. Além disso, algumas seqüências se originaram de RNA ribossomal ou de organismos contaminantes. Trechos como esses introduzem similaridade que não possuem relevância no processo de clusterização, tornando a remoção destes essencial para a criação de bons clusters.

A trimagem é o processo de refinamento das seqüências e neste projeto foi dividida

em várias etapas. A primeira etapa executada foi a remoção das seqüências provenientes de RNA ribossomal. Isto foi feito através da comparação das seqüências contra um banco de seqüências de RNA ribossomal utilizando o programa BLAST [5]. Um e-value menor que 10^{-10} foi utilizado para descartar as seqüências.

A segunda etapa da trimagem foi a remoção de seqüência de vetores e de adaptadores. Utilizando o programa cross_match [44], a seqüência era processada de modo que os trechos similares às seqüências dos vetores e adaptadores utilizados no seqüenciamento fossem substituídos por X. Assim, analisando as regiões marcadas com X foi possível identificar os trechos de vetores e adaptadores e removê-los.

Durante a remoção dos trechos marcados com X, qualquer trecho de poli-A que obtivesse uma pontuação de pelo menos 8 em um alinhamento com uma seqüência de prova composta apenas por adeninas e que estivesse no máximo a 10 bases de distância de um trecho de Xs era identificada como poly-A e removida. Trechos de poly-T também forão procurados e removidos. O alinhamento foi feito com o programa swat [44] e o esquema de pontuação escolhido foi de 1 para cada coincidência, -2 para cada diferença e -8 para cada gap aberto.

O passo seguinte foi a remoção de seqüências de baixa qualidade. Em cada uma das extremidas, uma janela deslizante de 20 bases percorria a seqüência em busca de um trecho que possuísse no máximo 12 bases com qualidade abaixo de 10.

Feito isto, os trechos de derrapagem eram removidos, e a seqüência analisada mais uma vez para identificação de poly-As e poly-Ts.

Finalmente, a contaminação da seqüência era verificada através do BLAST das seqüências trimadas contra as seqüências de organismos contaminantes. Uma seqüência era considerada contaminada se o BLAST indicasse uma correspondência de pelo menos 100 bases e uma similaridade maior que 90%.

O processo de clusterização foi feito com a utilização de programas montadores de fragmentos. As seqüências processadas pelo método de trimagem antigo foram clusterizadas com o programa phrap [44] utilizando os seguintes parâmetros (penalty -15, bandwith 14, minscore 100, shatter_greedy), produzido assim, 81.223 clusters (41.582 singletons). As seqüências processadas pelo novo método de trimagem foram clusterizadas de três formas diferentes. Nas duas primeiras formas o programa phrap foi utilizado com os seus parâmetros padrões e com os parâmetros listados acima, que são mais estritos. Na terceira forma foi utilizado o programa CAP3 [54] com seus parâmetros padrões. O phrap com parâmetros padrões produziu 69.381 clusters (32.202 singletons) e com os parâmetros mais estritos produziu 49.706 clusters (1.535 singletons). O clustering feito com o programa CAP3 produziu 43.141 clusters (16.838 singletons).

Para avaliar a consistência interna destes clusterings, os clusters com duas ou mais seqüências foram analisados para a identificação de seqüências discrepantes. Neste pro-

jeto, uma base foi considerada discrepante quando discordava da base equivalente na seqüência consenso e possuía uma probabilidade menor que 2% de erro na identificação feita pelo programa phred. Assim, uma seqüência era x% discrepante se possuía x% bases discrepantes. A análise mostrou que o clustering produzido com o CAP3 possuía um número menor de seqüências discrepantes. A análise mostrou também que entre os clustering produzidos com os parâmetros mais estritos do phrap, o clustering que utilizou as seqüências trimadas com o novo procedimento apresentou menor discrepância.

A avaliação da consistência externa destes clusterings foi feita através do BLAST de cada seqüência consenso contra todas as outras seqüências consenso do mesmo clustering. Isso foi feito para identificar clusters que possuíam consensos que se sobrepunham em 200 bases ou mais e possuíam identidade alta (75% ou mais), indicando que poderiam ter sido agrupados em um único cluster. Neste teste o clustering produzido pelo CAP3 mostrou-se melhor devido à menor produção de clustering redundantes.

A.2 Analysis and Functional Annotation of an Expressed Sequence Tag Collection for Tropical Crop Sugarcane [113]

Com o objetivo de aumentar o conhecimento sobre a complexidade do genoma da cana-de-açúcar, um projeto EST de larga-escala foi desenvolvido. Neste projeto, mais de 260.000 cDNA clones foram parcialmente seqüenciados a partir de 26 bibliotecas diferentes. Este trabalho faz uma apresentação dos resultados obtidos pelo projeto através da análise dos clusterings produzidos.

A partir de um conjunto de bibliotecas de diferentes tecidos da cana-de-açúcar, 291.689 ESTs foram produzidos. Após o processo de trimagem, um conjunto de 237.954 seqüências contendo um mínimo de 140 bases com qualidade phred maior ou igual a 20 foram separadas para o processo de clusterização. Neste projeto não foram separadas seqüências mitocondriais ou de cloroplastos devido à pequena quantidade de seqüências deste tipo.

O processo de clusterização foi feito com o programa CAP3. Um total de 221.616 ESTs foram agrupadas em 26.803 clusters e 16.338 ESTs permaneceram como singletons totalizando um conjunto de 43.141 seqüências consensos.

Foram identificadas 22.378 ESTs pares (44.756 seqüências) representando trechos das duas extremidades (3' e 5'). Destes, 60% foram reunidos no mesmo clusters, enquanto o restante não foi agrupado provavelmente pelo fato de fazerem parte de cDNAs muito longos.

Através do BLASTX [5] das 43.141 seqüências consensos contra o banco nr de proteínas não-redundantes utilizando um e-value limite de 10^{-40} foi possível observar que mais que

33% dos clones possuíam o inserto completo. Além disso, 65% possuíam semelhanças com proteínas conhecidas quando se utilizou o programa TBLASTX [5] e um e-value limite de 10^{-5}.

A estimativa do nível de redundância destas seqüências foi feita com a comparação de uma seqüência com a outra utilizando o programa cross_match com parâmetros que indicassem que duas seqüências teriam origem no mesmo transcrito se elas possuíssem 98% de identidade em um trecho de pelo menos 100 bases. Este procedimento mostrou uma redundâcia de aproximadamente 22%, indicando que 33.620 genes expressos foram identificados.

Uma comparação entre as 43.141 seqüências consensos e os genomas da *Arabidopsis* e do arroz foi feita utilizando o TBLASTx com e-value limite de 10^{-5}. O resultado obtido mostrou que 71% e 82% das seqüências possuem um nível de semelhança significante com os genomas da *Arabidopsis* e do arroz, respectivamente.

A partir deste conjunto final de seqüências, uma série de estudos foi conduzida para a identificação de polimorfismos em um determinado conjunto de genes, realização da anotação funcional dos ESTs, identificação de domínios protéicos Pfam, identificação de genes expressos exclusivamente por determinados tecidos, entendimento da regulagem da expressão gênica e da tradução de sinais e identificação dos genes relacionados ao sistema de defesa.

A.3 DNA sequence quality trimming and vector removal [26]

Em geral, as seqüências de DNA provenientes das máquinas de seqüenciamento automático possuem erros de identificação de bases, e para fazer um bom uso destas seqüências é preciso processá-las antes. Contudo, o processamento envolve a comparação destas seqüências com outras causando, desta maneira, um dilema, pois, como dito anteriormente, as seqüências possuem erros e não são confiáveis. O objetivo deste artigo é descrever a estratégia utilizada pelo programa LUCY, utilizado pelo TIGR (The Institute of Genomic Research), para o tratamento deste tipo de dados.

O programa LUCY foi desenvolvido para levar em consideração em seu processo de limpeza das seqüências os valores individuais de qualidade de cada base com o objetivo de obter uma seqüência com o melhor valor médio de qualidade. Os valores de qualidades individuais podem ser estimados através do uso de programas de base-calling como o phred [44] e o TraceTuner [108]. Os valores individuais de qualidade de cada base são geralmente confiáveis, mas uma seqüência pode conter valores muito diferentes de qualidades entre suas bases.

O primeiro passo do software é determinar qual é a mais longa e contínua região de alta qualidade de uma seqüência. Algumas bases dentro desta região podem estar erradas, mas a maioria (97,5%) deve ter valores maiores que um valor mínimo especificado pelo usuário.

Para realizar esta operação o programa converte os valores de qualidade para probabilidade de erros segundo a formula $Q = -10 \times \log_{10}(\textit{probabilidade de erro})$. Os valores de erros serão usados para calcular os valores médios de probabilidade de erro dentro de janelas de diversos tamanhos que correm ao longo da seqüência.

Como o início e o final da seqüência são em geral de baixa qualidade, o programa age de forma a identificar estes trechos primeiro. A partir da ponta esquerda da seqüência uma janela de tamanho 10 percorrerá a seqüência até encontrar um trecho que tenha uma probabilidade de erro menor ou igual a 2%. O mesmo será feito na ponta direita da seqüência. Estes trechos identificados são removidos e o restante passará pelas outras etapas do processo. Se a seqüência inteira não passar no teste, ela será inteiramente descartada.

A seguir, o programa opera de modo a identificar os trechos que possuem taxas de erros altas eliminando-os. Nesta etapa, duas janelas são utilizadas. A primeira tem tamanho 50 e um valor limite de probabilidade de erro igual a 8%. A segunda tem tamanho 10 e um valor limite de probabilidade de erro igual a 30%. A primeira elimina os trechos grandes de baixa qualidade, enquanto a segunda elimina os trechos pequenos que não são removidos pela primeira.

A primeira janela percorre a seqüência resultante do primeiro passo de limpeza. A partir do início desta seqüência, o programa calcula para a janela o valor médio de probabilidade de erro. Se o valor estiver dentro do limite, a janela será adicionada à seqüência candidata. Esta seqüência continuará crescendo enquanto existirem janelas consecutivas com valores médios dentro do limite. Se o valor estiver fora do limite, a seqüência candidata será terminada e separada para o próximo passo. A janela continuará a percorrer o resto da seqüência até o final. Caso uma nova janela apresente valor dentro do limite, uma nova seqüência candidata será iniciada.

Cada seqüência candidata será percorrida pela segunda janela seguindo o mesmo critério. Após este processo, todas as seqüências candidatas com tamanho menor que o mínimo (100) serão descartadas. Dentre as seqüências restantes, aquela que tiver um uma probabilidade de erro geral menor ou igual que 2,5% e probabilidades de erros nas extremidades menor que 2% será a seqüência final. A probabilidade de erros nas extremidades é avaliada com as duas últimas bases de cada ponta. No caso raro de mais de uma seqüência atender ao critério, a maior será mantida.

Após este processo de trimagem de qualidade o programa, opcionalmente, faz a extensão da seqüência consenso. Para isso, ele precisa de uma segunda seqüência obtida

para o mesmo cromatograma por um programa que utiliza um algoritmo diferente de base-calling. O programa considera que se algoritmos diferentes de base-calling concordam com a determinação de uma base, está base deve estar correta, mesmo que o valor de qualidade seja baixo. Desta maneira, utiliza-se os dados de qualidade do segundo algoritmo para tentar extender a região de alta qualidade através de um alinhamento entre as duas seqüências.

Note que os algoritmos de base-calling devem ser diferentes, pois algoritmos iguais ou muito semelhantes irão fornecer seqüências semelhantes. Isso acaba por reforçar trechos de baixa qualidade fazendo com que eles sejam incluídos na seqüência final diminuindo portanto a qualidade da trimagem.

Como as seqüências são muito semelhantes em suas regiões de alta qualidade, ao invés de executar um programa baseado em programação dinâmica que consome muito tempo, LUCY utiliza um mecanismo mais rápido. A primeira seqüência é dividida em pequenos trechos de tamanho 16. Os trechos são ordenados e os que forem repetidos são removidos do conjunto. Cada trecho possui um índice que indica a posição na seqüência. A segunda seqüência é seqüêncialmente dividida em trechos de 16 bases e cada trecho é procurado no conjunto da primeira. Se o trecho é encontrado os índices são comparados e a diferença entre eles é salva em um conjunto associada a um contador. Se um valor de diferença já existe no conjunto o contador associado é incrementado. No final do processo, o valor que tiver o maior número de coincidências indicará o deslocamento relativo entre as duas seqüências.

Com a posição relativa conhecida o alinhamento entre as regiões de melhor qualidade pode ser facilmente encontrado. Para as regiões de menor qualidade o programa realiza um procedimento de busca em profundidade para localizar as regiões de maiores afinidades a partir dos trechos finais da região de melhor qualidade. Para cada imcompatibilidade entre as duas seqüências, o algoritmo pode tomar três decisões: pular uma base na primeira seqüência, pular uma base na segunda seqüência, ou, pular as duas. A busca continuará a encontrar mais regiões de alta afinidade e a fazer novas escolhas a cada diferença encontrada até que uma das condições de parada seja alcançada: existem bases compatíveis em número suficiente ao longo do caminho de busca para que a incompatibilidade anterior seja tolerada; o final de cada seqüência foi alcançado; existem mais de 5 diferenças no caminho de busca; ou mais de 100 bases a partir da primeira diferença foram percorridas sem encontrar um número de coincidências suficiente que compensassem as diferenças encontradas. A vantagem do algoritmo de busca em relação a programação dinâmica é a quase linearidade do tempo de execução. A desvantagem é que este método não garante que o alinhamento mais longo seja encontrado.

O segundo passo da execução do programa é a remoção dos trechos de vetor e adaptador da seqüência. Para isso, o software precisa dos trechos de vetor e adaptador onde o

inserto é fixado (splice sites upstream e downstream).

Como o trecho de vetor está normalmente no início da seqüência onde geralmente a qualidade é baixa, uma comparação simples que busca pelo alinhamento mais longo pode não encontrar todos os trechos de vetor devido aos erros de base-calling. O programa faz, então, uma busca adaptativa pelos valores médios de qualidades das bases. Nas regiões de baixa qualidade o programa permite que pequenos trechos de vetor sejam identificados enquanto em regiões de melhor qualidade apenas trechos maiores são identificados.

Devido às diferentes regiões de qualidades existentes no início da seqüência o software considera três critérios diferentes. A busca é feita em áreas de 40, 60 e 100 bases com comprimentos mínimos de alinhamento de 8, 12 e 16 bases. Um alinhamento local ótimo dentro de cada área deve ter pelo menos o comprimento mínimo para ser considerado como vetor. Note que estas janelas são colocadas no início da seqüência original para evitar que fragmentos de vetores sejam perdidos em seqüências que possuem um trecho de baixa qualidade muito longo no início.

O splice site upstream será procurado nas primeiras 200 bases. LUCY procurará pelo maior alinhamento com pelo menos 3 bases corretas para cada base incompatível, o que não significa que haverá 25% de erro porque apenas o alinhamento com maior pontuação local é utilizado. Alinhamentos menores à esquerda podem ser ignorados. Se ainda existirem bons alinhamentos após o melhor, o programa continuará a busca até que todos os fragmentos sejam identificados. Depois de terminar a busca pelo splice site upstream, o dowstream será procurado utilizando-se o critério de 16 bases. Isto é feito para o caso de pequenos insertos.

Após a trimagem de vetor o programa trabalha na remoção de poli-A/T. A busca é feita por trechos que tenham tamanho de no mínimo 10 bases numa janela de busca inicial de 50 bases. São permitidos no máximo três bases incompatíveis entre cada trecho de 10 Ts ou As.

O último passo é a detecção de contaminantes. O software busca apenas pela contaminação pelo próprio vetor. Ele utiliza a seqüência completa do vetor e realiza um procedimento semelhante ao da extensão da seqüência para encontrar a contaminação. Ele cria uma coleção de trechos de vetor com tamanho de 10 bases. Fragmentos para o complemento reverso da seqüência do vetor também são produzidos. Os fragmentos repetidos são removidos. A trecho de boa qualidade da seqüência é dividido em fragmentos e são procurados pela coleção de fragmentos de vetor. Se pelo menos 20% da seqüência tiver correspondência nos fragmentos de vetor, ela será considerada contaminada.

A.4 Informatics for Efficient EST-based Gene Discovery in Normalized and Subtracted cDNA Libraries [87]

Este relatório técnico produzido pelo Departaments of Electrical and Computer Engineering, Pediatrics, and Physiology da Universidade de Iowa tem o objetivo de descrever as estratégias utilizadas em seus projetos de descoberta de genes.

O processo de descoberta de genes baseado em EST é uma estratégia eficiente para a definição do índice gênico de um organismo. Um índice gênico é uma coleção não-redundante de seqüências (todas as seqüências derivadas do mesmo gene são agrupadas juntas). Ele é útil para a realização de anotação e essencial em diversas análises como, por exemplo, a comparação de genomas de espécies diferentes. Além disso, uma coleção não-redundante de cDNA é útil na criação de microarrays.

A tecnologia EST baseia-se no seqüenciamento de bibliotecas de cDNA, que são essencialmente cópias parciais de DNA de mRNA transcritos. Estes fragmentos são seqüenciados em um seqüenciador, que produz uma série de arquivos binários chamados cromatogramas. O programa phred é utilizado para extrair a seqüência de nucleotídeos e os valores de qualidade por base a partir dos cromatogramas.

ESTs costumam conter trechos comuns em suas seqüências, tais como: trechos do vetor utilizado na clonagem, sítios de restrição, poli-A/T, etc. Além destes trechos, os ESTs dos projetos conduzidos por este departamento possuem um pequeno trecho de seqüência sintético chamado oligo tag entre o sítio de restrição e o poli-T para a identificação do tecido de origem da seqüência quando uma biblioteca é feita de diversos tecidos.

Em um projeto de descoberta de genes, normalmente é indesejável o seqüenciamento de clones de genes que já foram seqüenciados anteriormente. Desta forma, as técnicas de normalização e subtração são aplicadas na criação das bibliotecas para a manutenção de altas taxas de eficiência dos projetos. A normalização opera de forma a equilibrar a quantidade de cada clone, eliminando assim a predominância de certos ESTs. A subtração é uma técnica que visa a remoção de clones que já são conhecidos do conjunto que será seqüenciado.

O departamento desenvolveu uma série de procedimentos que ditam o processamento e a anotação dos ESTs. Estes procedimentos foram feitos para suportar todos os tipos de bibliotecas de cDNA - single-tissue ou pooled, normalized ou subtracted - para qualquer espécie.

Os procedimentos são divididos em 5 fases principais:

1. Obtenção de dados e arquivamento.

2. Avaliação de qualidade e trimagem

3. Anotação das seqüências

4. Avaliação de singularidade

5. Depósito ou submissão nas bases de dados públicas.

Na fase 1 é feita a geração da informação que será inserida no software de seqüenciamento ABI. Neste passo as seqüências recebem nomes de acordo com a nomenclatura padrão do departamento. No passo seguinte os cromatogramas são gerados e depositados em locais apropriados.

A nomenclatura padrão dos projetos é composta de 8 valores separados por traços (-). O primeiro valor indica a instituição de origem da seqüência. O segundo e o terceiro valor indicam respectivamente os códigos do projeto e da biblioteca. A placa, a coluna e a linha são os quarto, quinto e sexto valores. O sétimo valor é o número de replicação, sendo que o valor zero é utilizado para as placas masters. O último valor é para identificar a instituição que fez a replicação.

Na fase 2 são realizadas a avaliação de qualidade e a trimagem das seqüências. A avaliação de qualidade é baseado no conjunto de valores de qualidade atribuídas pelo software phred às bases da seqüência. A trimagem em busca de trechos de pouca complexidade é feita de acordo com a extremidade seqüenciada (3' ou 5') e com os detalhes específicos do processo de clonagem.

O programa ESTprep [88] é utilizado para executar uma avaliação inicial da qualidade e a identificação dos trechos comuns aos ESTs. Se o trecho inicial de 20 bases tiver menos que 8 bases com qualidade maior que 20, ele será removido. Se a qualidade média das primeiras 200 bases for menor que 20 a seqüência é descartada.

Em seguida, o programa tenta identificar os sítios de restrição usados durante o processo de clonagem. Primeiro identifica-se um sítio de restrição de alta qualidade e depois valida-se este sítio através da verificação da seqüência de vetor adjacente ao site. A qualidade do sítio de restrição é medida através do número de erros em relação ao sítio de restrição correto.

Nos projetos coordenados pelo departamento, um sítio de restrição de 8 bases é utilizado. Se o sítio não possui 8 bases, um sítio de restrição "sintético" é criado utilizando-se as últimas 8 bases do vetor (incluindo o sítio de restrição) que vêm antes do inserto.

Para seqüências 5' os dois passos acima completam a análise. No entanto, para as seqüências 3' é preciso trimar a cauda e o sinal poly-A e o trecho que identifica o tecido.

Após a determinação do sítio de restrição, a seqüência é percorrida em busca do primeiro nucleotídeo da cauda (tipicamente T para ESTs 3'). A partir desta posição, uma seqüência maximal formada apenas por Ts é construída de tal forma que ela possua similaridade maior que um limite pré-estabelecido (95%) em relação a seqüência original. Desta maneira, a região mais rica em T é encontrada. Se esta região não termina com um

T, ela é retraída em uma base, o que é repetido até que a última base seja um T. Se a cauda de poli-A não tiver tamanho suficiente (10 bases), a busca é refeita começando uma base à direita do ponto de início original. Se a cauda é identificada, a busca é repetida utilizando-se um limite menor (94% do limite original) para evitar o truncamento da cauda de poli-A. Se depois de todos estes passos a cauda não foi identificada, o programa tenta analisar uma janela com o tamanho da distância média entre o sítio de restrição e o trecho que identifica o tecido (18 bases) em busca de uma densidade suficiente de Ts (65%).

Após a identificação do poly-A, o programa tenta localizar os sinais de poliadenilação. Os sinais procurados podem ser canônicos (AAUAAA ou AUUAAA) ou alternativos. Eles devem estar dentro de 11 a 30 nucleotídeos a partir do final da cauda poli-A.

O último item procurado é o trecho que identifica o tecido de origem. Ele deve se localizar entre o sítio de restrição e a cauda poli-A. Um alinhamento local é feito para identificar o trecho. Se a cauda poli-A não foi identificada, a busca não é feita.

O ESTprep encerra a execução com uma segunda avaliação de qualidade. Uma janela deslizante de 20 bases é utilizada para identificar a primeira região com no máximo 8 bases com qualidade menor que 10. O ponto de trimagem na extremidade 3' será o início desta região. Se a seqüência que sobrar tiver tamanho menor que 100, ela será descartada.

A detecção de contaminação e o mascaramento de repeats são feitos com a utilização do pacote RepeatMasker. A vantagem da utilização deste pacote ao invés de um algoritmo de alinhamento alternativo é a sensiblidade na detecção de seqüêcias distantemente relacionadas. O pacote RepeatMasker [44] utiliza o programa cross_match para realizar os alinhamentos.

O tempo adicional ao processo de trimagem devido à utilização do RepeatMasker é significante somente se a base de dados de seqüências utilizada for muito grande, como no caso de genoma bacterial. Neste caso, é possível a utilização de um algoritmo de alinhamento diferente para a identificação de repeats. Note que este algoritmo deve ser sensível. Para a detecção de contaminação um algoritmo menos sensível pode ser utilizado como, por exemplo, o BLAST. RepBase é utilizada como base de dados básica para elementos repetitivos.

