Método para melhoria da eficiência na identificação computacional de RNAs não-codificantes

Cristina Teixeira de Oliveira

Método para melhoria da eficiência na identificação computacional de RNAs não-codificantes

Cristina Teixeira de Oliveira

Orientador: ***Prof. Dr. Guilherme Pimentel Telles***

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC/USP como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional.

USP - São Carlos
Fevereiro /2009

Agradecimentos

Gostaria de registrar aqui minha sincera gratidão a todos aqueles que estiveram ao meu lado no desenvolvimento deste trabalho, resultado de muito esforço e dedicação.

Em primeiro lugar gostaria de agradecer a Deus pela oportunidade, saúde, persistência e sabedoria, para mais esta conquista em minha vida.

Ao Prof. Dr. Guilherme Telles, orientador desta dissertação, por todo seu empenho e compreensão, que fizeram com que concluíssemos este trabalho.

Agradeço a todos professores do mestrado e funcionários da Universidade de São Paulo, pelo auxílio e atenção ao longo destes meses.

Aos amigos de São Carlos e colegas de mestrado pelo apoio e amizade demonstrados, em especial a Alê, Camila, Dalcimar, Eduardo, Endo, Jarbas, KLB, Lúcio, Marcella, Matrix, Mel, Otávio, Paula, Taty, Tott, Van e Vasco pela convivência de alegrias, expectativas, trabalhos, estudos e lazer!

Gostaria de agradecer a todos os meus amigos e amigas da Bahia que sempre estiveram presentes me aconselhando e incentivando com carinho e dedicação.

À minha mãe, pelo seu amor incondicional, incentivo e força. Ao meu Pai, que mesmo com sua rápida passagem, transbordou minha vida de amor e carinho, que sinto vivo e intenso até hoje. À minha irmã pelo amor, admiração e respeito. À minha vó, que sempre esteve orando e torcendo por mim.

Ao meu namorado, amigo e companheiro, Carlos, pelo incansável apoio durante o desenvolvimento deste trabalho, por sua paciência e compreensão reveladas, fundamental nesta trajetória.

Por fim, deixo aqui minha sincera gratidão a todas as pessoas que, direta ou indiretamente, contribuíram para a concretização deste trabalho.

Resumo

Até pouco tempo acreditava-se que a maioria das moléculas de RNA estava relacionada à tradução de proteínas. Porém, descobriu-se que outros tipos de moléculas de RNA que não são traduzidas estão presentes em muitos organismos diferentes e afetam uma variedade de processos moleculares, são os chamados RNAs não-codificantes (ncRNAs). Apesar de sua importância funcional, os métodos biológicos e computacionais para a detecção e caracterização de RNAs não-codificantes ainda são imprecisos e incompletos. A identificação de novas espécies de ncRNAs é difícil através de procedimentos experimentais e as técnicas computacionais existentes são lentas. O objetivo deste trabalho foi obter uma ferramenta mais eficiente para a comparação de uma seqüência de RNA não-codificante contra um banco de seqüências. Para isso foi proposto e implementado um modelo para identificação computacional de ncRNAs com apoio dos pacote Viena e Infernal e foram realizados experimentos para avaliá-lo.

Abstract

Until recently it was generally accepted that most RNA molecules were involved in the translation process. However, it was discovered that many types of untranslated RNA molecules are present in many different organisms and they are related to a wide variety of molecular processes. These molecules are called non-coding RNAs (ncRNAs). Despite their functional importance, the biological and computational methods to detect and identify non-coding RNAs are still imprecise and incomplete. The discovery of new ncRNAs species is difficult through experimental procedures and the existing computational techniques are slow. This project aimed at obtaining a more efficient tool that compares a non-coding RNA sequence against a sequence database. In order to achieve this, a computational model for ncRNAs identification using the Vienna and Infernal packages has been proposed and implemented. Experiments were conduced to evaluate the model.

Sumário

Lista de Figuras

Lista de Tabelas

CAPÍTULO

1

Introdução

1.1 Contextualização

A Bioinformática utiliza conhecimentos da área da Computação a fim de processar e solucionar problemas biológicos. Além da dificuldade intríseca dos problemas em bioinformática, o enorme volume de dados envolvidos torna-os ainda mais complicados. Neste contexto, a fim de extrair informações destes dados técnicas computacionais são utilizadas para possibilitar/auxiliar a tentativa de resolução de problemas na área de maneira mais rápida e eficiente.

1.2 Motivação

Até recentemente, pensava-se que a maioria das moléculas de ácidos ribonucleicos (RNAs) estavam relacionadas ao envio da informação genética para a tradução de proteínas, com exceção apenas do RNA transportador (tRNA) e do RNA ribossômico (rRNA), que também desempenham funções relacionadas diretamente à tradução de proteínas. Porém, desde a década de 90 descobriu-se outros tipos de moléculas de RNA, que não são traduzidas e estão presentes em muitos organismos diferentes, afetando uma grande variedade de processos (Liu et al., 2005). Essas moléculas são os chamados RNAs não-codificantes (ncRNAs).

Os RNAs não-codificantes controlam uma gama notável de reações biológicas e processos, como iniciação da tradução, controle da abundância de RNA mensageiro (mRNA), arquitetura do cromossomo, manutenção de células-tronco, desenvolvimento do cérebro e músculos, secreção de insulina, dentre outras (Michalak, 2006).

Por volta de 98% do que é transcrito pelo genoma humano é constituído de ncRNAs e a diferença na complexidade de um organismo pode ocorrer principalmente devido à vasta diferença na quantia de ncRNAs presentes nos organismos eucarióticos e nos organismos mais simples (Mattick, 2001). Apesar da importância funcional dos ncRNAs, a maior parte dos métodos já desenvolvidos têm se preocupado em identificar moléculas que estão relacionadas à codificação de proteínas. Nesse contexto, surge a idéia de que uma classe de genes poderia ter permanecido descartada por estar relacionada à transcrição de ncRNAs.

Apesar de sua importância, a identificação de novas espécies de ncRNAs é difícil através de procedimentos experimentais ou de técnicas computationais tradicionais. Nos ncRNAs existe uma escassez de características como as utilizadas na busca de genes que codificam proteínas. Os genes de ncRNAs são tipicamente curtos, têm padrões variados e são caracterizados mais pela estrutura secundária do que pela seqüência primária (Huttenhofer et al., 2005). Muitos dos ncRNAs conhecidos hoje foram descobertos casualmente em pesquisas de determinados genomas. Outra dificuldade é que a identificação dos ncRNAs se dá pela ausência de proteína traduzida e não pela presença de uma molécula.

1.3 Objetivo

O presente trabalho buscou cooperar com estudos que têm sido feitos na busca de soluções para a identificação de RNAs não-codificantes. A proposta deste trabalho, visou introduzir uma nova idéia junto à ferramenta Infernal que foi criada para detecção de ncRNAs por Eddy (2003), aplicando técnicas inteligentes na identificação de genes de ncRNAs contra um banco de seqüências com uma precisão aceitável e uma velocidade superior às ferramentas hoje disponíveis.

1.4 Organização do Trabalho

No Capítulo 2 são abordados conceitos relativos à Biologia Molecular. Neste Capítulo são descritos alguns de seus conceitos básicos da biologia molecular relacionados com o objetivo deste trabalho de mestrado, incluindo o ácido desoxirribonucléico (DNA), o ácido ribonucléico (RNA) e a síntese de proteínas, que desempenham papéis fundamentais em processos biológicos nos seres vivos.

No Capítulo 3 são apresentados e descritos os RNAs não-codificantes (ncRNAs). Esses RNAs são moléculas que não irão sintetizar proteínas. Também são apresentados algumas das classes conhecidas de ncRNAs e os desafios encontrados na detecção de ncRNAs, que serviram de motivação para esta pesquisa.

No Capítulo 4 são revistos alguns trabalhos que descrevem técnicas computacionais para identificação de ncRNAs. Neste mesmo capítulo também são apresentados repositórios (base de dados) de ncRNAs já identificados.

No Capítulo 5 é apresentado, em detalhes, o modelo proposto para busca de ncRNAS neste trabalho, o qual foi implementado com o apoio do pacote Viena, que consiste de uma biblioteca incluindo programa para a predição da estrutura secundária do RNA (Zuker e Stiegler, 1981). São descritos também os experimentos realizados, bem como seus resultados e desempenhos.

Por fim, no Capítulo 6 são apresentadas as conclusões deste trabalho, enfatizando as suas principais contribuições e apresentando propostas de trabalhos futuros em continuidade ao que foi realizado.

CAPÍTULO

2

Fundamentos da Biologia Molecular

2.1 Considerações Iniciais

O marco inicial da biologia molecular foi em 1953 quando James Watson e Francis Crick revelaram a estrutura em dupla hélice da molécula do ácido desoxirribonucleico, propiciando uma melhor compreensão das funções do DNA como material genético (Lodish et al., 2005).

Neste Capítulo são descritos alguns dos conceitos básicos da biologia molecular relacionados com o objetivo deste trabalho, apresentado no Capítulo 1. A Seção 2.2 apresenta o conceito da biologia molecular, a Seção 2.2.1 descreve o ácido desoxirribonucléico (DNA), uma importante molécula encontrada em quase todos os organismos, que carrega a informação genética e é responsável pela transmissão das características hereditárias de cada espécie. Na Seção seguinte, 2.2.2, descorre-se sobre os ácidos ribonucléicos (RNA). Na Seção 2.2.3 é descrita a síntese de proteína. As proteínas desempenham papéis fundamentais em processos biológicos nos seres vivos. E, finalmente, na Seção 2.3 são apresentadas as considerações finais sobre este Capítulo.

2.2 Conceitos da Biologia Molecular

A Biologia Molecular é uma área de estudo ligada à genética e à bioquímica, cujo foco é em estudar a vida em nível molecular, partindo da relação entre o DNA, o RNA e a síntese de proteínas.

As proteínas, longos polímeros de aminoácidos, constituem, além da água, a maior fração das células. Algumas proteínas têm atividade catalítica e funcionam como enzimas, outras servem como elementos estruturais, receptoras de sinais, ou transportadoras, responsáveis por carregar

determinadas substâncias para dentro ou para fora das células. Os ácidos nucléicos, DNA e RNA, são polímeros de nucleótidos. Eles armazenam e transmitem a informação genética e algumas moléculas de RNA também possuem papéis estruturais e catalíticos (Nelson e Cox, 2004).

2.2.1 DNA

As informações sobre como, quando e onde produzir cada tipo de proteína estão no material genético, em um polímero chamado ácido desoxirribonucléico (Lodish et al., 2005). O ácido desoxirribonucléico ou DNA contém as informações genéticas em todos os organismos vivos, com exceção de alguns vírus (Silva e Andrade, 2001). A maioria dos DNAs de células eucarióticas estão localizadas no núcleo (Lodish et al., 2005).