A contaminação bacterial ou mitocondrial será identificada se 85% da seqüência corresponder a base de dados bacterial e mitocondrial. A contaminação por vetor é separada em dois casos diferentes. No caso de contaminação completa, o mesmo parâmetro acima é considerado para identificar a contaminação. Outro caso de contaminação por vetor ocorrem em casos de insertos curtos de DNA. Neste caso, utiliza-se um limite menor, devido ao menor tamanho do vetor, e o trecho final do vetor deve estar no máximo a 6 bases do final da seqüência para evitar falsos-positivos. As seqüências completamente contaminadas são descartadas e as seqüências de inserto curto possuem o vetor do final

trimado.

Um último controle de qualidade é feito. Para cada placa seqüenciada 8 clones são escolhidos para serem reseqüenciados. Estas seqüências de verificação serão comparadas com as originais e em caso de diferença o clone será marcado.

A terceira fase é a de anotação: todas as seqüências de alta qualidade e não contaminadas são blastadas contra a base não-redundante de nucleotídeos e aminoácidos do NCBI. As seqüências podem ser blastadas contra ESTs de espécies específicas para prover uma informação inicial para a referência cruzada dentro dos clusters UniGene [15, 90].

Na fase seguinte é realizada a clusterização das seqüências para avaliação do grau de singularidade. O processo de clusterização agrupa as seqüências em clusters baseadas em similaridade. Isto auxilia a avaliação da singularidade e a definição de conjuntos não-redundantes.

O programa UIcluster [109] é utilizado para clusterização. A estratégia deste programa é a de adicionar uma seqüência mascarada de cada vez ao conjunto de clusters. Cada seqüência é comparada com um elemento representativo de cada cluster. Se a seqüência possui similaridade significante com algum elemento, ela é adicionada ao cluster que ele representa. No caso de mais de um elemento possuir similaridade significativa, ela é adicionada ao cluster mais similar, com anotação informando quais foram os outros clusters que também apresentaram similaridade.

Para construção de clustering mais precisos, as seqüências full-length de mRNA disponíveis são adicionadas ao processo de clusterização. No projeto Rat Gene Discovery, desenvolvido pelo departamento, a incorporação da seqüência de mRNA promoveu a união de diversos clusters que seriam criados separadamente por estarem separados por uma significativa fração de mRNA.

O critério utilizado para determinar a similaridade de seqüência mínima é um alinhamento de 19 de 20 bases (95%). Substituição, inserção e deleção são permitidos. O alinhamento mínimo é extendido ao máximo permitido pela homologia. As seqüências podem se equivaler em ambas as direções.

A última fase é a de submissão nas bases públicas. Os dados produzidos pelo projetos são submetidos em bases como o dbEST.

A.5 The TIGR Gene Indices: reconstruction and representation of expressed gene sequences [83]

ESTs forneceram um impulso inicial à coleção de sequências transcritas de uma variedade de organismos. Contundo, uma análise destes dados pode fornercer informação adicional significante sobre função, estrutura e evolução. TIGR Gene Indices, uma análise das

seqüências disponíveis em domínio público, é uma tentativa de identificar genes e prover informação adicional sobre eles. Este índice gênico é construído, para organismos selecionados, através da clusterização e montagem dos ESTs e seqüências gênicas anotadas. Este processo produz um conjunto de seqüências consensos (TC - *tentative consensus*) únicas e de alta fidelidade. O objetivo deste trabalho é explicar o método utilizado para construção destes índices.

A construção de cada TIGR Gene Index é feita com base em uma versão anterior, através da adição de novos ESTs e seqüências gênicas anotadas provenientes, respectivamente, dos bancos dbEST [29, 14] e GenBank [39, 13].

O primeiro passo é a construção de uma base de dados de seqüências gênicas anotadas. Todas as seqüências do GenBank são obtidas e as características CDS e CDS-Join [40] para genes full-length e seqüências de mRNA são extraídos dos registros. Em caso de seqüências redundantes uma seqüência representativa é escolhida, contudo, as informações que ligam aos números de acesso do GenBank são mantidas. A anotação destas seqüências de transcritos expressos (ET - *expressed transcript*) são verificadas para consistência e a informação é registrada no banco TIGR Expressed Gene Anatomy Database - EGAD [33].

ESTs são obtidos diariamente do banco dbEST. As seqüências são processadas para remoção de vetores, seqüências mitocondriais ou ribossomais, trechos de baixa qualidade, poli-A e poli-T.

Os ESTs trimados, as seqüências ET, os consensos TC da versão anterior e seqüências que não foram clusterizadas (singletons) são comparadas em pares para a identificação de sobreposição. As seqüências que possuírem 95% de identidade ao longo de uma região de 40 bases e que possuírem menos que 20 bases incompatíveis em cada final de seqüência serão agrupadas em um cluster.

Cada cluster é então montado separadamente. Se TCs aparecerem em clusters, ESTs componentes e seqüências ET são adicionadas a qualquer novo EST ou seqüências ET. As seqüências clusterizadas são montadas pelo programa CAP3. A montagem produz um ou mais consensos para cada cluster e rejeita seqüências quiméricas, seqüências de baixa qualidade e seqüências que não tiveram sobreposição.

Uma segunda rodada de clusterização e montagem é feita utilizando apenas os novos TCs com o objetivo de eliminar redundâncias criadas no processo.

A anotação funcional é feita com a utilização das seqüências consensos. Como a seqüência consenso é mais longa que as seqüências que constituem os clusters, ela tem mais informação sobre a seqüência codificante de proteína do que um EST individual.

Caso um cluster contenha uma seqüência ET, a anotação desta será dada ao cluster. Se não existir, então o consenso é comparado com um banco não-redundante de seqüências de aminoácidos e com uma seqüência de seqüências de nucleotídeos. Os resultados que tiverem altas pontuações serão registrados e utilizados na anotação.

O índice gênico é construído de forma a manter informação de versões anteriores. Os identificadores não são reutilizados. No caso, de um cluster ser agrupado com outro, um identificador novo é criado e a informação dos clusters eliminados é mantida.

Uma análise feita na criação do TIGR Gene Index da *Arabidopsis thaliana* foi a inclusão de seqüências de genes previstos (PT - *predicted transcripts*). Transcritos previstos são um importante meio de aproximação inicial dos genes que são codificados em organismos, mas eles se tornam menos significantes a medida que dados experimentais se tornam disponíveis. As seqüências PT não podem ser tratadas como seqüências anotadas, mas sua incorporação na análise de transcritos pode prover informações que liguem o EST à seqüência genômica, assim como, produzir dados que aprimorem os métodos de predição de genes.

A inclusão dos PTs foi feita na construção do índice gênico com base no projeto de seqüenciamento do cromossomo 2 da *Arabidopsis thaliana*. A adição de 1158 PTs resultou no aumento do número de TCs e na diminuição de singletons ESTs e ETs. Além desses resultados, a adição dos PTs forneceu uma ligação entre os consensos e a seqüência genômica.

A.6 A comprehensive Approach to Clustering of Expressed Human Gene Sequence: The Sequence Tag Alignment and Consensus Knowledge Base [71]

Muitos sistemas foram desenvolvidos para organizar e talvez enriquecer os ESTs disponíveis em domínio público, e cada um deles utiliza uma aproximação que seja capaz de atender os seus objetivos. Índices como o TIGR Human Gene Index e as bases de dados de ESTs como o UniGene descartam "ruídos" das informações durante suas construções e confiam nos mais longos e informativos ESTs, nos transcritos significantes ou nos éxons genômicos agrupados para alimentar as classes de índices.

Utilizando critérios estritos o método de montagem utilizado nos TIGR Gene Indices procura agrupar seqüências fortemente relacionadas, com um mínimo de quimerismo e contaminação. No entanto, este método baseado na redundância dos transcritos pode gerar consensos "curtos" e eliminar seqüências relacionadas que não entrem dentro do padrão muito estrito.

O UniGene é uma aproximação complementar que agrupa seqüências dentro de clusters baseado na sobreposição de seqüências acima de um limite, aceitando apenas o elemento representativo mais longo de um classe de índice como seu consenso.

O STACK (Sequence Tag Alignment and Consensus Knowledge) é baseado no desenvolvimento de um clustering exaustivo, definindo classes de índices conforme o número total e multiplicidade de palavras de 6 bases, ao invés da utilização do alinhamento com membros de classes previamente identificadas. Esta estratégia aumenta a diversidade dos clusters resultantes para identificação e acentuação das variações. Dada uma árvore de relações entre seqüências de cluster, consensos primários e secundários são gerados para maximizar a detecção de genes, éxons, possíveis parálogos e formas de expressão relacionadas.

Enquanto os métodos de clusterização tendem a minimizar comparações para ganhar velocidade, o método utilizado pelo STACK foi desenvolvido de forma a maximizar o número de comparações. Todos ESTs são comparados 2 a 2, o que requer um hardware de alto desempenho ou processamento distribuído. O benefício disto é que o menor número possível de seqüências são colocadas em um cluster, contribuindo para o valor do consenso.

O primeiro passo do esquema STACK é a seleção de ESTs humanos seguido pela quebra dos arquivos de seqüências, no formato do GenBank, em grupos. Conjuntos individuais de tecidos são organizados de acordo com as relações com os sistemas de orgãos e as seqüências são distribuídas entre estes conjuntos. Se a seqüência possui relação com doenças, ela é duplicada e colocada também em um conjunto de seqüências relacionadas com doenças para facilitar a comparação entre similaridades entre tecidos.

Como o procedimento de clusterização tem a função de agrupar seqüências que compartilham regiões idênticas, é preciso garantir que as seqüências estão livres de seqüências de artefatos comuns aos ESTs em estudo. Todas as seqüências são submetidas ao mascaramento contra seqüências repetitivas humanas utilizando as seqüências de vetores comuns do banco RepBase, de espécies que são potenciais contaminantes como os roedores, de DNA mitocondrial e de DNA ribossomal. Inicialmente o STACK utilizava BLASTN e XBLAST [5] no mascaramento, mas atualmente utiliza o cross_match que é mais sensível.

Para a clusterização o STACK utiliza o software d2_cluster [21, 47] que implementa um algoritmo guloso de clustering. O algoritmo utiliza uma aproximação para a clusterização de seqüências baseada na identificação e contagem de palavras de tamanho n (n = 6 neste trabalho) equivalentes, em contraste à aproximação estrita utilizada pelo TIGR Gene Index, em que as seqüências são agrupadas baseadas na equivalência de fragmentos inteiros de seqüência. Visto que o método rende membros de clusters que são altamente relacionados, a aproximação mais flexível apresenta a oportunidade de detecção de clusters que são relacionados por rearranjo ou splicing alternativo. Embora esta estratégia produza clusters com mais "ruídos", a combinação com uma ferramenta de verificação de alinhamento múltiplo de seqüências como o CRAW reduz o "ruído" e produz uma rede de seqüências altamente relacionadas. Duas seqüências ou seus reversos complementares caem em um mesmo cluster se compartilharem uma janela de 150 bases com pelo me-

nos 96% de identidade. Seqüências com tamanho menores que 50 bases são excluídas do clustering.

O programa phrap é utilizado na montagem. Ele é efetivo mas não é imune de problemas quando executado com ESTs de baixa qualidade. Uma vantagem do phrap é que pode utilizar seqüências de qualidade derivadas dos cromatogramas das seqüências. O uso de cromatogramas normalmente suporta a derivação de um consenso mais longo e exato, mas no esquema de clusterização frouxa utilizado pelo STACK, que gera um número maior de seqüências por cluster, provê a base para a geração de consensos mais longos.

As qualidades da anotação da direção da seqüência, da montagem do cluster e do alinhamento não podem ser garantidas. O phrap invoca uma etapa de alinhamento de seqüência mas não provê subclusters para distinção de splice alternativo ou outro dado cientificamente interessante de problemas de alinhamentos induzidos por seqüências de baixa qualidade. Para tirar proveito das vantagem deste método de clusterização, é necessário fazer um processamento adicional ao alinhamento. Para isso, foram desenvolvidas as ferramentas CRAW e CONTIGPROC.

CRAW é usada para maximizar o tamanho do consenso, partição de sub-montagens e montar artefatos e isoformas. Este software verifica a concordância ao longo das colunas de um alinhamento múltiplo e usa esta informação para ordenar as relacionadas dentro de cada cluster, gerando seqüências consensos para cada subcluster. Um subcluster é gerado se 50% ou mais bases de uma janela de tamanho 100 diferem das seqüências restantes de um cluster, excluindo as 100 bases iniciais de qualquer seqüência. A aproximação depende fundamentalmente da qualidade do alinhamento de cada cluster. Um alinhamento ruim pode render subclusters errados, e uma penalização muito baixa para gap pode render muitas colunas que concordam e assim, não criar os subclusters que deveriam ser criados.

Um procedimento dedicado de particionamento de alinhamento é adicionalmente utilizado para qualificar os alinhamentos. CONTIGPROC independentemente particiona as seqüências alinhadas geradas a partir do consenso CRAW e classifica os consensos de acordo com o número de seqüências atribuídas e com o número de bases atribuídas. O consenso com melhor classificação é escolhido como representativo primário. A determinação da orientação 5' ou 3' do cluster é baseada nos votos das anotações individuais de cada EST. Todos os consensos são rearranjados na orientação de 5' para 3'.

Todos ESTs gerados a partir do mesmo clone de cDNA correspondem a um único gene. Cada EST obtido do GenBank é procurado para obtenção da identificação do clone para rastrear os transcritos correspondentes ao mesmo gene. Esta informação é utilizada para unir clusters que possuam ESTs que compartilham identificadores de clones. Nesta fase, os ESTs que foram separados pelo phrap como singletons podem ser unidos aos clusters. O algoritmo básico consiste na formação de uma fila a partir de um cluster inicial. Para cada EST deste cluster com um identificador de clone, adiciona-se na fila todo cluster

que possua um EST com mesmo identificador. Isso deve ser feito até que nenhum cluster novo seja adicionado.

Para a indexação hierárquica, todos os consensos de clusters e singletons são submetidos como um conjunto único ao d2_cluster. Esta etapa requer um processamento de alta performance devido à grande quantidade de dados. Os clusters resultantes são expandidos pela substituição da seqüência consenso pelas seqüências que contribuíram para a formação dele. Feito isso, os conjuntos obtidos são submetidos aos processos descritos acima a partir do passo da montagem feita com o programa phrap.

O procedimento de clustering é assintoticamente estável na presença de artefatos biológicos e imperfeições do processo de seqüenciamento. A taxa de união é aproximadamente constante em todos os tamanhos de bases de dados e é praticamente não afetada por químicas alternativas no seqüenciamento ou taxas de erros de diferentes fontes de ESTs.

A.7 CAP3: A DNA Sequence Assembly Program[54]

A estratégia de seqüenciamento shotgun tem sido largamente usada em projetos de seqüenciamento de genomas. A fase mais importante nesta estratégia é a montagem de seqüências curtas em seqüências longas. Vários programas de montagem foram desenvolvidos [44, 51, 58, 60, 77, 93, 99, 101]. O contínuo desenvolvimento destes programas é necessário para superação dos desafios impostos pelos diversos projetos genoma.

O programa CAP3 é a terceira geração do programa CAP [51]. Ele possui a habilidade de realizar o corte de regiões 5' e 3' de baixa qualidade. Os valores de qualidade são utilizados para computação de sobreposições das seqüências. Restrições foward-revese são utilizadas para correção de erros de montagem.

O método de montagem do software possui 3 fases principais. A primeira fase é a de identificação e remoção de pontas de baixa qualidade, de cálculo das sobreposições e de eliminação das falsas sobreposições. A segunda fase envolve a união dos seqüências para criação dos contigs em ordem decrescente de pontuação de sobreposições e a utilização das restrições foward-reverse para correções. Na terceira fase, o alinhamento múltiplo das seqüências é construído e a seqüência consenso com suas qualidades é determinada.

Um método rápido para determinação de sobreposição entre duas seqüências foi desenvolvido. Sejam $f_1, f_2, ..., f_n$ todas as seqüências de entrada em uma dada orientação e r_x o reverso complementar da seqüência f_x. O método rapidamente encontra pares de seqüências f_x e f_y com $x < y$ que se sobrepõem, e pares de seqüências r_x e f_y com $x < y$ que se sobrepõem. Uma sobreposição entre as seqüências f_x e f_y é simétrica à sobreposição entre r_x e r_y, e uma sobreposição entre r_x e f_y é simétrica à sobreposição entre f_x e r_y.

Para determinação rápida de seqüências com potencial sobreposição, um alinhamento de sobreposição entre duas seqüências é simplificado como uma cadeia ordenada de pares de segmentos. Cada par de segmentos corresponde a uma porção sem gaps de tamanho suficiente do alinhamento. A cadeia de pares de segmentos de maior pontuação pode ser rapidamente computada utilizando-se uma técnica semelhante a do BLAST. Pares de seqüências possuem uma sobreposição potencial se possuem uma cadeia com pontuação de similaridade maior que um limite. Um modo de se encontrar as seqüências com potencial sobreposição é aplicar esta técnica para cada par de seqüências f_x e f_y com $x < y$ e para cada par de seqüências r_x e f_y com $x < y$, no entanto, uma estratégia mais eficiente é utilizada.

Todas as seqüências $f_1, f_2, ..., f_n$ são concatenadas com um caracter especial inserido entre cada uma. A seqüência resultante é chamada seqüência combinada. Seja a seqüência g um seqüência f_x ou r_x. Cadeias de pares de segmentos de alta pontuação são computadas entre a seqüência g e a seqüência combinada. O caracter especial é utilizado para garantir que uma cadeia consista somente de pares de segmentos provenientes de uma mesma seqüência na seqüência combinada. Para achar a seqüência correspondente f_y na seqüência combinada para uma cadeia, uma busca binária é feita em uma lista ordenada com as posições inicial e final de cada seqüência na seqüência combinada. Note g não é comparada com qualquer seqüência f_y com $x >= y$.

Para cada par de seqüências com uma sobreposição potencial, uma banda mínima de diagonais na matriz de programação dinâmica é determinada para cobrir todas as cadeias entre seqüências de pontuação maior que o limite de corte. Uma diagonal k na matriz de programação dinâmica consiste de todas as entradas (i, j) tais que $j - i = k$ [95]. Um par de segmentos com posição i de uma seqüência e posição j de outra ocorre na diagonal $j - i$. Uma cadeia de pares de segmentos será coberta pela banda se cada par estiver em uma diagonal dentro da banda. A banda de diagonais é utilizada posteriormente para computação eficiente da sobreposição.

Valores de qualidades de bases e de similaridade de seqüências são utilizados para cálculo de posições de corte 5' e 3' para remoção de pontas de baixa qualidade. Qualquer região suficientemente longa de valores de alta qualidade é definida como boa. Adicionalmente, qualquer região suficientemente grande altamente similar a uma região de alta qualidade é definida como boa. A posição de corte 3' de uma seqüência é a máxima das posições finais 3' das boas regiões de um seqüência. A posição de corte 5' de uma seqüência é a mínima das posições finais 5' das boas regiões de um seqüência. Regiões de seqüências com forte similaridade com outras seqüências são localizadas através do cálculo das posições inicial e final de um alinhamento local ótimo para cada par de seqüências.

O algoritmo de alinhamento local de Smith e Waterman [95] é generalizado para utilizar valores de qualidades de bases. As pontuações para acertos, erros e penalidades de

gaps possuem pesos relacionados aos valores de qualidades envolvidos. Sejam m um inteiro positivo associado a pontuação de acerto, n um inteiro negativo associado a pontuação de erro e g um inteiro positivo fator de penalidade de extensão de gaps. Para um acerto de bases com valores de qualidade q_1 e q_2 é dada uma pontuação de $m * min(q_1, q_2)$. Para um erro, é dada a pontuação $n * min(q_1, q_2)$. A uma base de qualidade q_1 em um gap, é dada a pontuação de extensão $-g * min(q_1, q_2)$, onde q_2 é o valor de qualidade da base na outra seqüência imediatamente antes do gap, se existir, caso contrário, utiliza-se a base imediatamente depois. A pontuação de um gap é a soma das pontuações de extensão de cada base no gap menos o valor de penalidade de abertura do gap, que por simplicidade é um valor inteiro independente dos valores de qualidade. A pontuação de similaridade de um alinhamento é a soma das pontuações de cada acerto, erro e gap.

Para seqüências quiméricas as posições de corte 5' e 3' são calculadas segundo o método descrito por Huang [53].

Uma sobreposição entre duas seqüências é definida como o alinhamento global com máxima pontuação de similaridade. A definição de uma sobreposição como um alinhamento global ótimo entre duas seqüências limpas (com as pontas de baixa qualidade removidas), ao invés, do alinhamento local ótimo entre duas seqüências não limpas é útil para detecção de falsas sobreposições. Um alinhamento global ótimo pode mostrar que algumas boas regiões das seqüências não são similares, indicando falsa sobreposição, enquanto o alinhamento local mostra apenas regiões similares.

Para cada par de seqüências com potencial sobreposição, uma banda de diagonais centrada na posição inicial do alinhamento local ótimo, calculado anteriormente, é formada. A banda é duas vezes maior que a banda utilizada no cálculo do alinhamento local ótimo. Uma técnica de divisão e conquista é utilizada para realizar o cálculo em espaço linear [25]. Assim como para o alinhamento local, os valores de pontuação para acertos, erros e gaps são calculados baseados nos valores de qualidade. O comprimento, a pontuação de similaridade e o percentual de identidade da sobreposição serão os mesmos valores obtidos para o alinhamento.

Cada sobreposição é avaliada com 5 medidas. Se ela falhar em qualquer uma delas, não será considerada na construção de contigs. As três primeiras medidas verificam se a sobreposição satisfaz os requisitos mínimos para tamanho, similaridade e identidade.

A quarta medida verifica as diferenças da sobreposição nas bases de alta qualidade. Se uma sobreposição contém um número suficiente de bases de alta qualidade diferentes, então ela deve ser falsa. Sejam b um valor inteiro de corte de alta qualidade e d um valor inteiro de corte de diferença de qualidade. A pontuação da diferença entre duas bases será $max(0, min(q_1, q_2) - b)$. A pontuação de diferença de qualidades da sobreposição será a soma das diferenças. Se esta pontuação exceder d a sobreposição é descartada.

Para a quinta medida, a taxa de diferença da sobreposição é examinada com respeito às

taxas de erro de seqüenciamento das duas seqüências envolvidas. As taxas de erro de cada seqüência são obtidas através do método do vetor de erro [53]. Para toda sobreposição verdadeira a taxa de diferença é próxima à soma das taxas de erros das duas regiões.

O procedimento de utilização de restrições foward-reverse na construção de contigs é dividido em 4 etapas. Na primeira etapa uma disposição inicial das seqüências é feita com a utilização de um método guloso [53]. No segundo passo, a qualidade da disposição corrente é avaliada através da verificação das restrições. No passo 3, a região com maior número de restrições não satisfeitas é localizada de modo que as restrições possam ser satisfeitas através de correções na região. Se esta região existe, as correções são feitas e os passos 2 e 3 são repetidos, caso contrário o procedimento de correção é terminado. No último passo os contigs são ligados com as restrições.