Cada molécula de DNA é constituída por cadeias de nucleotídeos. O nucleotídeo é formado por três moléculas: um ácido fosfórico, um açúcar e uma base nitrogenada. O açúcar que constitui o nucleotídeo que irá compor o DNA é uma pentose, chamada desoxirribose (Koolman e Roehm, 2005). As bases nitrogenadas podem ser divididas em dois grupos: as bases púricas ou purinas e as bases pirimídicas ou pirimidinas. Dentre esses grupos, nas bases púricas se encontram as bases formadas por dois anéis, a adenina (A) e a guanina (G), e nas bases pirimídicas as bases formadas por um único anel, a citosina (C) e a timina (T). Dessa forma, existem quatro tipos de nucleotídeos a depender da constituição de sua base nitrogenada (Figura 2.1).

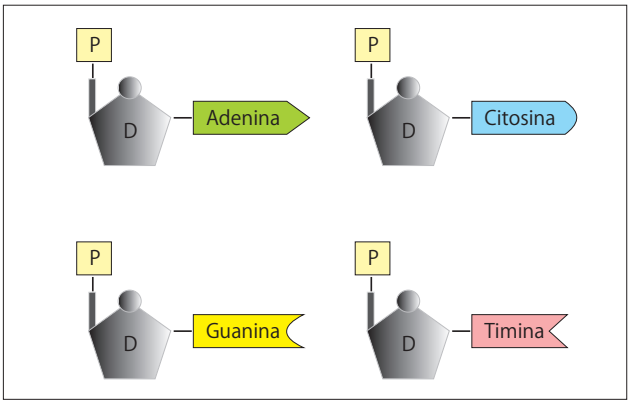

Figura 2.1: Tipos de nucleotídeos que compõem a molécula de DNA. P representa o ácido fosfórico, D a desoxirribose que vem seguida das bases nitrogenadas que podem ser adenina (A), guanina (G), citosina (C) ou timina (T).

Segundo o modelo proposto em 1953 por Watson e Crick, a estrutura do DNA é formada por duas cadeias ou fitas paralelas compostas por vários nucleotídeos, unidas através de pontes de hidrogênio formadas entre as bases nitrogenadas de cada fita, sendo que a base adenina estará pareada com timina (A-T) e citosina com guanina (C-G) (Stryer et al., 2002). Essas bases, A-T e C-G, são chamadas bases complementares. As duas fitas ficam dispostas em espiral em torno de um eixo. Cada fita possui uma extremidade chamada 3' e uma chamada 5' permitindo convencionar uma orientação, em que as duas cadeias ficam em direção opostas (antiparalelas), constituindo uma dupla-hélice (Figura 2.2).

Em alguns momentos o DNA sofre replicação através de um processo conhecido como duplicação semiconservativa, pois cada DNA recém formado possui uma das cadeias da molécula mãe (Lopes, 1998). Para replicar-se, a dupla-fita do DNA abre-se através do rompimento das pontes de hidrogênio e nucleotídeos livres encaixam-se na molécula através de novas pontes de hidrogênio. Os nucleotídeos vão sendo ligados entre si pela enzima DNA polimerase. Este processo irá resultar na formação de duas moléculas de DNA idênticas à original (Figura 2.2).

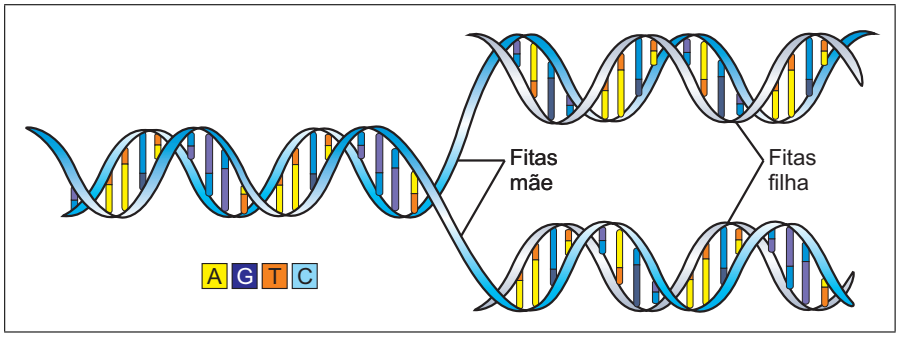

Figura 2.2: À esquerda a representação de um trecho de uma molécula de DNA que evidencia o aspecto de dupla-hélice e à direita as fitas mãe separadas servindo de molde para as filhas, resultando em duas moléculas idênticas à dupla-hélice original (Lodish et al., 2005).

Segundo Koolman e Roehm (2005) em todas as células vivas o DNA desempenha o papel de armazenar a informação genética. Para as informações genéticas armazenadas no DNA se tornarem eficazes, elas devem ser reescritas (transcritas) em RNA. O DNA apenas servirá como um modelo e não é alterado pelo processo de transcrição.

Segmentos passíveis de transcrição do DNA são chamados genes. Os genes são transcritos em RNAs, que realizam funções catalíticas ou estruturais ou fornecem a base para síntese de proteínas. Nesse último caso, o DNA codifica a estrutura primária das proteínas.

A estrutura primária é dada pela seqüência linear da molécula formada ao longo da cadeia, sendo o nível estrutural mais simples. A seqüência secundária consiste no arranjamento espacial resultante do desdobramento da cadeia, como as hélices (Lodish et al., 2005).

Estima-se que o genoma dos mamíferos contenha de 30000 a 40000 genes que, em conjunto, representam menos de 5% de sua molécula de DNA (Koolman e Roehm, 2005).

Duas moléculas são ditas homólogas se descendem de um ancestral comum. Moléculas homólogas podem ser divididas em duas classes: ortólogas e parólogas. As moléculas ortólogas são moléculas relacionadas por especiação, possuindo uma descendência vertical, já as parólogas são moléculas relacionadas por duplicação.

Compreender a homologia entre as moléculas pode revelar sua história evolutiva, bem como informações sobre a sua função (Stryer et al., 2002).

2.2.2 RNA

De forma semelhante ao DNA (Seção 2.2.1), o ácido ribonucléico ou RNA é uma molécula constituída por cadeias de nucleotídeos, ou seja, um polinucleotídeo.

O nucleotídeo que irá compor o RNA é formado por três moléculas: uma de ácido fosfórico, uma de açúcar e uma base nitrogenada. No RNA, o açúcar também é uma pentose: a ribose. Porém, nas bases nitrogenadas verifica-se a presença de uracila substituindo a timina do DNA (Lodish et al., 2005). Logo, no grupo das bases púricas se encontram a adenina (A) e a guanina (G), e no das bases pirimídicas, a citosina (C) e a uracila (U) (Figura 2.3).

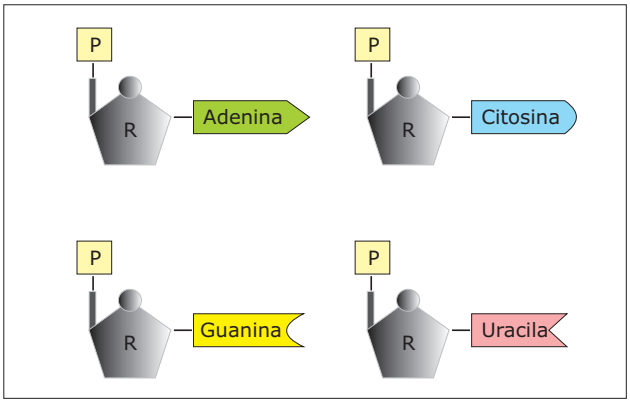

Figura 2.3: Tipos de nucleotídeos que compõem a molécula de RNA. P representa o ácido fosfórico, R a ribose e, em seguida, as bases nitrogenadas que podem ser adenina (A), guanina (G), citosina (C) ou uracila (U).

Diferentemente do DNA, o RNA é formado por uma única fita de nucleotídeos, ou seja, não possui o aspecto dupla-hélice (Koolman e Roehm, 2005). As bases complementares, que no caso dos RNAs são a adenina com uracila (A-U) e a citosina com guanina (C-G), podem unir-se através de pontes de hidrogênio, de tal forma que o RNA dobra-se sobre si mesmo.

Segundo Nelson e Cox (2004) todas moléculas de RNA, com exceção do genoma de alguns vírus, são derivadas das informações armazenadas no DNA.

Existem três tipos de RNAs que estão envolvidos diretamente na síntese de proteínas (Seção 2.2.3): o ribossômico (rRNA), o mensageiro (mRNA) e o transportador (tRNA). O RNA ribossômico é encontrado no nucléolo[1] de células procarióticas e no núcleo de células eucarióticas, onde ocorre sua produção. Esse tipo de RNA, mostrado na Figura 2.4, é o componente central dos ribossomos. Os ribossomos são organelas encontradas no citoplasma que possuem duas subunidades chamadas de subunidades 40S e 60S em células eucarióticas e 30S e 50S em bactérias (Lafontaine e Tollervey, 2001). O RNA mensageiro é encontrado tanto no núcleo, onde ocorre sua síntese, quanto no citoplasma, onde irá participar da tradução de proteínas. Os RNAs transportadores (tRNAs) são encontrados no citoplasma e funcionam, durante a tradução, como ligações entre as proteínas e

[1] Nucléolo é um subcompartimento do núcleo da célula onde a maioria dos rRNA são sintetizados (Lodish et al., 2005).

ácidos nucléicos. Esses RNAs são constituídos por pequenas moléculas de RNA contendo entre 70 e 90 nucleotídeos (Koolman e Roehm, 2005). Na Figura 2.5 pode ser vista a numeração dos nucleotídeos em um tRNA.