Uma restrição foward-reverse consiste de duas seqüências e dois inteiros que especificam um intervalo de distância entre os seqüências. Uma restrição deste tipo é satisfeita por uma disposição se as duas seqüências ocorrem no mesmo contig, a seqüência upstream na orientação direta, o downstream na orientação reversa e a distância entre as duas seqüências está dentro do intervalo. Uma sobreposição é unused se ela não é usada na disposição corrente. A sobreposição da seqüência f sobre a seqüência g é denotada por $f \rightarrow g$. Uma restrição não satisfeita envolvendo as seqüências h e r é satisfazível por uma sobreposição não utilizada $f \rightarrow g$ se a seqüência f ocorre downstream à seqüência $h(r)$ na orientação direta em um contig, a seqüência g ocorre upstream à seqüência $r(h)$ na orientação reversa, e a soma das distâncias entre $h(r)$ e a ponta 3' do contig e entre a ponta 5' do outro contig e $r(h)$ é menor que a máxima distância da restrição. O valor u determina o número mínimo de restrições não satisfeitas necessárias para indicar um problema com a disposição das seqüências.

No passo 2, toda restrição é verificada na disposição corrente. As restrições satisfeitas são usadas para calcular, para cada sobreposição usada na disposição, o número de restrições satisfeitas que suportam a disposição. As restrições não satisfeitas são divididas em grupos, onde todas restrições em um grupo são associadas com uma sobreposição unused ou com um par de contigs. Para cada restrição não satisfeita que é satisfazível por sobreposições unused, uma sobreposição com máxima pontuação é escolhida e associada a ela. Para cada restrição não satisfeita remanescente, se ela é uma ligação entre dois contigs, ela é associada a estes dois contigs.

No terceiro passo, o grupo com maior número de restrições não satisfeitas é selecionado. Primeiro considera-se o caso em que o grupo está associado a uma sobreposição unused $f \rightarrow g$. Se o número de restrições não satisfeitas é maior que a soma do valor u, do número de restrições satisfeitas suportando a sobreposição utilizada envolvendo f, e do número de restrições satisfeitas suportando a sobreposição utilizada envolvendo g, então a disposição é corrigida através da quebra das sobreposições utilizadas envolvendo f e

g, e unindo f e g utilizando a sobreposição $f \rightarrow g$. Depois, considera-se o caso em que o grupo está associado a um par de contigs. Se o gap entre os dois contigs pode ser fechado utilizando-se as seqüências de outras regiões para tornar a restrição não satisfeita satisfazível, então o fechamento do gap é implementado. Seqüências de outras regiões que são associadas por restrições com seqüências nos dois contigs são usadas para fechar o gap. Se nenhuma correção é feita para o grupo selecionado, o processo é repetido para os outros grupos até uma correção ser feita ou não houver mais nenhum grupo disponível para seleção.

No último passo os contigs são ordenados utilizando as restrições como ligações. No passo 2 são obtidos grupos de restrições não satisfeitas que servem como ligação entre contigs. Os grupos são considerados em ordem decrescente de tamanho de grupo. Seja v um inteiro que indica o número mínimo de restrições para ligar dois contigs. Para cada grupo de restrições que são associadas com um par de contigs, se o número de restrições não for menor que v, e nenhum dos dois contigs estão ligados a outros contigs na ponta correspondente, então a ligação entre os contigs é feita.

Um alinhamento múltiplo de seqüências é utilizado para construção de cada contig. A construção é feita repetitivamente através do alinhamento da próxima seqüência com o alinhamento corrente. As seqüências são considerados na ordem crescente de suas posições no contig. Os valores de qualidades são considerados para produção de um alinhamento mais preciso. Após a construção do alinhamento, uma seqüência consenso é construída e qualidades são atribuídas às suas bases. Para cada coluna do alinhamento, uma soma ponderada dos valores de qualidade é calculada para cada tipo de base e a base com maior soma é tomada como consenso. O valor da qualidade da base do consenso é a soma dos valores de qualidades das bases do mesmo tipo menos a soma dos valores de qualidades dos outros tipos de bases.

A soma ponderada de valores de qualidade é feita da seguinte maneira. Os valores de qualidade são particionados em dois grupos, um para cada fita. Assume-se que os valores de um grupo são independentes dos valores do outro grupo. Os valores de cada grupo são ordenados em ordem decrescente. O i-ésimo valor de cada grupo recebe o peso w_i e a soma das qualidades multiplicadas pelo peso é computada. O conjunto de pesos utilizado é $w_1 = 1$ e $w_i = 0,5$ para $i > 1$.

Quando se alinha uma seqüência com o alinhamento corrente, da maneira descrita acima, a maior parte dele permanece inalterada. Assim, apenas a sua porção 3' é considerada para alinhamento com a seqüência. A porção 3' do alinhamento corrente é substituída pelo alinhamento dela com a seqüência.

Um esquema de pontuação que utiliza os valores de qualidade das bases é utilizada no alinhamento múltiplo entre um bloco do alinhamento corrente (porção 3') e a seqüência. Considere uma coluna do bloco. A coluna consiste de k caracteres c_i, $1 \leq i \leq k$, $c_i \in$

$\{A, T, C, G, N, -\}$, com qualidade q_i. Se c_i é um gap, o valor q_i é o valor de qualidade da base anterior, se ela existir, caso contrário, será o valor da base posterior. Sete valores de qualidades são computados para cada coluna: 5 para substituições, um para deleção e um para inserção. O valor $q_s(d)$ denota a qualidade média para substituição envolvendo a base d da seqüência, valor q_d denota o qualidade média para deleção e o valor q_n denota a qualidade média para uma inserção.

$$q_s = \left[\left(\sum_{1 \le i \le k \text{ and } c_i = d} q_i \right) - \left(\sum_{1 \le i \le k \text{ and } c_i \ne d} q_i \right) \right] \ / \ k,$$

$$q_d = \left(\sum_{1 \le i \le k \text{ and } c_i \ne -} q_i \right) \ / \ k,$$

$$q_n = \left(\sum_{i = 1}^{k} q_i \right) \ / \ k,$$

$$q_s(N) = -q_n$$

Esta definição de valores médios de qualidade assume que erros em sobreposições de seqüências ocorrem independentemente. Esta hipótese não é verdadeira por que o mesmo erro costuma ocorrer no mesmo contexto local e seqüências sobrepostas contém diversos contextos locais semelhantes. Em uma definição refinada, os valores de qualidade são particionados em grupos por química utilizada no seqüenciamento e orientação, os valores de qualidade de cada grupo são ordenados de maneira decrescente e os valores de qualidade recebem pesos decrescentes. Esta solução não foi implementada nesta versão do software.

Considere uma substituição envolvendo uma coluna de um bloco e a base d de um seqüência com valor de qualidade q_r. Se $q_s(d) > 0$, então a substituição é considerada como um acerto e sua pontuação é $m * min(q_s(d), q_r)$, onde m é um fator positivo para a pontuação de acerto. Se $q_s(d) \le 0$, então a substituição é considerada como um erro e sua pontuação é $n * min(-q_s(d), q_r)$, onde n é um inteiro negativo utilizado como fator de erro.

A pontuação de um gap é a soma da pontuação de abertura do gap mais a pontuação de extensão do gap. A pontuação de abertura, por simplicidade, é um pequeno inteiro negativo. A pontuação de extensão depende dos valores de qualidade. Seja g um inteiro positivo fator de penalização de extensão de gap. A pontuação de extensão de uma coluna com qualidade média de deleção q_d é $-g*min(q_d, q_r)$, onde q_r é o valor de qualidade da base do seqüência imediatamente antes ou depois do gap. A pontuação de extensão de uma

base de um seqüência com valor de qualidade q_r em uma inserção de gap é $-g*min(q_n, q_r)$, onde q_n é a qualidade média de inserção de uma coluna do bloco imediatamente antes ou depois do gap.

Um alinhamento global do bloco com o seqüência com a pontuação máxima é computado em espaço linear usando um técnica de divisão e conquista [48, 52, 72]. Como o alinhamento é executado no máximo uma vez para cada seqüência, é possível realizar a computação utilizando a matriz de programação dinâmica inteira para obtenção de melhores resultados. O valor médio de qualidade para o bloco é pré-calculado de modo que cada entrada na matriz de programação dinâmica seja calculado em tempo constante.

Uma comparação entre o CAP3 e o phrap foi feita e mostrou que o phrap produz contigs mais longos e que o CAP3 produz contigs com taxa menor de erros.

O CAP3 é livre para uso acadêmico e pode ser conseguido através do envio de uma mensagem para o autor (*huang@mtu.edu*).

A.8 A quality control algorithm for DNA sequencing projects [116]

Este trabalho descreve um método estatístico para a detecção de seqüências heterólogas baseado na diferença na composição de hexâmeros existentes nos diferentes organismos, o que pode ajudar na detecção de seqüências contaminadas. O objetivo deste artigo é a descrição e a análise deste método.

O seqüenciamento parcial de seqüências de cDNA selecionadas aleatoriamente para geração de ESTs tornou-se um método popular para a rápida identificação de genes e caracterização de população de transcritos em tecidos. No entanto, estes projetos produzem também seqüências com problemas: clones sem insertos, quimeras, contaminação por clones de fontes heterólogas, etc.

Na busca pela contaminação, utiliza-se a técnica de busca por similaridade. As seqüências que são semelhantes aos contaminantes podem ser desta forma descartadas. Contudo, a maioria dos clones seqüenciados em um projeto EST não são nem mesmos identificáveis por famílias gênicas através dos métodos de similaridade. Da mesma forma, as contaminações, especialmente de origem heteróloga, podem não ser identificadas, o que pode gerar erros substanciais na interpretação dos dados. Por exemplo, no seqüenciamento de tecidos doentes podem existir ESTs provenientes dos organismos que causam a doença.

O método desenvolvido é capaz de reconhecer seqüências heterólogas rapidamente, mesmo que elas não possam ser identificadas por similaridade. Ela é baseada na diferença de composição de hexâmeros no DNA de diferentes organismos. Estas diferenças são pro-

nunciadas em espécies de filos diferentes e se tornam progressivamente menos significantes conforme a maior proximidade das espécies.

A informação contida em um oligômero aumenta conforme o tamanho. Oligômeros de tamanho 6 (hexâmeros) oferecem um bom compromisso entre o desejo de oligômeros informativos e a amostra de incertezas e os impraticáveis tempos de execução quando os oligômeros crescem de tamanho.

O método mais simples de medir a composição de hexâmeros é contar o número de ocorrências dos 4096 (4^6) possíveis hexâmeros. Seqüências de organismos diferentes tipicamente mostram diferentes quantias. Os hexâmeros, assim como nucleotídeos individuais, podem ser vistos como sendo arranjados aleatoriamente em uma seqüência. Assim, uma medida estatística é necessária para quantificar as diferenças. Uma seqüência de tamanho L possui $L-5$ hexâmeros. A priori, a probabilidade de achar qualquer hexâmero em uma seqüência menor que 4 kilobases é $(L-5)/4096$. ESTs tipicamente possuem 300 bases de comprimento, o que resulta em uma probabilidade de 7,2% de qualquer hexâmero ocorrer na seqüência. O teste de composição de hexâmeros usa uma medida de razão de semelhança que fornece um método preciso de comparação de um conjunto de dados contendo raros eventos contra um conjunto maior de dados de controle. Para uma seqüência X e uma conjunto grande A de seqüências do organismo de interesse, a razão de semelhança $\lambda(A, X)$ é a razão entre a probabilidade de se encontrar um particular hexâmero em X e a probabilidade de se encontrar o mesmo hexâmero no conjunto de controle A.

O log da razão de semelhança $D(A, X) = -2 \log \lambda(A, X)$ é a medida de dissimilaridade de X para A. Um valor alto de $D(A, X)$ indica que a composição de hexâmero de X é muito diferente da de A. Se A é um conjunto de controle formado por seqüências do organismo estudado e B é um conjunto de controle de seqüências de um grupo filogeneticamente distante, o valor do $Test(A, B, X) = D(A, X) - D(B, X)$ tenderá a um valor positivo para seqüências de teste menos similares ao organismo estudado do que para o conjunto heterólogo.

Os valores $D(A, X)$ são calculados usando um procedimento simples de contagem. Seja P_i um vetor cujos elementos contém a contagem separada de cada hexâmero encontrado na seqüência. Os elementos deste vetor correspondem aos hexâmeros ordenados alfabeticamente. Assim, a posição P_1 é acontagem para o hexâmero AAAAAA e o elemento P_{4096} é a contagem para o hexâmero TTTTTT. A medida de distância $D(A, X)$ é calculado com a seguinte fórmula:

$$D(A, X) = 2 \times (logL(A, A) - logL(AX, A) + logL(X, X) - logL(AX, X))$$

A expressão AX na fórmula denota a soma dos vetores A e X, ou seja, $AX_i = A_i + X_i$. A função $logL(P, Q)$ é uma razão convencional de log de similaridade, onde P e Q são quaisquer dois conjuntos de seqüências e:

$$logL(P,Q) = \sum_i [(P_i \times \log Q)/\sum_i P_i]$$

A contagem de hexâmeros é obtida para cada seqüência de teste e os valores de $Test(A,B,X)$ são calculado independentemente de outras seqüências de testes. A utilidade deste método como um indicador da composição de hexâmeros é independente do conjunto de testes. Contundo, a certeza com que os dados possam ser interpretados como indicativos da presença de seqüências heterólogas é dependente do tamanho do conjunto de teste.

Como o valor da função $Test(A,B,X)$ é dependente do tamanho do conjunto de testes, neste trabalho, as seqüências com comprimento maior que 400 bases foram trimadas por uma janela aletória de tamanho 300. Resultados obtidos com seqüências menores que 100 bases ou com mais de 2% de ambigüidades de nucleotídeos ou mesmo hexâmeros ambíguos podem ser difíceis de serem interpretados.

O método é independente da orientação das seqüências e sua precisão depende da representatividade das seqüências de controle utilizado e da disponibilidade de um conjunto heterólogo conveniente.

O método foi utilizado na comparação de seqüências provenientes de três organismos: Homem, *Escherichia coli* e *Saccharomyces cerevisiae*. ESTs das bibliotecas humanas de T-limfoblastóides e de tecido cerebral infatil foram comparadas utilizando-se seqüências de controles da *E. coli* e da *S. cerevisiae*. Os resultados mostraram que as seqüências de tecido cerebral possuíam distribuição de hexâmeros semelhante ao de um conjunto de controle de seqüências humanas, e que os ESTs de T-limfoblastóide possuíam uma distribuição similar a distribuição exibida pela levedura. Através desta observação, decidiu-se realizar buscas utilizando o BLAST de todas as seqüências da biblioteca de T-linfoblastóides contra diversos bancos públicos. O critério utilizado para considerar uma seqüência proveniente de um determinado organimo foi a detecção de uma identidade de no mínimo 90%. Os resultados obtidos mostraram que dez clones possuíam origem humana, 8 clones eram aparentemente idênticos aos genes de nucleares humanos e dois eram de origem mitocondrial humana. Para o resto dos resultados, o BLAST apontou que existiam seqüências provenientes dos organismos *Lactococcus latis* e *Saccharomyces cerevisiae*. Além disso, o BLAST apontou diversas seqüências com grande similaridade com seqüências procarióticas e uma seqüência com similaridade a uma proteína de *Drosophila*.

Este resultados mostraram que o método é capaz de identificar seqüências heterólogas provenientes de fontes múltiplas e não esperadas.

A.9 Efficient clustering of large EST data sets on parallel computers [56]

A clusterização de ESTs é uma estratégia poderosa para a identificação de genes. Com o objetivo de tornar esse processo mais rápido, principalmente para o processamento em larga-escala, foi desenvolvido o programa PaCE (Parallel Clustering of ESTs), um software para clusterização em computadores paralelizados. O objetivo deste artigo é descrever as características deste software.

O software se baseia no desenvolvimento de algoritmos eficientes em memória para que ela seja linear conforme o tamanho da entrada, na combinação de técnicas de algoritmos para redução do esforço computacional sem comprometimento da qualidade dos clustering, e no uso de processamento paralelo para redução do tempo de execução e possibilidade de executar o clustering de grandes conjuntos de dados.

Experimentos feitos com os softwares de clusterização disponíveis indicaram que o alinhamento de pares de seqüências utilizando programação dinâmica é a parte intensiva em tempo de execução, enquanto a geração de possíveis pares é a parte intensiva em uso de memória.

Inicialmente cada EST pode ser considerado como um cluster por si mesmo. Dois clusters de ESTs podem ser agrupados se for identificada forte sobreposição entre os ESTs de cada clusters através do alinhamento destas duas seqüências. Se dois ESTs de clusters diferentes não possuem forte sobreposição, os clusters não podem ser agrupados e o esforço computacional é jogado fora. Entretanto, se outro par de ESTs destes clusters apresentar forte sobreposição, os clusters podem ser unidos. Isso mostra que a ordem em que os pares de seqüências são processados não interfere no tamanho final do clusters, mas influencia no tempo gasto pelo processo.

Economia significante de tempo pode ser obtida através da rápida identificação de seqüências que produzirão um resultado positivo quando o alinhamento é executado. A estratégia utilizada pelo PaCE é a geração de pares em ordem decrescente de qualidade de sobreposição. A medida de qualidade da sobreposição é o tamanho do trecho de seqüência maximal comum ao dois ESTs. A razão desta métrica é o fato das seqüências que possuem seqüências comuns maiores tendem a passar no teste do alinhamento.

Para evitar a necessidade de uma grande quantidade de memória para o armazenamento de pares de seqüências promissoras ao alinhamento, um algoritmo sob-demanda foi desenvolvido de modo que ele fosse capaz de lembrar de seu estado e produzir o próximo conjunto de pares assim que fosse requisitado. Este algoritmo utiliza uma árvore de sufixos generalizada (GST - *Generalized Suffix Tree*) [45].

Uma árvore de sufixos (ST - *Suffix Tree*) de uma seqüência s com tamanho m é uma árvore orientada com m folhas numeradas de 1 a m. Qualquer nó interno da árvore possui

no mínimo dois filhos. Cada aresta da árvore é nomeada com uma substring não vazia. Nomes de arestas que saem do mesmo nó devem começar com diferentes caracteres. Para cada folha i da árvore, a concatenação de todas os nomes das arestas da raiz até a folha resulta no sufixo da seqüência s que começa na posição i.

Uma GST é uma árvore de sufixos que combina todos os sufixos de um conjunto de strings. Sejam s_1 e s_2 duas seqüências. Para construção de uma GST para essas duas seqüências, primeiro é feita a construção de uma ST da seqüência s_1. A partir da raiz desta árvore, deve-se comparar s_2 contra um caminho na árvore, até que uma diferença ocorra. Neste ponto deve-se adicionar os caracteres restantes de s_2 na árvore. Quando s_2 for inteiramente processada, a árvore combinará os sufixos das duas seqüências. Se N é a soma dos tamanhos de todas as seqüências, a árvore GST terá no máximo N folhas, e pode ser construída em tempo $O(N)$.

A idéia de se utilizar uma estrutura GST é que se dois ESTs compartilham uma substring maximal comum, então ela será representada como um caminho da raiz até um nó v na árvore GST, com sufixos dos dois ESTs ocupando sub-árvores de v. O número de caracteres existentes no caminho da raiz ao nó v é a profundidade do nó v.

O primeiro passo executado pelo PaCE é a construção da árvore de sufixos em paralelo. A estrutura é construída para as seqüências de entrada e para os seus complementos reversos e usada na geração sob-demanda dos pares promissores de ESTs, que também é feita em paralelo.

O algoritmo de geração de pares promissores sob-demanda é uma variação de um algoritmo para árvores de sufixo para cálculo de todas as repetições maximais de uma seqüência [45]. Um par de seqüências ESTs deve ser escolhido se compartilham uma substring maximal comum com tamanho maior que um valor limite. A árvore é ordenada em ordem decrescente de profundidade dos nós da árvore, e processada nesta ordem para garantir que um par com uma substring maximal comum seja processada antes de uma que tenha tamanho menor. Apesar desta ordenação ser custosa (requer tempo de execução da ordem de $O((N/p)\log(N/p))$, onde p é o número de processadores), o algoritmo tem uma taxa de geração por par $O(1)$ tempo de execução.

A manutenção e atualização dos clusters é feita por um único processador, que atua como processador mestre direcionando os outros processadores a gerarem os pares promissores e a realizarem o alinhamento com os pares promissores selecionados. Para redução do atraso causado pela comunicação, o processador mestre despacha lotes de seqüências para os outros processadores. Para melhorar o balanceamento da carga de processamento, um par promissor gerado por um processador não precisa ser obrigatoriamente alinhado pelo mesmo processador. O processador mestre também é responsável pela recepção do resultado do alinhamento de seqüências e decisão se os clusters devem ser unidos ou não.

Os clusters são mantidos em uma estrutrura union-find [102]. Duas operações são

necessárias, achar os clusters e realizar a união dos clusters. O tempo médio de execução usando este tipo de estrutura é dada pelo inverso da função de Ackermann [102], uma constante para todas aplicações práticas.

O PaCE foi implementado, testado e comparado com outros softwares existentes. O teste de qualidade dos clusterings produzidos foi feito através da comparação com o software CAP3 e mostrou que os resultados são semelhantes mas que os produzidos pelo CAP3 são levemente melhores. Testes de performance foram realizados comparando com os software phrap, TIGR Assembler [101] e CAP3 e mostraram que o PaCE fornece uma ganho substancial em economia de tempo de execução e de memória.

A.10 Complementary DNA Sequencing: Expressed Sequence Tags and Human Genome Project [2]

O objetivo deste artigo é descrever os resultados apresentados por um projeto piloto baseado na técnica EST (*Expressed Sequence Tag*) e a possibilidade de utilização desta técnica no Projeto Genoma Humano.

A utilização do seqüenciamento de cDNA foi muito debatida. Os defensores desta técnica argumentavam que as seqüências codificantes eram a maior parte da informações contidas no genoma e que, no entanto, eram equivalentes a apenas 3% de todo o DNA do organismo. Por isso, o seqüenciamento de cDNA deveria preceder o seqüenciamento genômico. Os defensores do seqüenciamento do genoma completo diziam que a obtenção de todos mRNAs expressos em todos os tecidos, tipos de células e condições seria muito difícil. Além disso, seriam perdidas informações valiosas de regiões de íntrons e intergênicas, incluindo seqüências de controle e regulatórias. Contudo, alguns defensores do seqüenciamento completo erroneamente apontaram que as regiões codificadoras poderiam ser preditas e que, por isso, o seqüenciamento de cDNA não seria necessário. Na verdade, a predição de regiões transcritas é aplicável apenas para éxons relativamente grandes.

Baseada nas discussões acima e na capacidade de rápido seqüenciamento fornecido pelas máquinas seqüenciadoras, um projeto piloto foi iniciado para avaliação do uso de seqüências parciais de cDNA (ESTs).