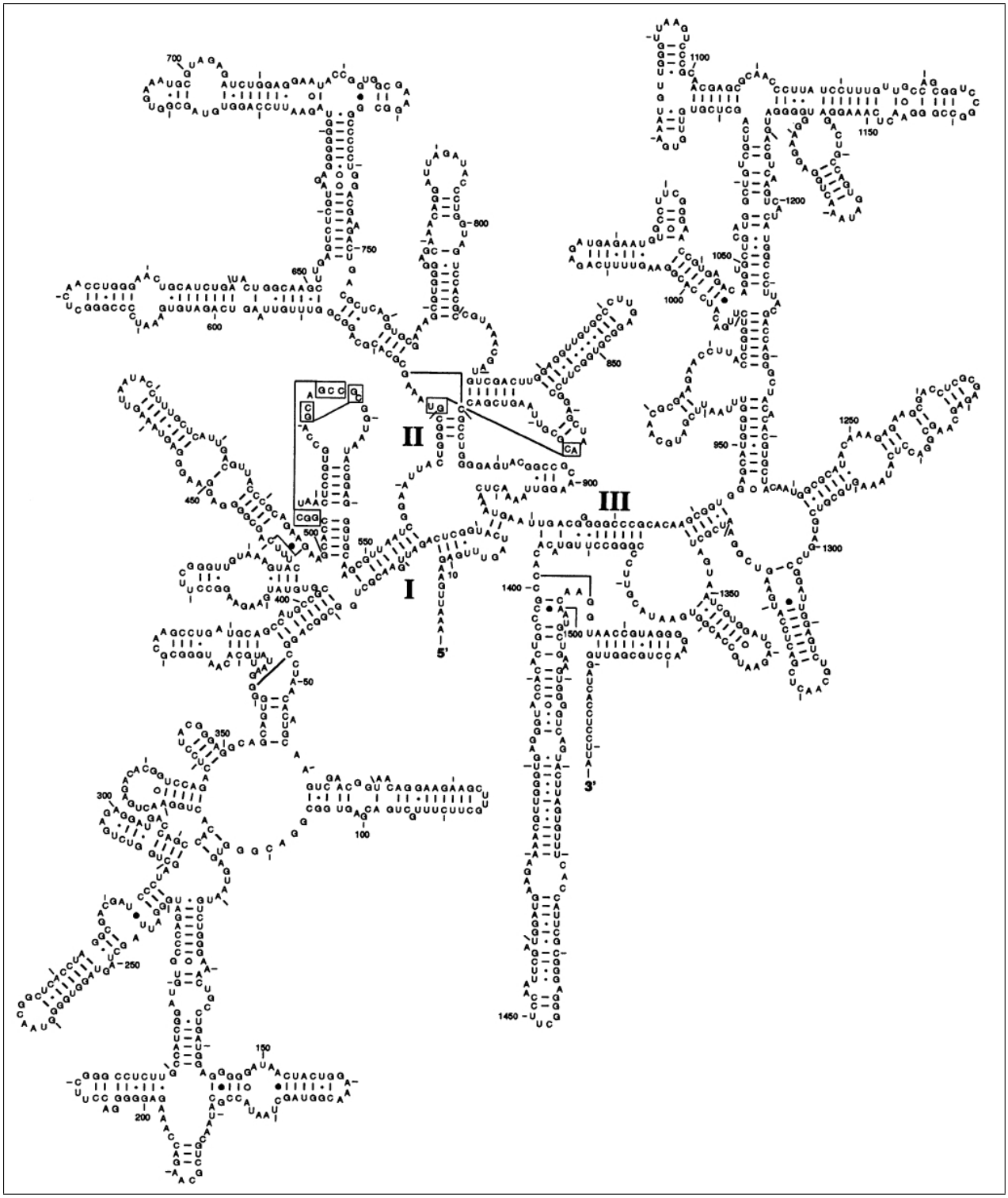

Figura 2.4: Modelo da estrutura de um rRNA 16S da bactéria *Escherichia coli* (Gutell et al., 1994).

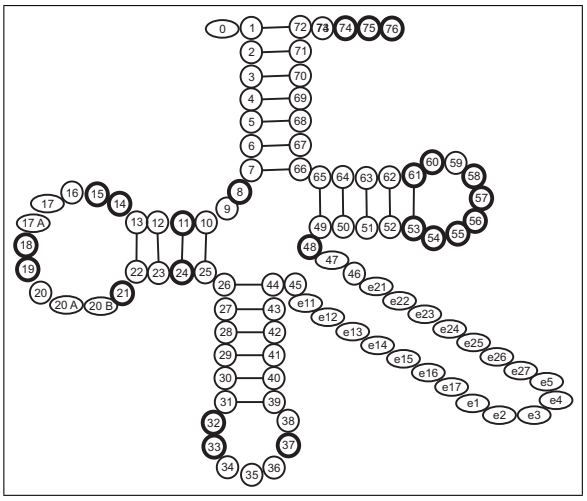

Figura 2.5: Numeração dos nucleotídeos em tRNA. Os círculos representam nucleotídeos que estão sempre presentes, a elipse são nucleotídeos que não estão presentes em todas estruturas, os nucleotídeos no talo variável têm o prefixo "e" e ficam situados entre as posições 45 e 46. Posições em que o nucleotídeo não varia são representadas por um círculo com uma linha mais grossa (Sprinzl et al., 1997).

No Capítulo 3 é apresentado o conceito de RNAs não-codificantes. Esses RNAs, a exemplo de alguns já descritos nesta seção, são moléculas que não serão traduzidas, ou seja, não irão sintetizar proteínas.

2.2.3 Síntese de Proteínas

A expressão gênica consiste na produção de componentes estruturais e funcionais necessários à manutenção da célula. A informação contida em um gene é transcrita pela enzima RNA polimerase em ácido ribonucléico (RNA) e em seguida traduzida em uma molécula que é constituída por aminoácidos, a proteína (Silva e Andrade, 2001). Esses dois processos, que são fundamentais para o funcionamento celular, são descritos com mais detalhes nesta seção.

Proteínas são compostos orgânicos constituídos por aminoácidos unidos através de ligações peptídicas. As proteínas estão envolvidas em todos os processos biológicos dos seres vivos, ou seja, constituem a maquinaria que dá vida, pois estão envolvidas nas funções estruturais, catalisadoras e reguladoras (Liu et al., 2006).

Conforme citado, a célula utiliza dois processos para converter a informação contida no DNA em proteínas, a transcrição e a tradução (Figura 2.6). Na Figura 2.7 é ilustrado o processo de síntese de proteínas. Primeiramente, ocorre a transcrição no núcleo. Um gene capaz de traduzir proteínas é transcrito em moléculas de RNA. A transcrição é feita pela enzima RNA polimerase, que se liga a determinadas seqüências de nucleotídeos do DNA, identificadas pelas regiões promotoras, e a percorre utilizando como molde até encontrar as regiões terminadoras. A transcrição se baseia no pareamento de bases complementares usando uma fita do DNA como molde, ou seja, adenina com

uracila (A → U), timina com adenina (T → A) e citosina com guanina (C ↔ G). Essa molécula de RNA recém sintetizada é chamada de RNA mensageiro (Lodish et al., 2005).

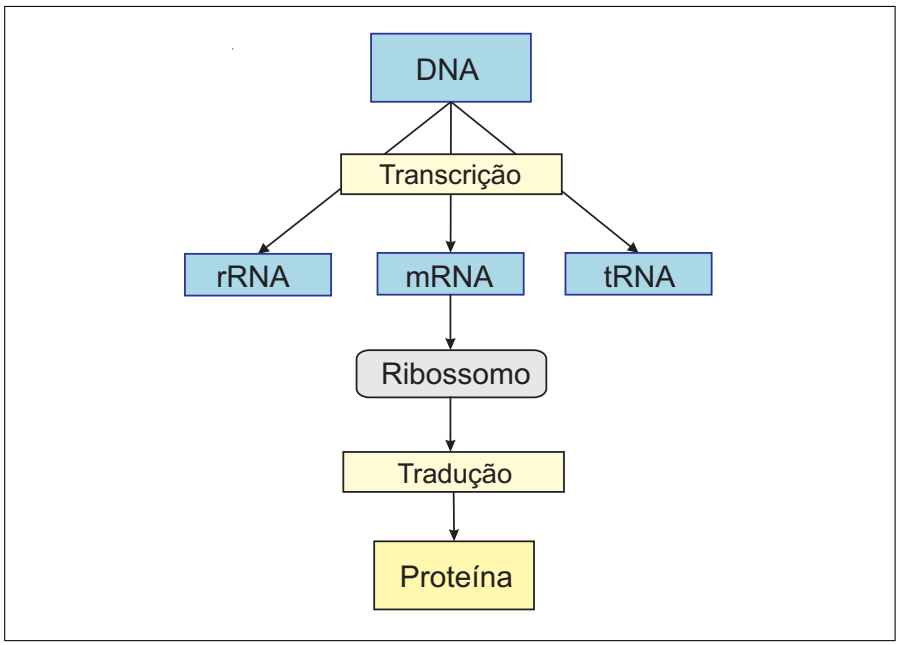

Figura 2.6: Etapas da síntese de proteína.

Em organismos superiores, o mRNA recém-transcrito é um pré-mRNA que irá sofrer algumas modificações antes que se transforme em um mRNA maduro (Silva e Andrade, 2001). Durante esse processo de maturação ocorre o *splicing*, ou seja, a eliminação dos íntrons do pré-mRNA (Figura 2.7), que são seções que não codificam qualquer parte da proteína produzida pelo gene e a junção dos éxons, partes do DNA que contém a informação genética que irá sintetizar proteínas (Lodish et al., 2005; Lundblad, 2007). Existem também os RNAs pequenos do núcleo (Seção 3.3.1) que participam do processo de *splicing*.

Segundo Lodish et al. (2005) todos os organismos possuem maneiras de controlar quando e onde seus genes podem ser transcritos. Por exemplo, quase todas as células do nosso corpo contêm o conjunto completo de genes humanos, mas em cada tipo de célula apenas alguns desses genes estão ativos, ou ligados, e são usados para codificar proteínas. É por isso que, por exemplo, as células hepáticas produzem algumas proteínas que não são produzidas pelas células renais e vice-versa. Além disso, muitas células são capazes de responder a sinais externos ou a alterações nas condições externas, ligando ou desligando genes específicos, dessa forma, ela estará se adaptando às necessidades do momento. Tal controle da atividade gênica depende das proteínas, chamadas fatores de transcrição, que se ligam ao DNA e atuam como interruptores, ativando ou desativando a transcrição de determinados genes.

O mRNA formado através da transcrição se move para o citoplasma. No citoplasma, mais precisamente onde os ribossomos localizam-se, irá ocorrer o segundo processo para a síntese da proteína: a tradução.