Sequence-tagged sites (STSs) se tornaram marcas padrões para o mapeamento físico do genoma humando. Os ESTs poderiam servir para o mesmo propósito com a vantagem de apontar a localização de um gene expresso. Além disso, a obtenção de seqüências codificantes tornaria possível a utilização da comparação mais sensível de seqüências peptídicas em adição a comparação de seqüências de nucleotídeos.

Para a avaliação das limitações de um projeto EST de larga-escala, foi preciso analisar a diversidade de representatividade de bibliotecas cDNA, identificar características de-

sejáveis e indesejáveis das bibliotecas e determinar o conteúdo da informação e a precisão de reações de seqüenciamento, realizadas uma única vez, de regiões codificadoras e de outras regiões.

Dados de seqüências provenientes de reações únicas de seqüenciamento foram obtidas de 609 clones cDNA de 3 bibliotecas de tecido cerebral humano. O tamanho médio das seqüências era de 397 bases com um desvio padrão de 99 bases. A hibridização subtrativa foi utilizada com o objetivo de reduzir a população de cDNAs altamente representados.

Os ESTs foram inicialmente examinados por similaridade contra o GenBank. ESTs que não retornassem resultados eram traduzidos em todos os 6 frames e cada frame era comparado com o banco de seqüências protéicas Protein Information Resource (PIR) [9] e com o banco de padrões protéicos ProSite [3].

Com base nas buscas, 8 grupos foram criados. Quatro grupos, com 197 seqüências (32% do total), eram compostos por seqüências humanas: elementos repetitivos, genes mitocondriais, genes ribossomais e outros genes nucleares. Quarenta e oito seqüências (8%) apresentaram alta similaridade com sequüências não humanas, enquanto 230 (38%) não apresentaram similaridade significativa com nenhuma seqüência. As 134 (22%) seqüências restantes não possuíam insertos ou possuíam apenas poly-A.

Sete genes foram representados por mais de um EST. Comparações dos ESTs um contra os outros revelou duas sobreposições de ESTs desconhecidos.

As análises dos resultados produzidos pelo projeto mostraram que a técnica EST é eficiente para obtenção de dados preliminares de clones de cDNA. Os resultados demonstraram também que uma seqüência com tamanho entre 150 e 400 bases possui informação suficiente para identificação do cDNA e de sua localização no cromossomo.

A utilização de seleção randômica de clones se mostrou ineficiente devido à alta representação de determinados clones, o que torna necessária a utilização de técnicas como a hibridização subtrativa.

Outra vantagem da técnica EST é o baixo custo se comparada com outras técnicas.

A.11 The contribution of 700,000 ORF sequence tags to the definition of the human transcriptome [22]

Open reading frame expressed sequence tags (ORF ESTs ou ORESTES) difere do EST convencional porque provê dados da parte central dos transcritos. O objetivo deste artigo é descrever a contribuição desta técnica na definição do transcriptoma humano.

A maioria das seqüências transcritas do ser humano foram obtidas através do seqüenciamento das extremidades de clones de cDNA, conhecidos como ESTs. Estes dados,

contudo, podem ser usado apenas para compilar seqüências curtas e abundantes através da sobreposição das pontas das seqüências obtidas. Em outros casos, é necessário a produção e seqüenciamento um-a-um de clones completos de cDNA.

Uma técnica foi então desenvolvida baseada na modificação da estratégia EST. Esta estratégia nomeada ORESTES ou ORF ESTs consiste na produção de seqüências ao longo do comprimento dos transcritos ao invés de apenas nas extremidades. Isto permite que um seqüenciamento efetivo através de shotgun de transcritos possa ser realizado, acelerando a definição do transcrito e a anotação do genoma.

Com o sucesso de um projeto piloto, um projeto ORESTES de larga-escala foi implementado pela Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) e pelo Instituto Ludwig para Pesquisa do Câncer Humano - Human Cancer Genome Project (LICR-HCGP).

O resultado do trabalho foi um conjunto de dados formado por 696.745 seqüências produzidas a partir de 24 diferentes tecidos (normais e malignos) que produziram 3.540 mini-bibliotecas. Das seqüências obtidas, 7,8% foram identificadas como seqüências mitocondriais, 5,8% demonstraram alta similaridade com seqüência bacterial e 6,1% eram de seqüências repetitivas. Das 559.675 seqüências restantes, 62% possuíam alta similaridade com seqüências transcritas humanas e 38% não tinham nenhuma semelhança com os transcritos humanos disponíveis publicamente.

Para estimar a taxa de descoberta de genes desta técnica foi calculada a porcentagem dos mRNAs completos cobertos por pelo menos um ORESTES. Cinqüenta porcento do mRNAs completos abundantes foram cobertos após apenas 30.000 ORESTES terem sido seqüenciados. A mesma porcentagem foi obtidas para os mRNAs com quantidade moderada após aproximadamente 50.000 ORESTES. Foram necessários entre 300.000 e 450.000 seqüências para os mRNAs pouco abundantes e raros. Após 700.000 seqüências, 94% dos abundantes e moderadamente abundantes foram cobertos. Estima-se que o conjunto de dados produzida cobre em torno de 60% dos genes humanos.

Baseada na cobertura observada e na característica da distribuição das seqüências ORESTES, uma estratégia eficiente para determinação das seqüências completas dos transcritos foi sugerida. As seqüências completas de mRNA disponíveis em bancos públicos foram determinadas através da produção e seqüenciamento de cDNAs completos. Uma estratégia alternativa seria a produção de ORESTES e ESTs 3' e 5' que permitiriam a geração de contigs, que eventualmente cobririam todos transcritos humanos. Neste contexto, estima-se que 800.000 ORESTES de um único tecido permitiriam que a maior parte dos transcritos expressos nele pudessem ter suas seqüências completas determinadas.

Para a união dos contigs ser efetiva, a produção de ORESTES deve abranger trechos da seqüências de modo que a distância entre dois ORESTES seja suficientemente próxima para que eles possam ser unidos por reações RT-PCR. As extremidades seriam determi-

nadas pelos ESTs 3' e 5'. Os ORESTES amplificados cobririam em torno de 80% do comprimento da seqüência e os ESTs cobririam os 20% restantes.

Em comparação com outras técnicas ESTs, ORESTES mostra vantagens e desvantagens. A técnica ORESTES é baseada em low-stringency RT-PCR e necessita de RNA de uma qualidade muito alta, pois qualquer DNA contaminante será amplificado. Existe ainda a presença de artefatos PCR na seqüência, mas isto não impede a eventual construção de transcritos de alta qualidade a partir de dados genômicos. Isso significa, no entanto, que cuidado deve ser tomado na utilização de ORESTES para determinação de polimorfismos. ORESTES tende a produzir seqüências originárias na porção central do trascrito, o que faz com que exista uma complementariedade com outras técnicas de seqüenciamento. Além disso, a técnica é capaz de produzir dados a partir de uma pequena quantia de material inicial.

A.12 ESTWeb: bioinformatics services for EST sequencing projects [75]

O objetivo deste artigo é descrever algumas funcionalidades providas pelo sistema de gerenciamento de projetos ESTs denominado ESTWeb. Este sistema realiza a receptação dos cromatogramas, processamento das seqüências (base-calling, trimagem e comparação com seqüências de bancos públicos), armazenamento dos dados e geração de relatórios.

O processamento das seqüências é iniciado com a determinação das bases da seqüência com utilização do programa phred. Em seguida, os vetores e adaptadores são identificados com utilização do cross_match. Os segmentos que não foram considerados vetores ou adaptadores que possuírem mais de 75 bases com qualidade acima de um valor mínimo dentro de uma janela de 100 bases serão considerados candidatos a insertos de cDNA. As pontas de baixa qualidade são trimadas com a utilização de uma janela deslizante de 100 bases, com um algoritmo que se ajusta conforme a presença de primers dentro da seqüência. O processamento permite a identificação de insertos quiméricos. As seqüências são comparadas contra bancos públicos com a utilização do BLASTN.

O ESTWeb realiza o armazenamento de todos os dados produzidos pelo processamento das placas. Ele armazena também todas as informações necessárias para a gerência do projeto, como laboratórios, vetores e adaptadores utilizados.

Esta versão do sistema não lida com a clusterização de seqüências. Ela roda em máquinas UNIX em um ambiente Apache-CGI-Perl e utiliza o gerenciador de banco de dados Postgres. O pacote é livre e possui licensa GNU GPL.

A.13 Bioinformatics of the Sugarcane EST Project [103]

O objetivo deste artigo é descrever o papel desempenhado e os procedimentos adotados pelo Laboratório de Bioinformática (LBI), localizado no Instituto de Computação (IC) da Universidade Estadual de Campinas (Unicamp), no consórcio que desenvolveu o projeto EST de seqüenciamento da cana-de-açúcar (SUCEST - Sugarcane EST Project).

Em geral, os projetos ESTs são desenvolvido por um laboratório único, no entanto, o projeto SUCEST foi formado por um consórcio que envolveu 76 laboratórios de seqüenciamento e de data mining. O LBI foi o responsável pela criação de um web site que provia serviços de recepção, processamento, análise e exploração de dados.

O LBI mantinha uma base de dados relacional para armazenamento de toda informação referente ao projeto: laboratórios participantes, bibliotecas, placas, seqüências, clusterings, etc.

O processo de obtenção de dados tinha início com a submissão de placas feita pelos laboratórios que realizavam o seqüenciamento. As placas eram recebidas pelo LBI, que executava um procedimento que se iniciava com a recepção e a verificação da integridade do arquivo zip com os cromatogramas das placas. O passo seguinte era a determinação das bases e das qualidades das seqüências, a partir dos cromatogramas, através da execução dos programas phred e phd2fasta.

Os clusterings de ESTs eram importantes meios de obtenção de conjuntos mais informativo de seqüências para estudo e estimativas das redundâncias das bibliotecas produzidas. Neste projeto, determinou-se que cada cluster deveria refletir um transcrito, ao invés de um gene, um alelo, ou outra entidade biológica. Determinou-se também que um cluster não consistiria apenas do conjunto de seqüências, mas também do alinhamento delas e de um consenso.

Com o objetivo de minimizar os artefatos, as seqüências eram trimadas antes da clusterização. Primeiro, utilizava-se o programa cross_match para identificação das seqüências de vetores. Então, os trechos de poli-A, vetor e adaptador eram removidos. Um procedimento de trimagem, baseado na qualidade das bases, era aplicado, removendo base a base das pontas da seqüência até que pelo menos 12 bases com qualidade superior a 15 fossem encontradas em uma janela de tamanho 20. As seqüências também eram verificadas em busca de contaminação por *Xylella fastidiosa*, *Xanthomonas citri*, *Escherichia cole* e outros potenciais contaminantes. As seqüências eram consideradas contaminadas se possuíssem pelo menos 100 bases com similaridade maior que 90% com o contaminante. As seqüências contaminadas não eram removidas do processo de clusterização, ficando a cargo dos pesquisadores decidirem como tratar esta informação.

As seqüências eram agrupadas pelo phrap utilizando-se os dados de qualidade das

seqüências e parâmetros mais estritos que os padrões utilizados pelo próprio phrap: `-penalty -15 -bandwidth 14 -minscore 100 -shatter_greedy`.

A atualização do clustering era feita diariamente, mas com o crescimento do conjunto de seqüências, a atualização passou a ser realizada uma vez por semana. Na fase final do projeto, o processo de clusterização consumia um processador inteiro por 20 horas do servidor do projeto: uma Compaq AlphaServer ES40 com dois processadores Alpha 667 MHz, 8 GB de RAM, e 384 Gb de disco, rodando o sistema opercional OSF-1.

A última execução do processo de clusterização inclui 261.609 seqüências trimadas e produziu 81.223 clusters. No entanto, análises feitas por pesquisadores participantes do projeto mostraram que o procedimento de clustering não era satisfatório devido à presença de muitos clusters mal formados ou que poderiam ser combinados em um só. Isso levou ao desenvolvimento de um novo processo de clustering, descrito detalhadamente por Telles e da Silva [104].

O novo esquema baseou-se em um procedimento mais elaborado de trimagem, e utilizou o programa CAP3 para a geração dos clusterings. Assim, 237.954 seqüências foram processadas pelo CAP3 que produziu 43.141 clusters.

O LBI também fornecia serviços para análise dos dados produzidos. Os clusters eram blastados contra os bancos nr, nt e dbEST para identificação de seqüências de transcritos completas ou seqüências novas. Buscas por palavra chaves encontradas no resultado dos BLASTs também podiam ser realizadas.

Os usuários também podiam realizar buscas por clusters com determinadas características, assim como registrar uma anotação manual nos clusters. Um procedimento de categorização de clusters automático também foi desenvolvido. As seqüências consensos dos clusters eram blastadas contra um banco de seqüências protéicas que representavam categorias. Se alguma seqüência cobrisse 70% ou mais da seqüência protéica e apresentasse um e-value melhor ou igual a 10^{-10}, ela seria considerada como participante da categoria.

Finalmente, o LBI realizava a comparação entre as seqüências obtidas da cana-de-acúcar contra os genomas de outros organismos, como por exemplo, *Arabidopsis thaliana*, *Lycopersicon esculentum*, *Glycine max*, etc, e disponibilizava os dados para consulta.

A.14 Automated Sequence Preprocessing in a Large-Scale Sequencing Environment [115]

O objetivo deste trabalho é descrever um sistema, criado para pré-processamento e montagem de seqüências produzidas por experimentos de seqüenciamento, chamado GASP.

O aumento da automação de processos em projetos genomas é um fator importante

para que os objetivos sejam atingidos e que os custos sejam diminuídos. Uma área em que isto é especialmente verdade é o processamento de dados provenientes de experimentos shotgun, em que é preciso extrair informação das imagens fluorescentes do gel de eletrosforese e transformá-la em seqüências montadas.

O pré-processamento de seqüências, também chamado, pré-montagem, é a transformação de dados extraídos das máquinas de seqüenciamento em informações prontas para a montagem das seqüências. Isto envolve tarefas como a conversão dos dados obtidos das máquinas de seqüenciamento em dados de seqüências e qualidades, identificação de trechos de vetor e avaliação da qualidade. Isto representa um esforço computacional significante, principalmente quando existem diversos clones sendo seqüenciados simultaneamente. A análise dos resultados pelo pré-processamento é importante para avaliação de alterações incrementais feitas nos protocolos de seqüenciamentos dos projetos.

Por causa da sua importância, o problema de pré-processamento de seqüências gerou uma coleção de softwares baseados em UNIX com diferentes paradigmas. Alguns são baseados em Bourne shell scripts, outros em Perl scripts, etc. Esta diversidade reflete as diferentes necessidades que os laboratórios e centros de seqüenciamento possuem.

O Centro de Seqüenciamento de Genoma (GSC - *Genome Sequencing Center*) da Universidade de Washington e o Sanger Center desenvolveram um software baseado em Perl chamado Genome Automated Sequence Preprocessor (GASP). Este software foi projetado para atender a demanda de pré-processamento de 100.000 seqüências produzidas semanalmente pelos dois centros.

O software opera de modo a realizar o pré-processamento das seqüências e a disponibilizar diversos tipos de relatórios para análise dos resultados obtidos. Os relatórios podem, por exemplo, sumarizar informações sobre os tamanhos das seqüências obtidas após o pré-processamento, a qualidade das seqüências, a porcentagem de vetor, etc, em forma de tabelas ou gráficos.

O primeiro passo do pré-processamento é a utilização do software phred para conversão dos dados provenientes das máquinas de seqüenciamento ABI em arquivos no formato padrão SCF. Em seguida, é verificada a nomenclatura da placa e dos reads para determinação de uma série de informações como, por exemplo, direção de seqüenciamento, química utilizada na reação, etc. O terceiro passo é a determinação das bases e das qualidades com utilização do programa phred. A quarta etapa envolve o mascaramento das seqüências de vetores de seqüenciamento e de clonagem e a execução de um filtro de qualidade.

Após estas estapas as seqüências são marcadas como aprovadas ou reprovadas. As seqüências aprovadas são encaminhadas para o processo de montagem de fragmentos incremental. O software phrap é utilizado como ferramenta de montagem primária, e um módulo do software que realiza interface com o programa gap [17], que realiza montagem

de seqüências, é usado para adicionar incrementalmente as seqüências processadas em uma montagem já existente.

Uma série de dados estatísticos são derivados dos dados produzidos pelas etapas acima para geração dos relatórios.

O software GASP é livre para uso acadêmico. Outras formas de uso requerem permissão dos autores.

A.15 New ways for automatic detection of contaminants in EST projects [79]

O objetivo deste trabalho é descrever novas metodologias para a detecção de contaminação em seqüências produzidas em projetos ESTs. São descritos métodos que não se baseiam na tradicional busca por similaridade entre as seqüências dos projetos e as de potenciais contaminantes.

Em projetos genoma baseados em ESTs, genes expressos dos organismos são amostrados e fragmentos de seus transcritos reversos são obtidos. Para projetos que envolvem o estudo de grandes genomas ($>$ 20 Mbp) esta técnica produz importante informação a um custo relativamente baixo se comparado a outras técnicas. A fase de busca de contaminação destes projetos é usualmente baseada em similaridade e, tipicamente, utiliza-se o programa BLAST e critérios como, por exemplo, "excluir seqüências que apresentarem pelo menos 98% de similaridade com o potencial contaminante em uma janela de 75 bases e apresentarem e-value $\leq 10^{-15}$". Naturalmente poderíamos perguntar: "Porque 98% e não 97%? Porque 10^{-15} e não 10^{-14}?"

Baseado nisso, uma metodologia mais racional foi proposta. A idéia básica era capturar características intrínsecas do transcriptoma alvo e prover alguma medida de confiança para o resultado obtido. Similaridade não foi usada neste trabalho, mas existe a possibilidade de que, futuramente, os métodos possam combinar a utilização desta técnica.

Uma característica desejável de um processo de detecção de contaminação é a habilidade de detectar qualquer seqüência estranha ao projeto, e não somente as pertencentes aos potenciais contaminantes. No entanto, neste trabalho foi assumido apenas que um conjunto de potenciais contaminantes era conhecido, para que a taxa de falsos positivos e falsos negativos pudesse ser estimada.

A metodologia desenvolvida procurava determinar as características intrínsecas das seqüências alvos do transcriptoma através da execução de diferentes programas de análise de seqüências, alguns deles de terceiros e desenvolvidos com outros objetivos, diferentes da busca por contaminação. As maneiras como esses dados eram combinados foram chamadas de *métodos*.

Todos métodos necessitavam de um conjunto de treino e produziam uma pontuação que indicava a probabilidade de uma seqüência pertencer a um transcriptoma alvo. Os métodos utilizados foram:

- ESTScan (ES) [55]: um programa desenvolvido para encontrar genes com um corretor de frames de leitura, desenvolvido especialmente para ESTs.
- Glimmer (GL) [32]: um programa identificador de genes utilizado em genomas de procariotos.
- Distribuição de dinucleotídeo (DN) e trinucleotídeo (TN): utilizava uma formúla de distribuição multinomial [68] que estima a probabilidade de um dado set de k-mêros pertencer a um organismo ($k = 2$ e 3).
- Assinatura de dinucleotídeo (DS) [57].
- %GC (BN): Utilizava uma fórmula de distribuição binomial [35] para estimar a probabilidade de um dado valor de %GC ser o mesmo do organismo alvo.

Os resultados dos métodos seriam obtidos e combinados. Esta metodologia proposta não possuía semelhança com nenhuma outra conhecida. Apenas o trabalho de White *et al.* [116] utilizava um único método semelhante ao DN e TN, utilizando $k = 6$. (Este valor foi testado, mas não produziu bons resultados neste trabalho). O trabalho de Hraber e Weller [50], baseado no de White *et al.* [116], também utilizava um único método.

Dado um conjunto de métodos $M = M_1, M_2, ..., M_k$, um conjunto de organismos $C = C_1, C_2, ..., C_l$ (assumiu-se que os genomas completos destes organismos eram conhecidos) e um conjunto de seqüências do organismo alvo $L = L_1, L_2, ..., L_m$, foram definidos três conjuntos de seqüências para cada potencial contaminante C_j:

- L^{train}: conjunto de treino (um subconjunto de L).
- *check*: contém uma amostra de seqüências de C_j e uma amostra de seqüências de L. Utilizado para estimativa de falsos positivos (FP) e negativos (FN).
- *project*: contém seqüências de C_j e L. Este conjunto contém de 4 a 40 vezes mais seqüências que o conjunto *check*. Um método é avaliado baseado nos resultados produzidos para este conjunto.

Os conjuntos foram criados para que a metodologia fosse testada em uma simulação. Isso significa que a origem precisa de cada seqüência utilizada era conhecida. Para cada par (C_j,L) cada método M_i era executado da seguinte maneira:

1. M_i era treinado em L^{train}.

2. M_i era executado com o conjunto *check* (fase de verificação).

3. FP e FN eram avaliados com base nos resultados do passo anterior.

4. M_i era executado com o conjunto *project* (fase de teste).

O passo 3 era executado da seguinte forma: o conjunto de pontuações $S = s_1, s_2, ..., s_n$, onde s_i é a pontuação da seqüência i, produzido no passo anterior era ordenado em ordem crescente de pontuação. Uma partição de pontuação σ era determinada de modo a minimizar FP+FN. Com isso, era possível determinar um conjunto B baseado em S, onde $B_i = 0$ (um resultado negativo, contaminante) se $s_i < \sigma$, ou $B_i = 1$ (positivo, seqüência legítima) se $s_i \geq \sigma$.

Desta maneira, cada seqüência i de *check* poderia cair em uma das 4 seguintes situações:

1. Se $B_i = 1$ e i é legítima, i é um positivo verdadeiro (TP).

2. Se $B_i = 1$ e i é contaminante, i é um falso negativo (FN).

3. Se $B_i = 0$ e i é legítima, i é um falso positivo (FP).

4. Se $B_i = 0$ e i é contaminante, i é um negativo verdadeiro (TN).

Neste trabalho, este conjunto de valores foi utilizado para comparar os valores de FP e FN previstos com o conjunto *check* com os valores obtidos na fase de teste. Em condições reais, os valores previstos poderiam ser utilizados como estimativas.

A combinação dos métodos era feita baseada na idéia que os resultados de dois métodos combinados poderiam proporcionar resultados melhores do que se eles fossem aplicados individualmente. Das muitas maneiras de se combinar os dados, duas foram estudadas:

- *Votação*: A estratégia de combinação mais simples. Uma seqüência era considerada legítima (ou contaminante) se uma maior porcentagem x de métodos diziam que ela era. O valor x era escolhido de modo que minimizasse FP+FN contra o conjunto *check*. Isto não garantia que os resultados combinados seriam melhores que um método único, pois os diferentes métodos tinham poder diferente de classificação para uma particular combinação (C_j, L). Na simulação feita, todos os métodos tinham o mesmo peso, mas a literatura sugere que melhores resultados poderiam ser obtidos com pesos diferentes ou com uma estratégia de votação mais complexa.