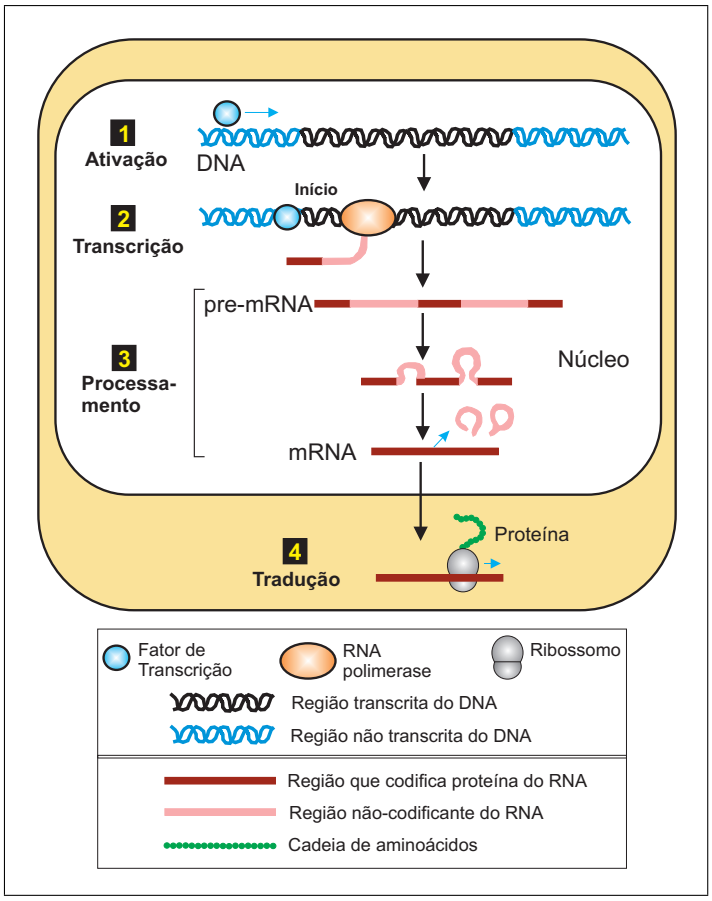

Figura 2.7: Síntese protéica em célula eucariótica: a informação do RNA é convertida em seqüências de aminoácidos que formam as proteínas. No passo 1 fatores de transcrição se ligam às regiões da regulação dos genes que controlam, ativando-os. No passo 2 a RNA polimerase começa a transcrição do gene ativado na região promotora, resultando na formação do pre-mRNA. A transcrição é processada para remover seqüências não-codificantes. E, por fim, no passo 4 o mRNA move-se para o citoplasma e é lido pelos ribossomos. Nesse estágio, a proteína é sintetizada pelo ribossomo que liga os aminoácidos em uma cadeia linear (Lodish et al., 2005).

Em 1966 verificou-se que cada aminoácido é codificado por um grupo de três bases do DNA, denominadas tríplex ou códon. Cada códon corresponde a um único aminoácido, porém um mesmo aminoácido pode ser definido por mais de um códon (Tabela 2.1) (Silva e Andrade, 2001; Lopes, 1998). Existem ainda três códons (UAG, UAA e UGA) que não correspondem a nenhum aminoácido, significando sinais de término da cadeia peptídica (Silva e Andrade, 2001).

Na tradução, primeiramente o mRNA se liga entre as duas subunidades do ribossomo. Cada códon do mRNA é pareado com o anticódon correspondente que está presente em moléculas de RNAs transportadores (Silva e Andrade, 2001). Após o pareamento do segundo tRNA, os aminoácidos são ligados e o primeiro tRNA é liberado. Esse processo é repetido até que apareça um sinal de terminação no mRNA e vai resultar na formação de uma cadeia polipeptídica. O pareamento da relação entre códon e anticódon pode ser visto na Figura 2.8.

Primeira posição	Primeira posição				Terceira posição
	U	C	A	G	
U	UUU Phe	UCU Ser	UAU Tyr	UGU Cys	U
	UUC Phe	UCC Ser	UAC Tyr	UGC Cys	C
	UUA Leu	UCA Ser	UAA Stop	UGA Stop	A
	UUG Leu	UCG Ser	UAG Stop	UGG Trp	G
C	CUU Leu	CCU Pro	CAU His	CGU Arg	U
	CUC Leu	CCC Pro	CAC His	CGC Arg	C
	CUA Leu	CCA Pro	CAA Gln	CGA Arg	A
	CUG Leu	CCG Pro	CAG Gln	CGG Arg	G
A	AUU Ile	ACU Thr	AAU Asn	AGU Ser	U
	AUC Ile	ACC Thr	AAC Asn	AGC Ser	C
	AUA Ile	ACA Thr	AAA Lys	AGA Arg	A
	AUG Met[a]	ACG Thr	AAG Lys	AGG Arg	G
G	GUU Val	GCU Ala	GAU Asp	GGU Gly	U
	GUC Val	GCC Ala	GAC Asp	GGC Gly	C
	GUA Val	GCA Ala	GAA Glu	GGA Gly	A
	GUG Val	GCG Ala	GAG Glu	GGG Gly	G

[a] AUG faz parte do sinal de inicialização.

Tabela 2.1: O código genético padrão (Voet e Voet, 1995).

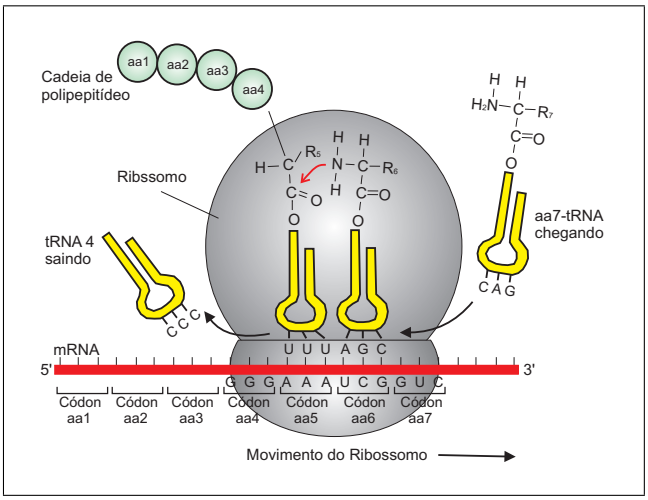

Figura 2.8: Pareamento da relação entre códon e anticódon. O alinhamento dos dois RNAs é antiparalelo. O RNA mensageiro (mRNA) é traduzido em proteína pela ação conjunta do RNA transportador (tRNA) e do ribossomo, que é composto por numerosas proteínas e duas importantes moléculas de RNA (rRNA). Nota-se o pareamento entre os anticódons dos tRNAs e os códons complementares no mRNA. Forma-se uma ligação entre N no aa-tRNA de entrada com C na cadeia de proteína crescente. (Lodish et al., 2005).