- *Vetores Binários*: Esta estratégia baseia-se na junção de predições bem-sucedidas, dando pesos maiores a elas do que as predições que obtiveram menos sucesso. O resultado de cada método era visto individualmente como um valor binário (0 = contaminante, 1 = legítima), e cada seqüência de teste possuía associado um vetor binário com a junção dos resultados. Os vetores eram comparados com os vetores das seqüências do conjunto *check*, e a fração de respostas certas e erradas era verificada. O vetor mostraria se uma seqüência era contaminante (fração de TNs maior que a de FNs) ou legítima (analogamente). Quando as frações eram iguais, a seqüência era considerada legítima, já que em situações reais as legítimas são mais numerosas que as contaminadas. O problema desta estratégia é quando vetores não representados pelo conjunto *check* apareciam no conjunto *project*. Como não era clara a decisão a ser tomada as seqüências não eram categorizadas. Contudo, o número de vetores cresce exponencialmente com o número de métodos.

As simulações foram feitas com seqüências obtidas no NCBI. Apenas as seqüências codificantes com tamanho entre 200 e 2000 bp foram consideradas. As seqüências que possuíam letras diferentes de A,C,T ou G foram descartadas. O organismo alvo era a *Drosophila melanogaster* (DM). Os organismos contaminates eram:

- *Clostridium perfrigens* (CP): bactéria, filogeneticamente distante do organismo alvo e com conteúdo GC muito diferente.
- *Escherichia coli* (EC): bactéria, com conteúdo GC próximo ao encontrado na *Drosophila melanogaster*.
- *Saccharomyces cerevisae* (SC): fungo, mais próximo do organismo alvo do que qualquer outra bactéria.
- *Caenorhabditis elegans* (CE): nematelminto, mais próximo filogeneticamente do que os outros contaminantes escolhidos.

Para cada contaminante, dois conjuntos *check* foram criados, um com 20 vezes mais seqüências que o outro para verificar se o número de seqüências importa na estimativa de FP e FN.

Os resultados de todos os métodos e das duas combinações foram obtidos e comparados. A análise mostrou que os vetores binários apresentaram melhor performance. O método de votação e o BN apresentaram os segundos melhores resultados, em números iguais de conjuntos, mas a votação apresentou melhores resultados na média. O teste que envolveu DM+CE apresentou maiores dificuldades devido à proximidade filogenética. Entre as bactérias, o caso que apresentou maiores dificuldades para realizar a separação foi o DM+EC que, inclusive, apresentou dificuldades maiores do que o caso DM+SC.

O método de vetores binários apresentou maior diferença entre as estimativas e os reais valores de FP+FN, provavelmente por causa dos vetores que não foram amostrados. Apesar do aumento do tamanho dos conjuntos *check* melhorarem a estimativa FP+FN em 15 casos, em 17 casos houve piora, o que mostra que nem sempre um grande número de seqüências implica em melhora nas estimativas de FP+FN.

Os resultados mostraram que a metodologia pode ser promissora, mas também mostraram que a detecção de contaminação através de características intrínsecas está longe de ser fácil. As simulações não levaram em conta casos que acontencem em situações reais como, por exemplo, contaminação genômica, que podem tornar o reconhecimento das seqüências do organismo alvo muito mais difícil.

A.16 EST contaminant detection by combination of multiple classifiers [80]

Este artigo tem o objetivo de apresentar os novos resultados do trabalho inicialmente apresentado em [79].

Como já mencionado no artigo anterior, a metodologia se baseia no uso de extratores de características de seqüências.

À lista de programas mencionadas anteriomente, foi adicionado o HBQCM (HN) [116]. Este programa utiliza uma fórmula de taxa de semelhança para estimar a probabilidade de um dado conjunto de hexâmeros pertencer ao organismo utilizado no processo de treinamento.

O procedimento de obtenção das informações das características sofreu algumas alterações. Foram criados dois conjuntos adicionais: o conjunto *eval* que contém todos os conjuntos *check*, e o conjunto *test* que contém todos os conjuntos *project*.

Cada método M_i era executado para cada organismo. Primeiro o método era executado com conjunto de treino, em seguida, ele era executado com o conjunto *eval* e, finalmente, com o conjunto *test*. As pontuações obtidas dos conjuntos *eval* e *test* eram linearmente normalizadas, com valores na faixa de 0 (baixa confiança) a 100 (alta confiança).

Para a combinação dos resultados dos métodos foi escolhida a estratégia da votação. A escolha se deve ao largo uso da votação, a sua simplicidade e aos bons resultados que ela apresenta. Quatro estratégias de votação foram determinadas. Toda estratégia de votação deveria levar em conta os valores de saídas dos programas, agrupados por organismos, e atribuir uma seqüência para um e somente um destes organismos. As estratégias usadas foram [110]:

- regra da soma: os valores de características obtidos de cada programa são somados.

- regra do produto: os valores de características obtidos de cada programa são multiplicados.

- pluralidade: para cada programa os valores de características obtidos são classificados de modo que o melhor valor fique em primeiro lugar. Assim, cada programa identificará mais fortemente um organismo. O organismo apontado pela maior parte dos programas é escolhido.

- *borda count*: para cada programa os valores de características obtidos são classificados de modo que o melhor valor fique em primeiro lugar. Se para um programa P_i, um organismo ficou classificado em último, ele recebe 1 ponto. Se ficou classificado em penúltimo, ele recebe 2 pontos, e assim, por diante. Cada organismo têm seus pontos somados e aquele que tiver a maior soma é escolhido.

Para evitar o problema em que um programa pode apresentar maus resultados para um certo conjunto de entradas, todas combinações de programas foram testadas e apenas aquelas que diminuíam a taxa geral de erros foram mantidas.

Usando as estratégias de votação, cada seqüência do conjunto *eval* foi atribuída a um organismo, e para cada organismo os valores TP, FN, FP e TN foram estimados. Estes valores foram usados como estimativas de FP e FN das seqüências do cojunto *test*. A melhor estratégia adotada, deveria apresentar a menor porcentagem FP+FN obtida para o conjunto *eval*, assim como a menor diferença entre FP+FN dos conjuntos *eval* e *test*. Uma vez que a melhor combinação de programas fosse determinada para o conjunto *eval*, ela seria aplicada ao conjunto *test*.

Para avaliar a metodologia foram criado quatro grupos com combinação de seqüências de diferentes organismos. O primeiro grupo possuía seqüências de DM e CE, o segundo de DM, EC e CE, o terceiro de DM, CP e SC, e o quarto de todos os organismos.

Após a execução da metodologia para todos os grupos, foi possível observar que o programa ESTScan é o melhor para extrair características das seqüências.

Em contraste com as metodologias baseadas em similaridade, os valores FP+FN podem ser utilizados em conjunção com as pontuações para refinar os resultados na busca por adicionais contaminantes ou seqüências legítimas que foram perdidas.

Para o grupo 1, pluralidade e borda count apresentaram melhores resultados tanto na fase de verificação como na fase de teste. Para o grupo 2, a pluralidade apresentou melhor resultado na fase de verificação mas, na fase de teste, o melhor resultado foi da regra do produto. A regra do produto apresentou melhor desempenho na verificação do grupo 3, mas pluralidade e borda count apresentaram menos erros na fase de teste. Finalmente, no grupo 4, a regra da soma apresentou melhor resultado na verificação e a regra do produto melhor desempenho na fase de teste.

Para avaliar como a metodologia se comportaria no caso de contaminação bacterial não prevista, um conjunto *test* formado por 200 seqüências de *Xanthomonas campestris* foi construído e avaliado com o conjunto *eval* do grupo 4. O valor FP+FN do conjunto *test* do grupo 4 para a regra da soma aumentou de 8,3% para 14,4%, o que mostra que a metodologia é altamente dependente de treino.

A.17 A novel algorithm for computational identification of contaminated EST libraries [98]

Um objetivo chave do projeto Genoma Humano é o entendimento do conjunto completo de proteínas humanas, o proteoma. Como a seqüência genômica completa não é suficiente para a predição de genes e de eventos de tradução alternativos, a técnica EST é usada para estas finalidades. No entanto, a grande quantidade de artefatos presentes nas seqüências disponíveis no dbEST freqüentemente causa predições inválidas. O objetivo deste trabalho é a descrição de um novo método para reconhecimento de contaminação por DNA genômico e por outros artefatos, que usualmente não são removidos pelos processos tradicionais de limpeza.

Uma coleção típica de ESTs é altamente redundante. Por isso, um importante processo para análise da estrutura gênica é o agrupamento dos ESTs, supostamente do mesmo gene, em clusters, baseada na sobreposição de seqüências. Alguns sistemas realizam a montagem de cada cluster para produzir um alinhamento múltiplo que aproxima a seqüência consenso do cDNA original. DNA genômico, quando disponível, pode ser utilizado para guiar a clusterização e identificar os limites dos éxons.

O maior obstáculo para a correta identificação de genes é a alta taxa de erros nos bancos de ESTs. As seqüências não são editadas, são lidas em uma única passagem, possuem comprimento de algumas centenas de bases e têm uma taxa de erro na determinação das bases tão alto quanto 3%.

Além dos erros de seqüenciamento, as seqüências sofrem vários tipos diferentes de contaminação, dependendo, em parte, de qual dos muitos protocolos foi utilizado na construção das bibliotecas de cDNA.

Um problema típico é a inclusão de sequências de vetores, utilizados no seqüenciamento, ou de sítios de restrição no final das seqüências. A contaminação por vetor também pode ocorrer devido aos eventos de rearranjo de DNA, dentro da bactéria hospedeira, que causam a inserção de seqüência bacterial no meio do EST. Em geral, este tipo de contaminação é detectado através da comparação das seqüências obtidas com a seqüência genômica do vetor.

As seqüências podem ser contaminadas por outros organimos, como os vírus. Estes

contaminantes podem ocorrer devido à infecção do tecido que originou a biblioteca ou à contaminação do laboratório. As seqüências contaminadas podem ser identificadas de forma semelhante a da contaminação por vetor.

Seqüências quiméricas são outro problema. Uma quimera é a concatenação de duas ou mais seqüências expressas de diferentes áreas. Se ela é utilizada na clusterização, ela pode realizar a combinação de dois genes em uma única predição incorreta de gene. A identificação de quimeras é mais difícil que a identificação de outros tipos de contaminação, pois a seqüência inteira tem origem no organismo correto. Como os ESTs são unidos aleatoriamente, eles são geralmente de diferentes cromossomos ou de regiões distantes do mesmo cromossomo. A comparação com o genoma completo do organismo pode auxiliar a identificação de ESTs quiméricos. Algoritmos sofisticados que modelam propriedades dos ESTs também são utilizados para encontrar quimeras.

Podem ocorrer também contaminações causadas por DNA genômico do próprio organismo. Este tipo de contaminação pode fazer com que íntrons apareçam como regiões expressas, levando à predição de traduções alternativas inexistentes. Contaminação por DNA intergênico pode resultar na predição de falsos genes. Os atuais métodos de remoção de contaminação são incapazes de identificar estes casos e técnicas que eliminam todas as seqüências não traduzidas, que criam novas supostas traduções alternativas, podem resultar na perda de verdadeiras traduções alternativas. Em particular, uma seqüência proveniente de um gene de éxon único pode ser perdida com este tipo de filtragem.

Outra forma comum de contaminação em ESTs é a contaminação por mRNA prematuro. Apesar de ESTs representando contaminação por pré-mRNA possam parecer retenção de íntrons, eles são muito mais artefatos do que seqüências exônicas reais.

Os processos de limpeza de ESTs normalmente processam um EST de cada vez. Isto é apropriado para detecção de erros de seqüenciamento, vetores, vírus e quimeras, mas não servem para o tratamento de contaminação por DNA genômico e por pré-mRNA. Este trabalho propôs uma nova metodologia que seria capaz de tratar todos esses casos.

Como a contaminação por DNA genômico e por pré-mRNA freqüentemente dependem do protocolo de criação das bibliotecas, a biblioteca inteira é afetada. Dessa forma, a metodologia foi desenvolvida de forma a analisar a biblioteca inteira através da utilização da informação de clustering e montagem.

A análise feita baseou-se no exame da estrutura gênica e nas variantes de traduções, preditas através do alinhamento de 3,86 milhões de ESTs humanos disponíveis no dbEST.

Os ESTs foram clusterizados e montados com a utilização do sistema LEADS [91]. O processo de clusterização considerou a sobreposição entre as seqüências para determinar os clusters e o processo de montagem considerou os padrões de sobreposição para prever as estruturas dos genes e as variantes de traduções de cada cluster.

Antes da clusterização, o sistema LEADS realiza a limpeza das seqüências através do

alinhamento dos ESTs com as seqüências de vetores, adaptadores e de outros organismos. Seqüências de imunoglobulina e de receptores de células T foram removidas devido aos seus complicados padrões de rearranjo, que dificultam a clusterização e a montagem.

Regiões repetitivas e de baixa complexidade foram removidas com a utilização de um modelo de alinhamento heurístico. Sementes de repetições eram filtradas e extendidas com a utilização de um alinhamento Smith-Waterman [94] com parâmetros: acerto = 1, erro = -3, abertura de gap = -5, extensão de gap = -5 e pontuação mínima = 22.

Após esta etapa inicial, sobraram 3,53 milhões de ESTs, que foram alinhados com o genoma completo, utilizando um modelo de tradução que permitia buracos longos. Alinhamento com identidade mínima de 94% eram necessários para que um EST fosse selecionado. Se a seqüência possuísse trechos que alinhassem com múltiplos cromossomos, ou se ela alinhasse com um único cromossomo e incluísse regiões intrônicas, ela era considerada uma quimera e então descartada. Regiões de baixa qualidades das extremidades dos ESTs, baseada na comparação com o DNA genômico, eram trimadas.

Os passos anteriores deixaram 3,05 milhões de seqüências para serem clusterizadas e montadas. A montagem de 14 clusters falhou devido a problemas nos dados e 110 clusters contendo mais de 1000 seqüências falharam por causa da limitação do processamento computacional. O conjunto final de análise ficou, portanto, com 2.72 milhões de ESTs de 6.649 bibliotecas.

As análises se basearam nos clusters. Utilizando as estruturas gênicas descritas pela montagem, calculou-se para cada biblioteca o percentual de seqüências únicas não traduzidas, de seqüências que sobrepuseram íntrons e de seqüências que continham íntrons não canônicos. Para cada característica foram calculados o desvio padrão e a média. As bibliotecas que possuíam percentuais maiores que a média mais três desvios padrões eram consideradas contaminadas com DNA genômico, contaminadas com pré-mRNA e com prevalecência de íntrons não canônicos, respectivamente. Para que as estatísticas fossem mais significantes, foram consideradas apenas bibliotecas com mais de 100 ESTs que apareciam em pelo menos 50 clusters. As 1906 bibliotecas que atenderam esse critério reuniam 2,52 milhões de seqüências.

Devido à característica aleatória da contaminação por DNA genômico, os ESTs contaminados tendem a ficar em clusters de tamanho 1, e por isso, as bibliotecas contaminadas tendem a ter uma abundância de clusters deste tipo. No entanto, estas seqüências também podem ser encontradas em clusters de tamanhos maiores. Utilizando o critério dos 3 desviões padrões, 21 bibliotecas foram consideradas contaminadas. As seqüências que foram encontradas em clusters de tamanho 1 destas bibliotecas foram removidas, assim como as seqüências não traduzidas encontradas em clusters maiores. Foram removidas 11.667 ESTs, o que eliminou a predição de 1.175 transcritos espúrios. Para verificar a validade da remoção, comparou-se o conteúdo de seqüências repetitivas das seqüências.

A concentração de seqüências repetitivas é maior em regiões não expressas. Das 11.667 seqüências removidas, 30,2% apresentou concentração alta, enquanto em um conjunto de 10.000 ESTs escolhidos aleatoriamente, apenas 8,1% apresentaram alta concentração, o que indica que as seqüências podiam ser contaminações.

Ao contrário da contaminação por DNA genômico, a contaminação por pré-mRNA gera seqüências mais próximas de genes. Para a identificação destas seqüências, primeiro foi feita a identificação dos íntrons existentes no genoma, sendo que um íntron foi definido como um buraco de pelo menos 15 bases no alinhamento de uma seqüência expressa com o genoma, que começava com as bases "GC" ou "GT" e terminava com "AG". O passo seguinte era a procura por seqüências supostamente não traduzidas, que sobrepunham (mas que não eram inteiramente contidas) os íntrons e a avaliação das porcentagens destes casos em cada biblioteca. Após a análise, 14 bibliotecas foram consideradas contaminadas. Todas as seqüências destas bibliotecas que possuíam sobreposição com um íntron foram removidas. Isto resultou num total de 2.128 ESTs, o que ocasionou a eliminação de 538 predições espúrias. A mesma verificação aplicada na contaminação genômica foi feita e mostrou que 16,1% das seqüências possuíam alta concentração de seqüências repetitivas.

Íntrons que começam com GT e terminam com AG ocorrem em 98,12% dos casos, e os íntrons do tipo GC/AG correspondem à 0,76% dos casos. No entanto, existem os íntrons não canônicos, que podem contaminar as seqüências. Para avaliar este tipo de contaminação, primeiro foi feita uma avaliação dos íntrons obtidos no processo descrito acima, e um íntron não canônico foi identificado. Ao analisar todas as bibliotecas, verificou-se que 19 bibliotecas entravam no critério dos três desvios padrões. Estas bibliotecas estranhamente apresentaram uma porcentagem muito alta de seqüências com este tipo de contaminação. Uma análise mais cuidadosa mostrou que os íntrons não canônicos tinham tamanhos entre 51 e 59 bases, enquantos os íntrons normais costumam ter 3.000 bases. Devido a isso, considerou-se que estes casos podiam ser causados por erros na produção das bibliotecas. Todas as 10.971 seqüências contendo íntrons não canônicos foram descartadas e 7.862 possíveis predições espúrias foram eliminadas.

Por ser baseado em estatísticas de características de uma biblioteca, este método pode não funcionar bem para bibliotecas muito pequenas.

A.18 Identification of human chromosome 22 transcribed sequences with ORF expressed sequence tags [30]

O objetivo deste trabalho é descrever a análise de 250.000 seqüências ORESTES produzidas a partir de mRNA humano e mostrar a importância desta técnica na tarefa de

descoberta de genes. O enfoque da análise se encontra nas seqüências de genes que são contidos no cromossomo 22.

As seqüências de genoma bacterial completo permitem que análises computacionais relativamente precisas e completas dos genes e regiões codificadoras. Em complexos genomas eucariotos, esta tarefa é bem mais difícil devido à fragmentação dos genes em pequenos éxons, divididos freqüentemente por grandes íntrons. Neste contexto, a determinação da seqüência completa do cromossomo 22 permitiu que uma análise mais detalhada dos mecanismos de predições de genes mostrasse resultados como apenas 20% de genes preditos corretamente com todos os éxons, 16% de todos os éxons conhecidos que não foram preditos e 40% de genes preditos que não foram confirmados por outros meios.

Devido a essa ausência de técnicas computacionais adequadas para a predição de genes em eucariotos, a descoberta de genes ainda depende do alinhamento de seqüências, experimentalmente validadas, com o genoma completo do organismo. Resultados anteriores a este trabalho indicavam 545 genes ligados ao cromossomo 22 (247 obtidos através do seqüenciamento de transcritos completos, 150 identificados por similaridade com outros genes e 148 resultantes do alinhamento da seqüência do cromossomo com ESTs), o que era um valor extremamente baixo se comparado com os 35.000 genes humanos estimados. Isto mostrava que o número de seqüências de cDNA completos ou de ESTs era insuficiente para definir os transcritos humanos com confiabilidade.

A utilização de seqüências ORESTES [22] pode constituir numa estratégia para melhorar a descoberta de genes. Como os ORESTES são trechos parciais de seqüências expressas derivados da porção central do gene, eles são complementares aos ESTs que possuem origem nas extremidades 3' e 5' dos transcritos. ORESTES e ESTs unidos podem fornecer a seqüência completa do transcrito.

Uma base de 250.000 ORESTES foi criada a partir de mRNA derivados de diversos tecidos humanos, inclusive de tumores. Uma análise preliminar mostrou que 18% das seqüências tinham origem em transcritos de RNA ribossomal e em DNA mitocondrial, ou compostos quase que totalmente por seqüências repetitivas. Estas seqüências foram excluídas e o restante foi clusterizado com o programa CAP3, que produziu 81.429 clusters. Antes da comparação das seqüências com o cromossomo 22, elas eram processadas com o programa RepeatMasker [44]. As seqüências processadas eram comparadas com o cromossomo 22 com a utilização do BLAST, e aquelas que possuíssem 94% de identidade ao longo de 80% de seus tamanhos eram consideradas como significantes.

Dos 81.429 clusters, 1.181 (1,45%) tiveram seqüências equivalentes no cromossomo 22. O cromossomo 22 equivale a apenas 1,1% de todo o genoma humano, e o grande número de seqüências encontradas refletiu a alta densidade de genes existente nele. Além disso, o cromossomo possui um grande número de seqüências altamente expressas.

Os alinhamentos dos ORESTES com as seqüências do cromossomo 22 produziram

clusters para 162 (65,6%) dos 247 genes completos conhecidos até então.

Dois novos casos de tradução alternativa no cromossomo 22 foram identificados com auxílio das seqüências ORESTES

Foram obtidos clusters relacionados a 67 (44.6%) das 150 seqüências que possuíam semelhança com outros genes. Além disso, 15% das 150 seqüências foram totalmente cobertas pelos clusters.

Dos 148 genes preditos por seqüências ESTs no cromossomo 22, 45 (30,4%) foram confirmadas pela sobreposições entre os ESTs e as seqüências ORESTES. Esta baixa porcetagem era esperada por que os ORESTES tendem a ser complementares aos ESTs, não necessariamente com a ocorrência de sobreposição. As seqüências preditas pelos ESTs tinham em média tamanho de 1.022 bp, e ao serem complementadas com os ORESTES que apresentaram sobreposição, o tamanho médio cresceu para 1.153 bp, uma extensão de 13%.

Ao todo, 50,5% dos genes humanos anotados apresentaram alguma simililaridade significante com os ORESTES. Em comparação, todos os ESTs convencionais disponíveis foram clusterizados usando o CAP3 e representaram 48,8% dos genes. Isso indica que, aparentemente, uma base de ORESTES contendo 250.000 seqüências é tão informativa quanto uma base 10 vezes maior de ESTs, encontrada no dbEST (2000).

Análises adicionais revelaram um conjunto de 219 clusters relacionados às regiões não anotadas do cromossomo 22. Das 219 seqüências, 171 foram confirmadas por ESTs encontrados no dbEST. Como os ORESTES não possuem uma posição fixa no transcrito como os ESTs, que ficam nas extremidades, é possível que o número de genes identificados seja menor que 219, o que deve ser confirmado através da construção de clusters completos ou do seqüenciamento de cDNAs completos relativos a estes clusters.

Por ser bastante informativo e por ser fácil de produção, a técnica ORESTES pode ser de grande importância na cobertura de todos os transcritos humanos.

A.19 An optimized protocol for analysis of EST sequences [62]

Este trabalho descreve a avaliação de uma série de programas de montagem de seqüências para avaliar qual é o melhor e mais confiável para este tipo de tarefa. O programa considerado mais confiável foi utilizado no protocolo de contrução do banco TIGR Gene Indices.

O banco TIGR Gene Indices utiliza algoritmos de montagem de seqüência, ao invés da clusterização para produzir consensus (TC - *tentative consensus*), que representem os transcritos. Este procedimento tem vantagens como, por exemplo, separar genes relaci-

onados proximamente em consensus diferentes, separar variações de tradução e produzir seqüências mais longas. Contudo, a qualidade e a utilidade das seqüências montadas depende da habilidade do programa de montagem gerar, de forma efetiva, consensus altamente confiáveis a partir dos dados ESTs disponíveis.

Entre os programas de montagem desenvolvidos para projetos de seqüenciamento, os mais usados são: o phrap [44], o CAP3 [54] e o TIGR Assembler [101]. O TIGR Assembler original era otimizado para montagem de ESTs, enquanto a versão mais atual é otimizada para montagem de seqüências genômicas. Assim, a versão original será identificada como TA-EST e a outra como TIGR Assembler.

As seqüências ESTs representam uma série de problemas computacionais distintos para os programas de montagem. Em um projeto de seqüenciamento baseado em shotgun, que tipicamente utiliza um único clone, duas seqüências com menos de 98% de identidade podem ser consideradas como vindo de diferentes cópias de um elemento repetitivo de seqüência. Em contraste, os ESTs têm origem em variadas fontes representando o espectro de polimorfismos das amostras. Além disso, existem os erros, causados pelo seqüenciamento realizado em uma única passagem, altas taxas de inclusões e deleções, contaminações por vetores e adaptadores. Tudo isso, faz com que o grau de identidade de duas seqüências que se sobrepõem seja menor do que para o seqüenciamento genômico.

Utilizando as seqüências de rato existente no dbEST (2000), os quatro programas listados acima foram avaliados para determinar qual era o mais confiável na geração de consensos, para comparar o número de consensos e de seqüências únicas produzidos e para avaliar a performance relativa.

As seqüências obtidas foram trimadas para remoção de vetores, poli-A, poli-T, adaptadores e contaminações bacteriais. Os 118.473 ESTs limpos foram clusterizados através da comparação de todos os pares de seqüências com o programa WU-BLAST [4]. As seqüências que apresentaram $\geq$95% de identidade em regiões que tivessem pelo menos 40 bases e não tivessem saltos de mais de 20 bases diferentes foram agrupadas nos mesmo cluster. Um total de 16.183 clusters foram gerados.

Os programas foram executados com as seqüências de cada cluster. Os parâmetros padrões foram utilizados. Os resultados mostraram que todos os programas produziram o mesmo número de seqüências montadas. Os programas TA-EST e TIGR-Assembler produziram, respectivamente, 20 e 54 vezes mais seqüências únicas que os programas phrap e o CAP3, o que mostra que eles são bem menos tolerantes em relação as discrepâncias que existem entre as seqüências. A grande diferença entre os dois primeiros e os dois últimos, é que a construção de clusters com dezenas ou centenas de seqüências, realizada pelo phrap e pelo CAP3, mostrando que eles são capazes de montar os ESTs corretamente, apesar da presença de erros de seqüenciamento e de polimorfismos.

Uma primeira análise nos dados produzidos apontaria o phrap como melhor que o

CAP3, mas um teste de qualidade de consenso provou o contrário. O teste foi feito utilizando ESTs de genes anotados. As seqüências foram montadas pelos quatro programas e os consensos comparados com as seqüências de referência (genes). Por exemplo, no caso do gene do citocromo c oxidase sub-unidade II, o CAP3, o TA-EST e o TIGR Assembler foram capazes de reproduzir a seqüência de referência. Contudo, o CAP3 foi capaz de agrupar todas as seqüências, enquanto os outros dois agruparam algumas seqüências de baixa qualidade em um segundo consenso, que representava apenas uma parte da seqüência do gene. O phrap foi capaz de juntar todas as seqüências em um único consenso, mas a seqüência resultante apresentou um grande número de inserções e outros erros, gerando uma taxa de 5% de erros.

Os erros gerados pelas seqüenciamento automático de DNA são concentrados no início e no final das seqüências. Em projetos de seqüenciamento genômico, a distribuição das seqüências ao longo do genoma conseguem compensar este fato, mas em projetos ESTs isto não ocorre. Em geral, os erros em ESTs se concentram no mesmo local, pois o seqüenciamento do cDNA se inicia aproximadamente na mesma posição, e os programas de montagem devem ser capazes de contornar esta situação.

Para testar este aspecto, um modelo de distribuição de erros ao longo de uma seqüência foi criado. A partir de uma seqüência de referência de 600 bases, um conjunto de seqüências com tamanhos entre 450 e 550 bases foram criadas segundo o modelo de distribuição de erros. Taxas de erros de 1% a 8% foram simuladas. Estas seqüências foram fornecidas aos programas e, novamente, verificou-se a produção de seqüências únicas e consensos e as qualidades dos consensos. Os programas CAP3 e phrap conseguiram agrupar as seqüências em um único consenso. Em contraste, os programas TA-EST e TIGR Assembler dividiam os ESTs em seqüências únicas ou mais consensos conforme a taxa de erro crescia. Neste teste, o programa CAP3 mostrou novamente melhores resultados que o phrap. As qualidades dos melhores consensos dos programas TA-EST e TIGR Assembler se aproximavam da do CAP3, mas isto foi considerado menos importante do que o fato destes programas estarem gerando mais consensos do que deveriam. Os resultados também mostraram que o phrap tende a reter os erros das seqüências ESTs, gerando inserções nos consensos.

Além de saberem lidar com os erros, os programas devem ser capazes de separar ESTs de transcritos distintos mas proximamente relacionados. Para avaliar a capacidade dos programas lidarem com dados de famílias gênicas, foi criada uma família gênica composta de seqüências modelo. A partir de inserções e substituições feitas em um segmento de 1800 bp do gene ECA1, 7 seqüências foram geradas contendo 99, 98, 97, 96, 95, 94 e 90% de identidade com a seqüência original. As seqüências geradas e a original representariam, cada uma, um membro da família. Elas foram fragmentadas em trechos que iam de 450 a 550 bases, como se tivessem passado por um processo de shotgun. Dois conjuntos de

fragmentos foram criados. O primeiro continha as seqüências dos 8 membros da família e o segundo apenas de 6 membros (95% ou mais de identidade). Os programas foram executados com estes dois conjuntos em uma bateria de 6 testes independentes. O programa, TA-EST não era capaz de distingüir as seqüências, sempre agrupando 6 membros de uma família em um único consenso e 8 membros de outra família em uma média de 1,38 consensos. O phrap não se saiu melhor, gerando 2 e 4,33 consensos, respectivamente, para os cojuntos de 6 e 8 membros. O CAP3 produziu bons resultados na separação das seqüências, mas para ESTs que compartilhassem >96% de identidade, ele também falhou, gerando 4,5 e 6,67 consensos. O programa TIGR Assembler forneceu a maior discriminação, gerando em média 5,83 e 8,5 consensos.

Um teste para validar os resultados obtidos nas simulações anteriores foi feito. Um conjunto de 73 genes humanos altamente representados foi utilizado como entradas dos programas. Mais uma vez os resultados apontaram o CAP3 como melhor programa.

O CAP3 apresentou os melhores resultados e, por isso, foi o programa escolhido para ser o montador oficial do banco TIGR Gene Indices.

A.20 d2_cluster: A Validated Method for Clustering EST and Full-Length cDNA Sequences [20]

O objetivo deste trabalho é descrever um método de clusterização de seqüências chamado d2_cluster e apresentar resultados que validem o método.

O d2_cluster é um método de clusterização aglomerativo. Cada seqüência começa em seu próprio cluster e a clusterização é feita por uma série de fusões dos clusters. O d2_cluster pode ser descrito em termos de clusterização através de ligação mínima ou fecho transitivo.

O termo fecho transitivo se refere à propriedade que quaisquer duas seqüências que possuem um determinado nível de similaridade estarão no mesmo cluster. Assim, duas seqüências A e B estarão no mesmo cluster, mesmo que elas não tenham similaridade, se existir uma seqüência C com similidaridade suficiente em relação a A e a B.

O critério de união dos clusters é a detecção de duas seqüências que possuam uma janela de tamanho w bases com um percentual mínimo s de identidade. O método só leva em conta a informação de sobreposição, e nenhuma informação de anotação é utilizada na clusterização.

Os critérios de detecção de sobreposição são descritos em trabalhos anteriores [47, 107, 118].

Para melhor explicação do algoritmo, seguem algumas convenções:

1. A distância d2 entre duas seqüências A e B é denotada por $d2(A, B)$. A notação

$d2(A, B)$ é utilizada por conveniência pois a função d2 contém outros parâmetros, definidos nos outros trabalhos já citados, além das seqüências A eB.

2. O conjunto a ser clusterizado é composto por N seqüências numeradas de 0 a $(N-1)$. A seqüência i é denotada por S_i.

3. A relação de pertinência de uma seqüência S_i a um cluster é denotada pelo valor C_i.

4. Dados dois clusters i e j, a operação de fusão $MERGE(i, j)$ significa que todas as seqüências do cluster j serão atribuídas ao cluster i. Dessa maneira temos que na operação $MERGE(i, j)$, para todas as seqüências S_r que possuem $C_r = j$, C_r é atualizado de forma que $C_r = i$.

A progressão do algoritmo pode ser mostrada por indução. A iterações I_0 e I_1 são exibidas e a seguir descreve-se como a iteração I_k acontecerá:

- *Estado inicial*: Cada seqüência está em seu próprio cluster (para todo S_i, $C_i = i$).

- *Iteração* I_0: A seqüência S_0 é escolhida. Para cada seqüência $S_i, 1 \leq i < N$, $MERGE(C_0, C_i)$, se $d2(S_0, S_i) < THRESHOLD$, onde $THRESHOLD$ é um valor máximo pré-definido.

- *Iteração* I_1: A seqüência S_1 é escolhida. Note que $C_1 = 1$ a não ser que a seqüência tenha sido atribuída ao cluster 0 na iteração anterior. Para cada seqüência $S_i, 2 \leq i < N$, $MERGE(C_1, C_i)$, se $d2(S_1, S_i) < THRESHOLD$.

- *Iteração* I_k: A seqüência S_k é escolhida. Para cada seqüência $S_i, k \leq i < N$, $MERGE(C_k, C_i)$, se $d2(S_k, S_i) < THRESHOLD$.

A clusterização termina após $N - 1$ iterações. O fecho transitivo é obtido porque os clusters são fundidos se eles possuem identidade suficiente.

O método d2_cluster foi comparado com os dados existentes no banco UniGene. Este banco foi escolhido para comparação devido à alta qualidade do seus dados e pela ampla aceitação que ele possuía. Assim, as seqüências de rato existentes na versão 19, de agosto de 1998, foram obtidas e submetidas ao método d2_cluster. Como o algoritmo do UniGene não estava disponível, as seqüências não foram clusterizadas segundo este método. Os clusters produzidos pelo d2_cluster foram comparados com os existentes no banco UniGene.

O banco de seqüências de rato do UniGene, na versão 19, possuía 43.612 ESTs e seqüências completas de mRNA. Como a informação de limpeza das seqüências não eram

disponíveis no UniGene, elas tiveram que ser trimadas para remoção de seqüências repetitivas e seqüências mitocondriais, o que foi feito com o programa cross_match [44].

As 42.441 seqüências restantes foram clusterizadas com o d2_cluster com os parâmetros: window_size = 100, stringency = 0.9, min_seq = 100 e rev_comp = 1. Estes parâmetros indicam que duas seqüências seriam colocadas no mesmo cluster se elas possuíssem uma janela de pelo menos 100 bases com 90% de identidade. Indicam também que as seqüências com menos de 100 bases seriam descartadas e que o complemento reverso também seria considerado. O procedimento de clusterização tomou aproximadamente 31 horas de processamento em uma máquina SUN E450 com um processador de 400MHz. Após o processo, cada seqüência pertencia a exatamente um cluster d2_cluster e a um cluster UniGene.

O método d2_cluster produziu aproximadamente 20% menos clusters com apenas uma seqüência e reduziu o número total de clusters em aproximadamente 10%. Em geral, o número de clusters com poucas seqüências diminuiu e o número de clusters com muitas seqüências aumentou.

Verificou-se que 60 clusters (menos que 0,5%) do UniGene eram fusões de clusters do d2_cluster. No entanto, 1078 clusters do d2_cluster (aproximadamente 8%) eram fusões de clusters do UniGene, o que indica que o método d2_cluster é mais agressivo na união de seqüências. Um total de 12.389 clusters (83% de clusters do UniGene e 90% dos clusters do d2_cluster) eram idênticos, mostrando que o resultado produzido pelos dois algoritmos são consistentes em larga escala.

A junção realizada por um método de seqüências, que foram colocadas em cluster de tamanho 1 pelo outro, pode ser explicada por vários motivos: um método pode falhar ao não unir duas seqüências; um método pode introduzir uma falsa união; ou os critérios de clusterização podem ser diferentes. O UniGene, por exemplo, utiliza informação do processo de clonagem, enquanto o d2_cluster não utiliza nenhuma informação sobre origem das seqüências ou anotação. Alguns casos observados, mostraram que clusters que não foram unidos pelo UniGene, mas foram unidos pelo d2_cluster, deveriam realmente ser unidos.

Testes foram realizados para avaliar a taxa de erros do método d2_cluster. Primeiro avaliou-se a ocorrência da união de seqüências que não deveriam ser unidas em um mesmo cluster. Neste ponto, é preciso observar que existem casos especiais na avaliação deste tipo de erro. Por exemplo, gene parálogos que têm origem em genes distintos podem ser perfeitamente alinhados. Alguns casos, como a tradução alternativa, necessitam de mais que um simples consenso para representar o cluster inteiro. Em casos como estes, este tipo de erro pode ser excluído se o consenso do cluster tiver domínios suficientemente largos de identidade com cada uma das traduções.

Os clusters produzidos pelo d2_cluster foram alinhados com o programa CRAW [21].

Este programa foi configurado de modo que o alinhamento de uma seqüência com o consenso do subcluster não pudesse ter uma janela de 50 bases com mais que 10% de erros. Empiricamente foi definido um limite superior para este tipo de erro. Após processamento dos 13.755 clusters produzidos pelo método, todos, com execeção de 1617, podiam ser representados por um único consenso produzido pelo CRAW, o que resultou em um limite superior inicial de 11,8%. Um limite superior mais rígido foi obtido com a inspeção do alinhamento múltiplo dos 1617 clusters com o múltiplos consensos produzidos pelo CRAW para identificar casos que possuíssem janelas de 100 bases com pelo menos 90% de identidade. Um total de 106 clusters não satisfizeram este critério e o limite superior ficou em 0,8%.

Para avaliação do erro de não unir seqüências que deveriam ser unidas, o procedimento adotado foi a comparação de todas seqüências contra todas seqüências com a utilização do algoritmo de Smith-Waterman [94]. Os erros seriam identificados por um subconjunto de todas as similaridades intercluster identificadas pelo algoritmo. Os testes mostraram que 51 clusters apresentaram este tipo de erro resultando em uma taxa de aproximadamente 0,4%.

O d2_cluster foi utilizado no projeto STACK, onde os ESTs são hierarquicamente clusterizados dentro de tecidos e categorias arbitrárias. Neste projeto, ele foi configurado para unir seqüências que apresentassem mais que 96% de identidade em uma janela de 150 bases.

A.21 A new DNA sequence assembly program [17]

Este trabalho descreve um programa de montagem de seqüências de DNA chamado GAP. Os algoritmos implementados por este programa foram escritos em ANSI C e FORTRAN 77, e a interface gráfica foi escrita em Tcl/Tk. Ele pode ser utilizado com X Windows em máquinas SUN, DEC e SGI UNIX. O desenvolvimento deste software focou principalmente no fato de que os resultados produzidos pelos algoritmos pudessem ser visualizados graficamente, permitindo melhor análise dos dados.

O GAP possui a implementação de três algoritmos de montagem diferentes.

O primeiro é o algoritmo padrão de montagem de shotgun. Ele considera uma seqüência por vez e comparam com todo o dado que já foi montado. Se a seqüência atender os critérios mínimos, ela é alinhada. Se o alinhamento for considerado bom o suficiente, ele será registrado. Se uma seqüência possuir um bom alinhamento com dois contigs diferentes, ela é alinhada com um dos contigs e o resultado é alinhado com o outro. Se o alinhamento for bom, os dois contigs são unidos. Se a seqüência alinhar bem com mais de 2 contigs, os dois melhores serão considerados. Se uma seqüência não possuir correspondência com nenhum contig, ela iniciará um contig novo. E, finalmente, se uma

seqüência tiver correspondência, mas não alinhar bem, ela poderá iniciar um contig novo ou ser descartada.

O segundo algoritmo realiza montagem em regiões com fita única. O funcionamento do algoritmo é semelhante ao anterior, com exceção que novas seqüências só serão unidas em regiões que possuem apenas uma fita ou que fazem fronteira ou sobrepõem regiões deste tipo. As regiões que possuem as duas fitas completas já têm informação suficiente e por isso não necessitam que novas seqüências sejam adicionadas.

O último algoritmo é denominado Montagem Direta. Ele necessita que cada seqüência tenha uma posição de montagem, que indica onde a seqüência será montada. Esta posição não é absoluta, mas relativa à qualquer outra seqüência (seqüência âncora) que já tenha sido montada. A definição desta posição de montagem inclui o nome da seqüência âncora, o sentido da seqüência que será montada, o deslocamento em relação à âncora e um valor de tolerância em relação ao deslocamento. O valor de tolerância é normalmente positivo e indica que a primeira base da nova seqüência deve estar dentro de ± 'tolerância' bases de distância do deslocamento. Se a tolerância é negativa, o alinhamento não é feito e a seqüência é simplesmente colocada na posição 'deslocamento' relativa à âncora. Se a o nome da seqüência âncora é '*new*', a seqüência iniciará um novo contig. O algoritmo pega uma seqüência, lê seus atributos, busca a seqüência âncora na base de dados e recupera o consenso para a região definida por seqüência âncora, deslocamento e tolerância. Feito isso, o algoritmo realiza o alinhamento e verifica o número de erros. Se o número de erros estiver dentro do limite, estão a seqüência é inserida no contig.

O software possui um série de funcionalidades que podem ser acessadas através da interface gráfica. Essas funcionalidades permitem que os contigs criados sejam visualizados, comparados e editados.

O software pode ser obtido através do envio de uma mensagem para Roger Staden (*rs@mrc-lmb.cam.ac.uk*).

A.22 TIGR Gene Indices clustering tools (TGICL): a software system for fast clustering of large EST datasets [78]

Este artigo tem o objetivo de descrever aspectos do software TGICL. Esta ferramenta realiza um conjunto de operações para análise de ESTs e de bancos de mRNA.

O TGICL opera de forma a clusterizar com base em similaridade as seqüências fornecidas. Após esta primeira etapa, cada cluster é montado individualmente para produção de seqüências consensos. Opcionalmente, o programa pode utilizar valores de qualidades das seqüências para produção de melhores resultados.

As seqüências fornecidas devem ser trimadas antes de serem utilizadas como entrada do programa, pois o TGICL não realiza este serviço.

Trechos de repetições conhecidos podem ser mascarados para que o TGICL exclua-os durante a fase de preparação do hash de palavras utilizado na clusterização. O mascaramento deve ser feito de forma que os trechos de repetições fiquem em letras minúsculas.

Transcritos completos conhecidos podem ser utilizados como "sementes" para a clusterização. A utilização de transcritos completos ajuda na criação de clusters menores e melhores particionados e evita a montagem de clusters quiméricos. A extensão lateral das "sementes" só é permitida quando a seqüência alinha com o transcrito quase que perfeitamente.

A clusterização realiza primeiro uma indexação do arquivo de entrada para posterior busca de similaridade entre todos pares de seqüências utilizando uma versão modificada do programa megablast [121]. Esta versão oferece filtros específicos para o resultados, utiliza deslocamento dinâmico durante buscas incrementais na base de dados de seqüências e produz uma saída onde cada linha é uma sobreposição identificada.

Os alinhamentos são classsificados em ordem decrescente de pontuação de alinhamento. Esta ordenação global permite a utilização de uma estratégia "gulosa" para controle da clusterização. Os melhores alinhamentos são encontrados primeiros e direcionam a formação dos clusters.

O critério padrão para a sobreposição de seqüências é de no mínimo 40 bases. O percentual mínimo de similaridade padrão é de 95%. Para clusterização com utilização de "sementes", as sobreposições das seqüências com os transcritos completos devem ter cobertura próxima da completa, caso contrário, a semente é ignorada.

A montagem dos clusters é realizada com a utilização do programa CAP3. Este realiza o alinhamento múltiplo das seqüências e a criação do consenso.

O software tem dificuldades de lidar com genes altamente expressos que formam clusters com muitos milhares de seqüências. Neste caso, o CAP3 sofre problemas de falta de memória. Para contornar esse problema, o usuário pode utilizar os utilitários sclust e nrcl, que estão no pacote do TGICL, para realizar montagem iterativa destes clusters.

O TGICL foi desenvolvido e testado sob uma plataforma Linux e tem a capacidade de executar em máquinas multi-processadas. O programa principal é um script perl e as ferramentas auxiliares utilizadas por este script, incluindo a versão modificada do megablast, foram desenvolvidas em linguagem C.

O programa está disponível em `http://www.tigr.org/tdb/tgi/software/`.

A.23 ESTIMA, a tool for EST management in a multi-project environment [59]

O objetivo deste artigo é descrever o software ESTIMA e os detalhes de sua implementação, apresentando as funcionalidades que ele possui.

O software ESTIMA foi desenvolvido com a finalidade de gerenciar projetos de seqüenciamento de ESTs, sendo que ele possui a capacidade de controlar múltiplos projetos simultaneamente.

Esta ferramenta realiza, basicamente, a recepção de dados, processados ou não, e os armazena em uma base de dados diferente para cada projeto, com o auxílio de um conjunto variado de scripts desnvolvidos em Perl. Além disso, ela possui um conjunto de buscas que podem ser feitas via interface web para que os dados armazenados possam ser analisados. A interface web permite fácil visualização dos contigs produzidos pelo projeto, assim como as anotações realizadas e a relação que as seqüências ou contigs têm com os dados de Gene Ontology [28].

Uma característica importante do ESTIMA é que ele é independente de pipelines de processamento de ESTs. Cada projeto realiza o processamento dos cromatogramas produzidos da maneira que for mais adequada as suas necessidades. O programa apenas recebe os dados e os disponibiliza para análise.

Entre os projetos de seqüenciamento de ESTs que utilizam o ESTIMA, podemos citar os dos organismos: *Apis mellifera* (abelha), *Bos taurus* (gado), *Taeniopygia guttata* (passáro mandarim), *Diabrotica vergifera* (verme que ataca a raiz do milho), *Ictalurus punctatus, Ictalurus furcatus* (catfish) e *Malus x domestica* (maçã).

Dentre os organismos citados acima, 3 são objetos de estudos de projetos disponíveis publicamente. A abelha é estudada pelo projeto Honey Bee Brain EST [49], o gado é estudado pelo projeto Cattle EST [23] e o passáro mandarim pelo projeto Songbird Neurogenomics Initiative [97].

O software é disponível gratuitamente em `http://titan.biotec.uiuc.edu/ESTIMA` para uso acadêmico.

A.24 Comparative analysis of 82 expressed sequence tags from a cattle ovary cDNA library [64]

O trabalho apresentado neste artigo descreve o processo de seqüenciamento e análise de 82 ESTs provenientes de 51 clones aleatoriamente selecionados de uma biblioteca de cDNA proveniente do tecido de ovário de indivíduos da espécie *Bos taurus*. Além disso, o trabalho apresenta uma metodologia para análise comparativa de genomas, chamada

COMPASS.

A montagem de mapas genéticos pode facilitar o estudos para identificação de genes que afetam processos economicamente importantes, como a produção de leite ou carne, por exemplo. Além disso, pode fornecer informações para o entendimento da evolução das espécies. Sendo assim, mapeamentos genéticos de organimos, como o boi, começaram a ser realizados.

Na época em que este trabalho foi realizado (1998), existiam aproximadamente 300 genes atribuídos a cromossomos do gado, e a maior parte deles tinha sido obtida através de mapeamento com a utilização de métodos físicos. Como alternativa complementar a estas técnicas, surgia a caracterização e mapeamento de ESTs que permitiria que o índice gênico fosse obtido de forma mais rápida.

Estudos mostravam que a comparação dos ESTs disponíveis entre espécies diferentes sugeriam que uma grande fração dos ESTs exibiam similaridade suficiente para que genes ortólogos pudessem ser identificados [65]. No entanto, a simililaridade de seqüência não era grande o suficiente para que primers desenvolvidos em uma espécie para amplificação por PCR, pudessem ser usados de maneira confiável em outras. Por exemplo, somente 50% ou menos dos primers desenvolvidos segundo a estratégia CATS (*Comparative anchor-tagged sequences*) [63] aplificaram o produto correto em diferentes espécies.

Aliado ao fator descrito acima, a pouca disponibilidade de seqüências de gado dificultava a criação de um mapa comparativo entre bovinos e humanos. Existia portanto a necessidade de se produzir bibliotecas de cDNAs de tecidos específicos de diferentes espécies para a criação de mapas. Neste trabalho se optou pelo tecido do ovário porque este seria fonte de genes associados à divisão celular, à reprodução e ao desenvolvimento.

A biblioteca de ovário foi produzida a partir de um tecido saudável. Os cDNAs foram escolhidos aleatoriamente e seqüenciados, em sua maioria, nas duas extremidades. As seqüências obtidas foram trimadas para remoção de vetor e de trechos de baixa qualidade. O programa BLAST foi utilizado para busca de simililaridade entre as seqüências e os bancos nr e dbEST. O limite adotado para relacionar um gene humano com a seqüência bovina era de pelo menos 75% de homologia sobre um trecho contínuo de 60 nucleotídeos.

Análises preliminares da biblioteca mostraram que 22% (31/141) clones selecionados aleatoriamente apresentavam ausência de inserto ou inserto com tamanho menor que 200 bp. Nos 110 clones restantes, existiam 5 duplicatas. Um total de 164 seqüências aproveitáveis foram produzidas a partir dos 105 clones únicos. Estas seqüências possuíam tamanho médio de 458 bp. Cinqüenta e quatro clones que apresentaram elementos repetivivos, seqüências de íntrons, seqüências mitocondriais ou ribossomais foram eliminados.

A partir dos 51 clones escolhidos, 82 seqüências foram obtidas. Destas, 22 (43,1%) possuíam extremidades 5' e/ou 3' que correspondiam a outras seqüências conhecidas de humanos ou outros mamíferos, 18 (35,3%) correspondiam a ESTs humanos ou outros

ESTs, e 11 (21,6%) representam transcritos novos.

Para confirmar a identificação de ortólogos e testar a utilidade do mapeamento EST para o mapeamento gênico comparativo, 11 pares de primers oligonucleotídeos foram desenvolvidos a partir dos ESTs que foram considerados supostos ortólogos humanos. Todos os 11 pares amplificaram o produto correto.

Entre os 11 genes representados pelos ESTs bovinos mapeados, 4 haviam sido mapeados anteriormente em humanos e as localizações destes 4 genes no mapa eram consistentes com informações de mapeamento comparativo produzidas por trabalhos anteriores. Apesar dos outros 7 genes não terem sido mapeados anteriormente, as suas localizações no mapa humano puderam ser preditas com a utilização das informações dos mapeamentos comparativos disponíveis.

No sentido contrário, as informações de oito ortólogos, identificados por similaridade de seqüência, do mapa humano permitiram que 4 ESTs bovinos pudessem ser distribuídos em cromossomos.

Um EST bovino já havia sido mapeado em ambas as espécies. Apenas 2 ESTs com ortólogos conhecidos não puderam ser preditos com auxílio das informações comparativas disponíveis.

A análise descrita acima foi denominada pelos autores como COMPASS (*comparative mapping by annotation and sequence similarity*). Ela difere de outras estratégias como, por exemplo, a CATS ao usar informação de ESTs homólogos e identificação de ortólogos através de busca por similaridade utilizando o BLAST.

A estratégia CATS utiliza informações de seqüências de genes previamente mapeados para produção de primers para amplificação de ortólogos que poderão ser então mapeados por outros métodos. No entanto, ela não se mostrou robusta.

Devido as suas características, a estratégia COMPASS é mais adequada a operações de larga-escala e aliada a outras técnicas pode ser de grande utilidade para o desenvolvimento de mapas gênicos e para o entendimento da evolução dos cromossomos dos mamíferos.

A.25 An Ordered Comparative Map of the Cattle and Human Genomes [8]

O trabalho apresentado neste artigo é uma continuação do trabalho iniciado por R. Z. Ma *et al.* [64]. Um mapa comparativo dos genomas completos de humanos e bovinos foi construído com a utilização do mapeamento por Radiação Híbrida paralela (RH) em conjunto com o seqüenciamento EST, a busca em bancos de dados por genes de gado não mapeados e a estratégia preditiva COMPASS para identificação de regiões homólogas específicas.

Neste trabalho foram utilizadas bibliotecas de cDNAs provenientes de tecidos extraídos do ovário e do baço de uma vaca saudável adulta da raça Aberdeen-Angus. As bibliotecas foram construídas com a utilização do vetor pBluescript SK(±). O primer T3 (5' - AATTAACCCTCACTAAAGGG - 3') foi utilizado no seqüenciamento na direção 5' e o primer T7 modificado (5' - TACGACTCACTATAGGGCGAAT - 3') foi utilizado no seqüenciamento na direção 3'. Os ESTs provenientes do ovário foram seqüenciados em ambas as direções, e os provenientes do baço foram seqüenciados apenas na direção 3'. Os cromatogramas foram processados manualmente com a utilização do software SeqEd da Applied Biosystems. As seqüências foram trimadas para eliminação de vetores. O programa BLASTN foi utilizado para busca por similaridade nos bancos dbEST e nr. Clones contendo RNA mitocondrial, RNA ribossomal ou elementos repetitivos foram removidos do conjunto de dados.

Um total de 768 genes foram colocados no mapa RH em adição a 319 microsatélites utilizados como marcadores âncoras. Destes genes, 638 possuíam ortólogos humanos com dados mapeados, permitindo a construção de um mapa comparativo ordenado. O grande número de *loci* ordenados revelaram pelo menos 105 segmentos conservados entre os dois genomas.

O mapa comparativo sugere 41 eventos de translocação, um mínimo de 54 rearranjos internos e o reposicionamento de todos, menos um, centrômeros podem ser observados na organização dos dois genomas.

Adicionalmente, a estratégia COMPASS mostrou 95% de precisão na predição da localização em cromossomos do gado a partir de dados aleatórios, demonstrando sua eficiência na identificação de regiões específicas para um mapeamento detalhado.

O mapa comparativo produzido servirá como importante fonte de informação para elucidar a filogenia dos cromossomos dos mamíferos e para identificação de genes economicamente importantes.

Bibliografia

[1] M. D. Adams, S. E. Celniker, R. A. Holt, C. A. Evans, J. D. Gocayne, P. G. Amanatides, S. E. Scherer, P. W. Li, R. A. Hoskins, R. F. Galle, R. A. George, S. E. Lewis, S. Richards, M. Ashburner, S. N. Henderson, G. G. Sutton, J. R. Wortman, M. D. Yandell, Q. Zhang, L. X. Chen, R. C. Brandon, Y.-H. C. Rogers, R. G. Blazej, M. Champe, B. D. Pfeiffer, K. H. Wan, C. Doyle, E. G. Baxter, G. Helt, C. R. Nelson, G. L. G. Miklos, J. F. Abril, A. Agbayani, H.-J. An, C. Andrews-Pfannkoch, D. Baldwin, R. M. Ballew, A. Basu, J. Baxendale, L. Bayraktaroglu, E. M. Beasley, K. Y. Beeson, P. V. Benos, B. P. Berman, D. Bhandari, S. Bolshakov, D. Borkova, M. R. Botchan, J. Bouck, P. Brokstein, P. Brottier, K. C. Burtis, D. A. Busam, H. Butler, E. Cadieu, A. Center, I. Chandra, J. M. Cherry, S. Cawley, C. Dahlke, L. B. Davenport, P. Davies, B. de Pablos, A. Delcher, Z. Deng, A. D. Mays, I. Dew, S. M. Dietz, K. Dodson, L. E. Doup, M. Downes, S. Dugan-Rocha, B. C. Dunkov, P. Dunn, K. J. Durbin, C. C. Evangelista, C. Ferraz, S. Ferriera, W. Fleischmann, C. Fosler, A. E. Gabrielian, N. S. Garg, W. M. Gelbart, K. Glasser, A. Glodek, F. Gong, J. H. Gorrell, Z. Gu, P. Guan, M. Harris, N. L. Harris, D. Harvey, T. J. Heiman, J. R. Hernandez, J. Houck, D. Hostin, K. A. Houston, T. J. Howland, M.-H. Wei, C. Ibegwam, M. Jalali, F. Kalush, G. H. Karpen, Z. Ke, J. A. Kennison, K. A. Ketchum, B. E. Kimmel, C. D. Kodira, C. Kraft, S. Kravitz, D. Kulp, Z. Lai, P. Lasko, Y. Lei, A. A. Levitsky, J. Li, Z. Li, Y. Liang, X. Lin, X. Liu, B. Mattei, T. C. McIntosh, M. P. McLeod, D McPherson, G. Merkulov, N. V. Milshina, C. Mobarry, J. Morris, A. Moshrefi, S. M. Mount, M. Moy, B. Murphy, L. Murphy, D. M. Muzny, D. L. Nelson, D. R. Nelson, K. A. Nelson, K. Nixon, D. R. Nusskern, J. M. Pacleb, M. Palazzolo, G. S. Pittman, S. Pan, J. Pollard, V. Puri, M. G. Reese, K. Reinert, K. Remington, R. D. C. Saunders, F. Scheeler, H. Shen, B. C. Shue, I. Sidén-Kiamos, M. Simpson, M. P. Skupski, T. Smith, E. Spier, A. C. Spradling, M. Stapleton, R. Strong, E. Sun, R. Svirskas, C. Tector, R. Turner, E. Venter, A. H. Wang, X. Wang, Z.-Y. Wang, D. A. Wassarman, G. M. Weinstock, J. Weissenbach, S. M. Williams, T. Woodage, K. C. Worley, D. Wu, S. Yang, Q. A. Yao, J. Ye, R.-F. Yeh, J. S. Zaveri, M. Zhan, G. Zhang, Q. Zhao, L. Zheng, X. H. Zheng, F. N.

Zhong, W. Zhong, X. Zhou, S. Zhu, X. Zhu, H. O. Smith, R. A. Gibbs, E. W. Myers, G. M. Rubin, and J. C. Venter. The genome sequence of *Drosophila melanogaster*. *Science*, 287(5461):2185–2195, March 2000.

[2] M. D. Adams, J. M. Kelley, J. D. Gocayne, M. Dubnick, M. H. Polymeropoulos, H. Xiao, C. R. Merril, A. Wu, B. Olde, R. F. Moreno, A. R. Kerlavage, W. R. McCombie, and J. C. Venter. Complementary DNA Sequencing: Expressed Sequence Tags and Human Genome Project. *Science*, 252:1651–1656, June 1991.

[3] R. Altman, D. Brutlag, P. Karp, R. Lathrop, and D. Searls, editors. *A generalized profile syntax for biomolecular sequence motifs and its function in automatic sequence interpretation*, Menlo Park, USA, 1994. AAAI Press.

[4] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[5] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25:3389–3402, 1997.

[6] Applied Biosystems. *Automated DNA Sequencing - Chemistry Guide*, 1998. Part Number: 4305080B.

[7] O. T. Avery, C. M. MacLeod, and M. McCarty. Studies on the chemical nature of the substance inducing transformation of pneumococcal types. *J. Exp. Med.*, 79:137–158, 1944.

[8] M. R. Band, J. H. Larson, M. Rebeiz, C. A. Green, D. W. Heyen, J. Donovan, R. Windish, C. Steining, P. Mahyuddin, J. E. Womack, and H. A. Lewin. An Ordered Comparative Map of the Cattle and Human Genomes. *Genome Research*, 10:1359–1368, 2000.

[9] W. C. Barker, J. S. Garavelli, D. H. Haft, L. T. Hunt, C. R. Marzec amd B. C. Orcutt, G. Y. Srinivasarao, L. L. Yeh, R. S. Ledley, H. Mewes, F. Pfeiffer, and A. Tsugita. The PIR-International Protein Sequence Database. *Nucleic Acids Research*, 26(1):27–32, 1998.

[10] C. Baudet and Z. Dias. New EST Trimming Strategy. In J.C. Setubal and S. Verjovski-Almeida, editors, *Lecture Notes on Bioninformatics*, volume 3594, pages 206–209. Springer-Verlag Berlin Heildelberg, July 2005. Brazilian Symposium on Bioinformatics (BSB 2005).

[11] C. Baudet and Z. Dias. New EST trimming strategy. Technical Report IC-05-09, Institute of Computing - University of Campinas, May 2005.

[12] C. Baudet and Z. Dias. Analysis of slipped sequences in EST projects. *Genetics and Molecular Research*, 5(1):169–181, 2006.

[13] D. A. Benson, M. S. Boguski, D. J. Lipman, and J. Ostell. GenBank. *Nucleic Acids Research*, 22:3441–3444, 1994.

[14] M. S. Boguski, T. M. Lowe, and C. M. Tolstoshev. dbEST – database for "expressed sequence tags". *Nature Genetics*, 4(4):332–333, 1993.

[15] M. S. Boguski and G. D. Schuler. Establishment of a transcript map. *Nat. Genet.*, 10:369–371, 1995.

[16] M. Bonaldo, G. Lennon, and M. B. Soares. Normalization and subtraction: two approaches to facilitate gene discovery. *Genome Research*, 6:791–806, 1996.

[17] J. K. Bonfield, K. F. Smith, and R. Staden. A new DNA sequence assembly program. *Nucleic Acids Research*, 23:4992–4999, 1995.

[18] Brazilian Genome (BrGene) – The Virtual Institute of Genomic Research, September 2004. http://www.brgene.lncc.br.

[19] M. Browne. Critics see humbler origin in "dinosaur" DNA, June 1995. New York Times.

[20] J. Burke, D. Davison, and W. Hide. d2_cluster: A Validated Method for Clustering EST and Full-Length cDNA Sequences. *Genome Research*, 9:1135–1142, 1999.

[21] J. Burke, H. Wang, W. Hide, and D. Davison. Alternative gene form discovery and candidate selection from gene indexing projects. *Genome Research*, 8:276–290, 1998.

[22] A. A. Camargo, H. P. B. Samaia, E. Dias-Neto, D. F. Simão, I. A. Migotto, M. R. S. Briones, F. F. Costa, M. A. Nagai, S. Verjovski-Almeida, M. A. Zago, L .E. C. Andrade, H. Carrer, H. F. A. El-Dorry, E. M. Espreafico, A. Habr-Gama, D. Giannella-Neto, G. H. Goldman, A. Gruber, C. Hackel, E. T. Kimura, R. M. B. Maciel, S. K. N. Marie, E. A. L. Martins, M. P. Nóbrega, M. L. Paçó-Larson, M. I. M. C. Pardini, G. G. Pereira, J. B. Pesquero, V. Rodrigues, S. R. Rogatto, I. D. C. G. da Silva, M. C. Sogayar, M. F. Sonati, E. H. Tajara, S. R. Valentini, F. L. Alberto, M. E. J. Amaral, I. Aneas, L. A. T. Arnaldi, A. M. de Assis, M. H. Bengston, N. A. Bergamo, V. Bombonato, M. E. R. de Camargo, R. A. Canevari, D. M. Carraro, J. M.

Cerutti, M. L. C. Corrêa, R. F. R. Corrêa, M. C. R. Costa, C. Curcio, P. O. M. Hokama, A. J. S. Ferreira, G. K. Furuzawa, T. Gushiken, P. L. Ho, E. Kimura, J. E. Krieger, L. C. C. Leite, P. Majumder, M. Marins, E. R. Marques, A. S. A. Melo, M. B. de Melo, C. A. Mestriner, E. C. Miracca, D. C. Miranda, A. L. T. O Nascimento, F. G. Nóbrega, E. P. B. Ojopi, J. R. C. Pandolfi, L. G. Pessoa, A. C. Prevedel, P. Rahal, C. A. Rainho, E. M. R. Reis, M. L. Ribeiro, N. da Rós, R. G. de Sá, M. M. Sales, S. C. Sant'anna, M. L. dos Santos, A. M. da Silva, N. P. da Silva, W. A. Silva Jr., R. A. da Silveira, J. F. Sousa, D. Stecconi, F. Tsukumo, V. Valente, F. Soares, E. S. Moreira, D. N. Nunes, R. G. Correa, H. Zalcberg, A. F. Carvalho, L. F. L. Reis, R. R. Brentani, A. J. G. Simpson, and S. J. de Souza. The contribution of 700,000 ORF sequence tags to the definition of the human transcriptome. *PNAS*, 28(21):12103–12108, October 2001.

[23] Cattle EST Project - The W. M. Keck Center for Comparative and Functional Genomics, University of Illinois at Urbana-Champaign, January 2005. http://titan.biotec.uiuc.edu/cattle/cattle_project.htm.

[24] Celera Genomics, July 2004. http://www.celera.com.

[25] K. M. Chao, W. R. Pearson, and W. Miller. Aligning two sequences within a specified diagonal band. *Comput. Appl. Biosci.*, 8:481–487, 1992.

[26] H. Chou and M. H. Holmes. DNA sequence quality trimming and vector removal. *Bioinformatics*, 17:1093–1104, 2001.

[27] CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico, July 2004. http://www.cnpq.br.

[28] The Gene Ontology Consortium. Creating the Gene Ontology Resource: Design and Implementation. *Genome Research*, 11(8):1425–1433, 2001.

[29] dbEST – The International Expressed Sequence Tags Database, July 2004. http://www.ncbi.nlm.nih.gov/dbEST.

[30] S. J. de Souza, A. A. Camargo, M. R. Briones, F. F. Costa, M. A. Nagai, S. Verjovski-Almeida, M. A. Zago, L. E. Andrade, H. Carrer, H. F.El-Dorry, E. M. Espreafico, A. Habr-Gama, D. Giannella-Neto, G. H. Goldman, A. Gruber, C. Hackel, E. T. Kimura, R. M. Maciel, S. K. Marie, E. A. Martins, M. P. Nobrega, M. L. Paco-Larson, M. I. Pardini, G. G. Pereira, J. B. Pesquero, V. Rodrigues, S. R. Rogatto, I. D. da Silva, M. C. Sogayar, M. F. Sonati, E. H. Tajara, S. R. Valentini, M. Acencio, F. L. Alberto, M. E. Amaral, I. Aneas, M. H. Bengtson, D. M. Carraro

DM, A. F. Carvalho, L. H. Carvalho, J. M. Cerutti, M. L. Correa, M. C. Costa, C. Curcio, T. Gushiken, P. L. Ho, E. Kimura, L. C. Leite, G. Maia, P. Majumder, M. Marins, A. Matsukuma, A. S. Melo, C. A. Mestriner, E. C. Miracca, D. C. Miranda, A. N. Nascimento, F. G. Nobrega, E. P. Ojopi, J. R. Pandolfi, L. G.Pessoa, P. Rahal, C. A. Rainho, N. da Ros, R. G. de Sa, M. M. Sales, N. P. da Silva, T. C. Silva, W. da Silva Jr, D. F. Simao, J. F. Sousa, D. Stecconi, F. Tsukumo, V. Valente, H. Zalcbeg, R. R. Brentani, F. L. Reis, E. Dias-Neto, and A. J. Simpson. Identification of human chromosome 22 transcribed sequences with ORF expressed sequence tags. *PNAS*, 97(23):12690–12693, November 2000.

[31] A. T. R. de Vasconcelos, D. F. de Almeida, M. Hungria, C. T. Guimarães, R. V. Antônio, F. C. Almeida, L. G. P. de Almeida, R. de Almeida, J. A. Alves-Gomes, E. M. Andrade, J. Araripe, M. F. F. de Araújo, S. Astolfi-Filho, V. Azevedo, A. J. Baptista, L. ArturMendesBataus, J. S. Batista, A. Beló, C. den Berg, M. Bogo, S. Bonatto, J. Bordignon, M. M. Brigido, C. A. Brito, M. Brocchi, H. A. Burity, A. A. Camargo, D. D. P. Cardoso, N. P. Carneiro, D. M. Carraro, C. M. B. Carvalho, J. C. M. Cascardo, B. S. Cavada, L. M. O. Chueire, T. B. Creczynski-Pasa, N. C. da Cunha-Junior, N. Fagundes, C. L. Falcão, F. Fantinatti, I. P. Farias, M. S. S. Felipe, L. P. Ferrari, J. A. Ferro, M. I. T. Ferro, G. R. Franco, N. S. A. de Freitas, L. R. Furlan, R. T. Gazzinelli, E. A. Gomes, P. R. Gonçalves, T. B. Grangeiro, D. Grattapaglia, E. C. Grisard, E. S. Hanna, S. N. Jardim, J. Laurino, L. C. T. Leoi, L. F. A. Lima, M. F. Loureiro, M. C. C. P. de Lyra, H. M. F. Madeira, G. P. Manfio, A. Q. Maranhão, W. S. Martins, S. M. Z. di Mauro, S. R. B. de Medeiros, R. V. Meissner, M. A. M. Moreira, F. F. do Nascimento, M. F. Nicolás, J. G. Oliveira, S. C. Oliveira, R. F. C. Paixão, J. A. Parente, F. de O. Pedrosa, S. D. J. Pena, J. O. Pereira, M. Pereira, L. S. R. C. Pinto, L. S. Pinto, J. I. R. Porto, D. P. Potrich, C. E. Ramalho-Neto, A. M. M. Reis, L. U. Rigo, E. Rondinelli, E. B. P. do Santos, F. R. Santos, M. P. C. Schneider, H. N. Seuanez, A. M. R. Silva, A. L. C. da Silva, D. W. Silva, R. Silva, I. C. Simões, D. Simon, C. M. A. Soares, R. de B. A. Soares, E. M. Souza, K. R. L. de Souza, R. C. Souza, M. B. R. Steffens, M. Steindel, S. R. Teixeira, T. Urmenyi, A. Vettore, R. Wassem, Arnaldo Zaha, and A. J. G. Simpson. The complete genome sequence of *Chromobacterium violaceum* reveals remarkable and exploitable bacterial adaptability. *PNAS*, 100(20):11660–11665, September 2003.

[32] A. L. Delcher, D. Harmon, S. Kasif, O. White, and S. Salzberg. Improved microbial gene identification with GLIMMER. *Nucleic Acids Research*, 27(23):4636–4641, 1999.

[33] EGAD – TIGR Expressed Gene Anatomy Database, March 2004. http://www.tigr.org/tdb/egad/egad.html.

[34] Entrez Genome – Whole Genomes Page, September 2004. http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Genome.

[35] W. J. Ewens and G. R. Grant. *Statistical methods in bioinformatics*. Springer, 2001.

[36] FAPESP – Fundação de Amparo à Pesquisa do Estado de São Paulo, May 2004. http://www.fapesp.br.

[37] FORESTs: Eucalyptus Genome Sequencing Consortium, July 2004. https://forests.esalq.usp.br/.

[38] Agronomical & Environmental Genomes, July 2004. http://watson.fapesp.br/AEG/agro.htm.

[39] GenBank, March 2004. http://www.ncbi.nlm.nih.gov/Genbank.

[40] The DDBJ/EMBL/GenBank Feature Table: Definition, March 2004. http://www.ncbi.nlm.nih.gov/projects/collab/FT/index.html.

[41] Genolyptus, July 2004. http://www.lge.ibi.unicamp.br/eucalyptus/.

[42] Genome Network of the State of Minas Gerais, May 2004. http://www.cpqrr.fiocruz.br/genoma/.

[43] Genopar - genoma do paraná, May 2004. http://www.genopar.org/.

[44] P. Green. Phrap Homepage: phred, phrap, consed, swat, cross_match and RepeatMasker Documentation, March 2004. http://www.phrap.org.

[45] D. Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

[46] A. D. Hershey and M. Chase. Independent functions of viral protein and nucleic acid in growth of bacteriophage. *J. Gen. Physiol.*, 36:39–56, 1952.

[47] W. Hide, J. Burke, and D. Davison. Biological evaluation of d^2, an algorithm for high-performance sequence comparison. *J. Comput. Biol.*, 1(3):199–215, 1994.

[48] D. S. Hirschberg. A linear space algorithm for computing maximal common subsequences. *Commun. Assoc. Comput. Mach.*, 18:341–343, 1975.

[49] Honey Bee Brain EST Project, January 2005. http://titan.biotec.uiuc.edu/bee/honeybee_project.htm.

[50] P. T. Hraber and J. W. Weller. On the species of origin: diagnosing the source of symbiotic transcripts. *Genome Biology*, 2001.

[51] X. Huang. A contig assembly program based on sensitive detection of fragments overlap. *Genomics*, 14:18–25, 1992.

[52] X. Huang. On global sequence alignment. *Comput. Appl. Biosci.*, 10:227–235, 1994.

[53] X. Huang. An improved sequence assembly program. *Genomics*, 33:21–31, 1996.

[54] X. Huang and A. Madan. CAP3: a DNA sequence assembly program. *Genome Research*, 9:868–877, 1999.

[55] C. Iseli, C. V. Jongeneel, and P. Bucher. ESTScan: a program for detecting, evaluating and reconstructing potential coding regions in EST sequences. In *Proceedings of International Conference on Intelligent Systems for Molecular Biology*, pages 138–148, 2000.

[56] A. Kalyanaraman, S. Aluru, S. Kothari, and V. Brendel. Efficient clustering of large EST data sets on parallel computers. *Nucleic Acids Research*, 31(11):2963–2974, 2003.

[57] S. Karlin. Detecting anomalous gene clusters and pathogenicity islands in diverse bacterial genomes. *Trends in Microbiology*, 9:335–343, 2001.

[58] J. D. Kececioglu and E. W. Myers. Combinatorial algorithms for DNA sequence assembly. *Algorithmica*, 13:7–51, 1995.

[59] C. G. Kumar, R. LeDuc, G. Gong, L. Roinishivili, H. A. Lewin, and L. Liu. ESTIMA, a tool for EST management in a multi-project environment. *BMC Bioinformatics*, 5(176), 2004.

[60] C. B. Lawrence, N. W. Parrott, T. C. Flood, L. Gu, L. Zhang, M. Jain, S. Larson, and E. W. Myers. The genome reconstruction manager: A software environment for supporting high-troughput DNA sequencing. *Genomics*, 23:192–201, 1994.

[61] LGE - *Crinipellis perniciosa* - Projeto Vassoura de Bruxa, September 2004. http://www.lge.ibi.unicamp.br/vassoura/.

[62] F. Liang, I. Holt, G. Pertea, S. Karamycheva, S. L. Salzberg, and J. Quackenbush. An optimized protocol for analysis of EST sequences. *Nucleic Acids Research*, 28(18):3657–3665, 2000.

[63] L. A. Lyons, T. F. Laughlin, M. A. Copeland, J.E. Womack, and S. J. O'Brien. Comparative anchor tagged sequences (CATS) for integrative mapping of mammalian genomes. *Nat Genet*, 15:47–56, 1997.

[64] R. Z. Ma, M. J. T. van Eijk, J. E. Beever, G. Guérin, C. L. Mummery, and H. A. Lewin. Comparative analysis of 82 expressed sequence tags from a cattle ovary cDNA library. *Mammalian Genome*, 9:545–549, 1998.

[65] W. Makalowski, J. Zhang, and M. S. Boguski. Comparative analysis of 1196 orthologous mouse and human full-length mRNA and protein sequences. *Genome Research*, 6:846–857, 1996.

[66] U. Manber. *Introduction to Algorithms*. Addison-Wesley, 1989.

[67] A. M. Maxam and W. Gilbert. A new method for sequecing DNA. *Proceedings of the National Academy Science, USA*, 74:560–564, 1977.

[68] A. D. McLachlan, R. Staden, and D. R. Boswell. A method for measuring the non-random bias of codon usage table. *Nucleic Acids Research*, 12(24):9567–9575, 1984.

[69] Ministério da Ciência e Tecnologia, July 2004. http://www.mct.gov.br.

[70] Gregor Mendel. Experiments in Plant Hybridization. Mendel's Paper in English.

[71] R. T. Miller, A. G. Christoffels, C. Gopalakrishnan, J. Burke, A. A. Ptitsyn, T. R. Broveak, and W. A. Hide. A Comprehensive Approach to Clustering of Expressed Human Gene Sequence: The Sequence Tag Alignment and Consensus Knowledge Base. *Genome Research*, pages 1143–1155, 1999.

[72] E.W. Myers and W. Miller. Optimal alignments in linear space. *Comput. Applic. Biosci.*, 4:11–17, 1988.

[73] NCBI - National Center for Biotechnology Information, May 2004. http://www.ncbi.nlm.nih.gov/.

[74] NCBI Taxonomy Homepage – The Genetic Codes, July 2004. http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi?mode=c.

[75] A. C. M. Paquola, M. Y. Nishyiama Jr, E. M. Reis, A. M. da Silva, and S. Verjovski-Almeida. ESTWeb: bioinformatics for EST sequencing projects. *Bioinformatics*, 19(12):1587–1588, 2003. Applications Note.

[76] Projeto Genoma Pb, May 2004. https://www.biomol.unb.br/Pb/.

[77] H. Peltola, H. Soderlund, and E. Ukkonen. SEQAID: A DNA sequence assembly program based on a mathematical model. *Nucleic Acids Research*, 12:307–321, 1984.

[78] G. Pertea, X. Huang, F. Liang, V. Antonescu, R. Sultana, S. Karamycheva, Y. Lee, J. White, F. Cheung, B. Parvizi, J. Tsai, and J. Quackenbush. TIGR Gene Indices clustering tools (TGICL): a software system for fast clustering of large EST datasets. *Bioinformatics*, 19(5):651–652, 2003.

[79] J. P. Piazza and J. C. Setubal. New ways for automatic detection of contaminants in EST projects. In S. Lifschitz, editor, *Proceedings of Workshop of Bioinformatics (WOB'2003)*, Macaé - RJ, Brazil, December 2003.

[80] J. P. Piazza and J. C. Setubal. EST contaminant detection by combination of multiple classifiers. January 2004.

[81] Progene - Programa Genoma Nordeste, May 2004. http://www.progene.ufpe.br/index.jsp.

[82] Rede Sul de Análise de Genomas e Biologia Estrutural - Programas de Investigação de Genomas Sul, May 2004. http://www.sct.rs.gov.br/index.htm.

[83] J. Quackenbush, F. Liang, I. Holt, G. Pertea, and J. Upton. The TIGR Gene Indices: reconstruction and representation of expressed gene sequences. *Nucleic Acids Research*, 28(1):141–145, 2000.

[84] Rede da Amazônia Legal de Pesquisas Genômicas – REALGENE, July 2004. https://www.biomol.unb.br/GR/body.html.

[85] RIOGENE - Virtual Institute of Genomic Research, May 2004. http://www.riogene.lncc.br/.

[86] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain termination inhibitors. *Proceedings of the National Academy Science, USA*, 74:5463–5467, 1977.

[87] T. Scheetz and T. Casavant. Informatics for Efficient EST-Based Gene Discovery in Normalized and Subtracted cDNA Libraries. Technical Report TR_CLCG_030131, Departaments of Electrical and Computer Engineering, Pediatrics, and Physiology, University of Iowa, 1995.

[88] T. E. Scheetz, N. Trivedi, C. A. Roberts, T. Kucaba, B. Berger, N. L. Robinson, C. L. Birkett, A. J. Gavin, B. O'Leary, T. A. Braun, M. F. Bonaldo, H. P. Robinson, V. C. Sheffield, M. B. Soares, and T. L. Casavant. ESTprep: preprocessing cDNA sequence. *Bioinformatics*, 19(11):1318–1324, November 2003.

[89] *Schistosoma mansoni* EST Genome Project, July 2004. http://verjo18.iq.usp.br/schisto.

[90] G. Schuler, M. Boguski, E. Stewart, L. Stein, G. Gyapay, K. rice, R. White, P. Rodriguez-Tome, A. Aggarwal, and E. Bajorek et al. A gene map of the human genome. *Science*, 274:540–546, 1996.

[91] A. Shoshan, V. Grebinskiy, A. Magen, A. Scolnicov, E. Fink, D. Lehavi, and A. Wasserman. Designing oligo libraries taking alternative splicing into account. In M. L. Bittner, Y. Chen, A. N. Dorsel, and E. R. Dougherty, editors, *Proceedings of SPIE: Microarrays: Optical Technologies and Informatics*, volume 4266, pages 86–95, 2001.

[92] A. J. G. Simpson, F.C. Reinach, P. Arruda, F. A. Abreu, M. Acencio, R. Alvarenga, L. M. C. Alves, J. E. Araya, G. S. Baia, C. S. Baptista, M. H. Barros, E. D. Bonaccorsi, S. Bordin, J. M. Bove, M. R. S. Briones, M. R. P. Bueno, A. A. Camargo, L. E. A. Camargo, D. M. Carraro, H. Carrer, N. B. Colauto, C. Colombo, F. F. Costa, M. C. R. Costa, C. M. Costa-Neto, L. L. Coutinho, M. Cristofani, E. Dias-Neto, C. Docena, H. El-Dorry, A. P. Facincani, A. J. S. Ferreira, V. C. A. Ferreira, J. A. Ferro, J. S. Fraga, S. C. França, M. C. Franco, M. Frohme, L. R. Furlan, M. Garnier, G. H. Goldman, M. H. S. Goldman, S. L. Gomes, A. Gruber, P. L. Ho, J. D. Hoheisel, M. L. Junqueira, E. L. Kemper, J. P. Kitajima, J. E. Krieger, E. E. Kuramae, F. Laigret, M. R. Lambais, L. C. C. Leite, E. G. M. Lemos, M. V. F. Lemos, S. A. Lopes, C. R. Lopes, J. A. Machado, M. A. Machado, A. M. B. N. Madeira, H. M. F. Madeira, C. L. Marino, M. V. Marques, E. A. L. Martins, E. M. F. Martins, A. Y. Matsukuma, C. F. M. Menck, E. C. Miracca, C. Y. Miyaki, C. B. Monteiro-Vitorello, D. H. Moon, M. A. Nagai, A. L. T. O. Nascimento, L. E. S. Netto, A. Nhani, F. G. Nobrega, L. R. Nunes, M. A. Oliveira, M. C. De Oliveira, R. C. De Oliveira, D. A. Palmieri, A. Paris, B. R. Peixoto, G. A. G. Pereira, H. A. Pereira, J. B. Pesquero, R. B. Quaggio, P. G. Roberto, V. Rodrigues, A. J. De M. Rosa, V. E. De Rosa, R. G. De Sá, R. V. Santelli, H. E. Sawasaki, A. C. R. Da Silva, A. M. Da Silva, F. R. Da Silva, W. A. Silva, J. F. Da Silveira, M. L. Z. Silvestri, W. J. Siqueira, A. A. De Souza, A. P. De Souza, M. F. Terenzi, D. Truffi, S. M. Tsai, M. H. Tsuhako, H. Vallada, M. A. Van Sluys, S. Verjovski-Almeida, A. L. Vettore, M. A. Zago, M. Zatz, J. Meidanis, and J. C. Setubal. The Genome Sequence of the Plant Pathogen *Xylella fastidiosa*. *Nature*, 406(6792):151–159, July 2000.

[93] S. Smith, W. Welch, A. Jakimciu, T. Dahlberg, E. Preston, and D. Van Dyke. High throughput DNA sequencing using an automated electrophoresis analysis system and a novel sequence assembly program. *Biotechniques*, 14:1014–1018, 1993.

[94] T. F. Smith and M. S. Waterman. Comparison of biosequences. *Adv. Appl. Math.*, 2:482–489, 1981.

[95] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, 1981.

[96] M. B. Soares, M. F. Bonaldo, P. Jelene, L. Su, L. Lawton, and A. Efstratiadis. Construction and characterization of a normalized cDNA library. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 91, pages 9228–9232, 1994.

[97] Songbird Neurogenomics Initiative, January 2005. http://titan.biotec.uiuc.edu/songbird/.

[98] R. Sorek and H. M. Safer. A novel algorithm for computational identification of contaminated EST libraries. *Nucleic Acids Research*, 31(3):1067–1074, 2003.

[99] R. Staden. A new computer method for the storage and manipulation of DNA gel reading data. *Nucleic Acids Research*, 8:3673–3694, 1980.

[100] The Sugar Cane EST Genome Project, June 2006. http://sucest.lbi.ic.unicamp.br/en/.

[101] G. G. Sutton, O. White, M. D. Adams, and A. R. Kerlavage. TIGR assembler: A new tool for assembling large shotgun sequencing projects. *Genome Sci. Technol.*, 1:9–19, 1995.

[102] R. E. Tarjan. Efficiency of a good but not linear set union algorithm. *J. ACM*, 22:215–225, 1975.

[103] G. P. Telles, M. D.V. Braga, Z. Dias, L. T. Li, J. A. A. Quitzau, F. R. da Silva, and J. Meidanis. Bioinformatics of the Sugarcane EST Project. *Genetics and Molecular Biology*, 24(1-4):9–15, December 2001.

[104] G. P. Telles and F. R. da Silva. Trimming and clustering sugarcane ESTs. *Genetics and Molecular Biology*, 24(1-4):17–23, December 2001.

[105] The Human Cancer Genome Project, September 2002. http://www.ludwig.org.br/ORESTES.

[106] *Trypanosoma cruzi*, October 2004. http://www.dbbm.fiocruz.br/TcruziDB/.

[107] D. C. Torney, C. Burkes, D. Davidson, and K. M. Sirkin. *Computation of d2: A mesure of sequence dissimilarity, computers and DNA*, volume 2 of *SFI studies in the sciences of complexity*. Addison-Wesley, New York, NY, g. bell and t. marr edition, 1990.

[108] TraceTuner. http://www.paracel.com/sas/tt.htm.

[109] N. Trivedi, J. Bischof, S. Davis, K. Pedretti, T. E. Scheetz, T. A. Braun, C. A. Roberts, N. L. Robinson, V. C. Sheffield, M. B. Soares, and T. L. Casavant. Parallel Creation of Non-redundant Gene Indices from Partial mRNA Transcripts. *Future Generation Computer Systems*, 18(6):863–870, 2002.

[110] M. van Erp, L. Vuurpijl, and L. Schomaker. An Overview and Comparison of Voting Methods for Pattern Recognition. In *Proceedings of 8th International Workshop on Frontiers in Handwriting Recognition*, pages 195–200, Ontario - Canada, August 2002.

[111] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. G. Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. Di Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R.-R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K Naik, V. A. Narayan, B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Y. Wang, A. Wang, X. Wang, J. Wang, M.-H. Wei, R. Wides, C. Xiao, C. Yan, A. Yao, J. Ye, M. Zhan, W. Zhang, H. Zhang, Q. Zhao, L. Zheng, F. Zhong, W. Zhong, S. C. Zhu, S. Zhao, D. Gilbert, S. Baumhueter, G. Spier, C. Carter, A. Cravchik, T. Woodage, F. Ali, H. An, A. Awe, D. Baldwin, H. Baden, M. Barnstead, I. Barrow, K. Beeson, D. Busam, A. Carver, A. Center, M. L. Cheng, L. Curry, S. Danaher, L. Davenport, R. Desilets, S. Dietz, K. Dodson, L. Doup, S. Ferriera, N. Garg, A. Gluecksmann, B. Hart, J. Haynes, C. Haynes, C. Heiner, S. Hladun, D. Hostin, J. Houck, T. Howland, C. Ibegwam, J. Johnson, F. Kalush, L. Kline, S. Koduru, A. Love, F. Mann,

D. May, S. McCawley, T. McIntosh, I. McMullen, M. Moy, L. Moy, B. Murphy, K. Nelson, C. Pfannkoch, E. Pratts, V. Puri, H. Qureshi, M. Reardon, R. Rodriguez, Y.-H. Rogers, D. Romblad, B. Ruhfel, R. Scott, C. Sitter, M. Smallwood, E. Stewart, R. Strong, E. Suh, R. Thomas, N. N. Tint, S. Tse, C. Vech, G. Wang, J. Wetter, S. Williams, M. Williams, S. Windsor, E. Winn-Deen, K. Wolfe, J. Zaveri, K. Zaveri, J. F. Abril, R. Guigó, M. J. Campbell, K. V. Sjolander, B. Karlak, A. Kejariwal, H. Mi, B. Lazareva, T. Hatton, A. Narechania, K. Diemer, A. Muruganujan, N. Guo, S. Sato, V. Bafna, S. Istrail, R. Lippert, R. Schwartz, B. Walenz, S. Yooseph, D. Allen, A. Basu, J. Baxendale, L. Blick, M. Caminha, J. Carnes-Stine, P. Caulk, Y.-H. Chiang, M. Coyne, C. Dahlke, A. D. Mays, M. Dombroski, M. Donnelly, D. Ely, S. Esparham, C. Fosler, H. Gire, S. Glanowski, K. Glasser, A. Glodek, M. Gorokhov, K. Graham, B. Gropman, M. Harris, J. Heil, S. Henderson, J. Hoover, D. Jennings, C. Jordan, J. Jordan, J. Kasha, L. Kagan, C. Kraft, A. Levitsky, M. Lewis, X. Liu, J. Lopez, D. Ma, W. Majoros, J. McDaniel, S. Murphy, M. Newman, T. Nguyen, N. Nguyen, M. Nodell, S. Pan, J. Peck, M. Peterson, W. Rowe, R. Sanders, J. Scott, M. Simpson, T. Smith, A. Sprague, T. Stockwell, R. Turner, E. Venter, M. Wang, M. Wen, D. Wu, M. Wu, A. Xia, A. Zandieh, and X. Zhu. The Sequence of the Human Genome. *Science*, 291(5507):1304–1351, 2001.

[112] A. L. Vettore, F. R. da Silva, E. L. Kemper, and P. Arruda. The libraries that made SUCEST. *Genetics and Molecular Biology*, 406:151–157, 2001.

[113] A. L. Vettore, F. R. da Silva, E. L. Kemper, G. M. Souza, A. M. da Silva, M. I. T. Ferro, F. Henrique-Silva, A. Giglioti, M. V. F. Lemos, L. L. Coutinho, M. P. Nobrega, H. Carrer, S. C. Fran, M. Bacci Jr., M. H. S. Goldman, S. L. Gomes, L. R. Nunes, L. E. A. Camargo, W. J. Siqueira, M. A. V. Sluys, O. H. Thiemann, E. E. Kuramae, R. V. Santelli, C. L. Marino, M. L. P. N. Targon, J. A. Ferro, H. C. S. Silveira, D. C. Marini, E. G. M. Lemos, C. B. Monteiro-Vitorello, J. H. M. Tambor, D. M. Carraro, P. G. Roberto, V. G. Martins, G. H. Goldman, R. C. de Oliveira, D. Truffi, C. A. Colombo, M. Rossi, P. G. de Araujo, S. A. Sculaccio, A. Angella, M. M. A. Lima, V. E. de Rosa Jr., F. Siviero, V. E. Coscrato, M. A. Machado, L. Grivet, S. M. Z. Di Mauro, F. G. Nobrega, C. F.M.Menck, M. D. V. Braga, G. P. Telles, F. A. A. Cara, G. Pedrosa, J. Meidanis, and P. Arruda. Analysis and Functional Annotation of an Expressed Sequence Tag Collection for the Tropical Crop Sugarcane. *Genome Research*, 13:2725–2735, 2003. Submitted: 12/May/2003. Acepted: 08/September/2003.

[114] J. D. Watson and F. H. C. Crick. Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, 171:737–738, 1953.

[115] M. C. Wendl, S. Dear, D. Hodgson, and L. Hillier. Automated Sequence Preprocessing in a Large-Scale Sequencing Environment. *Genome Research*, 8:975–984, 1998.

[116] O. White, T. Dunning, G. Sutton, M. Adams, J. C. Venter, and C. Fields. A quality control algorithm for DNA sequencing projects. *Nucleic Acids Research*, 21:3829–3838, 1993.

[117] S. R. Woodward, N. J. Weyand, and M. Bunnel. DNA sequences from Cretaceous period bone fragments. *Science*, 266:1229–1232, 1994.

[118] T. J. Wu, J. P. Burke, and D. B. Davidson. A measure of DNA sequence dissimilarity based on mahalanobis distance between frequencies of words. *Biometrics*, 53:1431–1439, 1997.

[119] *Xanthomonas axonopodis pv. citri* and *Xanthomonas campestris pv. campestris* Genomes Project, September 2004. http://cancer.lbi.ic.unicamp.br/xanthomonas/.

[120] *Xylella fastidiosa* Genome Project, May 2004. http://aeg.lbi.ic.unicamp.br/xf/.

[121] Z. Zhang, S. Schwartz, L. Wagner, and W. Miller. A greedy algorithm for aligning DNA sequences. *Journal of Computational Biology*, 7:203–214, 2000.

[122] H. Zischler, M. Hoss, O. Handt, A. von Haeseler, A. C. van der Kuyl, J. Goudsmit, and S. Paabo. Detecting dinosaur DNA. *Science*, 268:1191–1193, 1995.