2.3 Considerações Finais

Neste capítulo foram abordados alguns dos conceitos relacionados à Biologia Molecular que formaram a base para o desenvolvimento deste trabalho.

As moléculas mais importantes são o DNA, o RNA e as proteínas. As proteínas são construídas a partir de informação armazenada pelos genes no DNA com intermediação de moléculas de RNA. Os processos de transcrição e tradução formam o chamado dogma central da biologia molecular.

Até pouco tempo acreditava-se que a única função do RNA relacionava-se à transcrição. Mais recentemente descobriu-se os RNAs não-codificantes, que são o foco principal deste trabalho e são descritos no próximo capítulo.

CAPÍTULO

3

RNAs Não-Codificantes

3.1 Considerações Iniciais

Neste capítulo são apresentados os RNAs não-codificantes (ncRNAs) que estão diretamente relacionados com o objetivo deste trabalho de mestrado e são de fundamental importância para sua compreensão.

Na Seção 3.2 são apresentados e descritos os RNAs não-codificantes. Esses RNAs são moléculas que não serão traduzidas, ou seja, não irão sintetizar proteínas. Logo em seguida, na seção 3.3, são descritas algumas das classes conhecidas de ncRNAs. Nessa seção são apresentadas as seguintes classes de ncRNA: RNA pequeno do núcleo (Seção 3.3.1), RNA pequeno do nucléolo (Seção 3.3.2), microRNA (Seção 3.3.3) e RNA pequeno do corpo de Cajal (Seção 3.3.4). Na Seção 3.4 seguem as características e desafios na detecção de ncRNAs, que serviram de motivação para esta pesquisa. Por fim, na Seção 3.5 são apresentadas as considerações finais sobre este Capítulo.

3.2 RNAs Não-Codificantes

O RNA não-codificante (ncRNA) é qualquer molécula funcional de RNA que não será traduzida em uma proteína, possuindo funções biológicas diversas. Ou seja, os genes de ncRNAs produzem RNAs funcionais ao invés de codificar proteínas (Eddy, 2001). Os ncRNAs também são conhecidos como pequenos RNAs (sRNA, do inglês *small RNA*), pelo pequeno número de nucleotídeos que os constituem, ou RNAs funcionais (fRNA, do inglês *function RNA*).

A classe de RNA funcional foi predita pela hipótese da adaptação de Francis Crick. Crick previu a existência de uma molécula que provê a relação entre a tripla do código genético e os aminoácidos codificados (Crick, 1958).

Anteriormente, o RNA era uma molécula envolvida unicamente no dogma central da biologia. O RNA detinha três classes que estavam ligadas à síntese de proteínas: o rRNA, o tRNA e todo o resto assumido como mRNA. A fração conhecida de rRNA e tRNA era complexa, não abundante e principalmente instável. Nesse contexto, havia pouca motivação para estudo dessas moléculas de RNAs(Eddy, 2001). Logo, as famílias de ncRNAs conhecidas até os anos 80 eram os tRNAs e rRNAs (Seção 2.2.2). Entretanto, com o passar do tempo, inúmeras descobertas de ncRNAs foram feitas, com as mais diversas funções. Atualmente, o número e a diversidade de genes de RNAs que não codificam proteínas são alvos de inúmeras pesquisas.

Apenas uma fração do genoma é traduzido em proteínas. Por exemplo, no genoma humano apenas 1.4% do DNA é traduzido em proteína e 25% do genoma é predito para ser transcrito mas não traduzido, o que aumenta a possibilidade de descoberta de novos ncRNAs e sugere a existência de um grande número de RNAs não-codificantes ainda não caracterizados (Figura 3.1) (Huttenhofer et al., 2005). O número de ncRNAs foi previamente subestimado, porém a importância e a quantidade de novas classes de ncRNA vem crescendo continuamente nos últimos anos devido a estudos e novas descobertas (Figura 3.2). Muitos desses ncRNAs ainda não foram relacionados à sua função. Algumas das classes conhecidas de ncRNA são descritas na Seção 3.3.

Geralmente, os ncRNAs não possuem uma seqüência conservada, tendo como principal característica a conservação de sua estrutura tridimensional, tornando sua identificação mais difícil. Os mais bem conhecidos ncRNAs possuem uma estrutura tridimensional complexa e têm funções tanto catalizadoras como estruturais (Eddy, 2002).

Desde a descoberta dos ncRNAs muitas perguntas têm sido feitas e muitos estudos têm sido direcionados à procura de respostas. Porém, essas moléculas ainda não são bem conhecidas. A principal causa disso é que maior parte das pesquisas para detecção de genes, durante muito tempo, foram voltadas na direção de RNAs mensageiros e proteínas.

3.3 Classes de ncRNAs

Grande parte dos ncRNAs conhecidos hoje foram identificados experimentalmente. Nesta seção são apresentadas algumas classes já conhecidas de RNAs não-codificantes. Os RNAs transportadores e os RNAs ribossômicos foram descritos na Seção 2.2.2.

3.3.1 RNA Pequeno do Núcleo

Os RNAs pequenos do núcleo, ou snRNAs do inglês *Small Nuclear* RNAs (snRNAs), são uma classe de RNAs não-codificantes encontrados no interior do núcleo das células. O núcleo

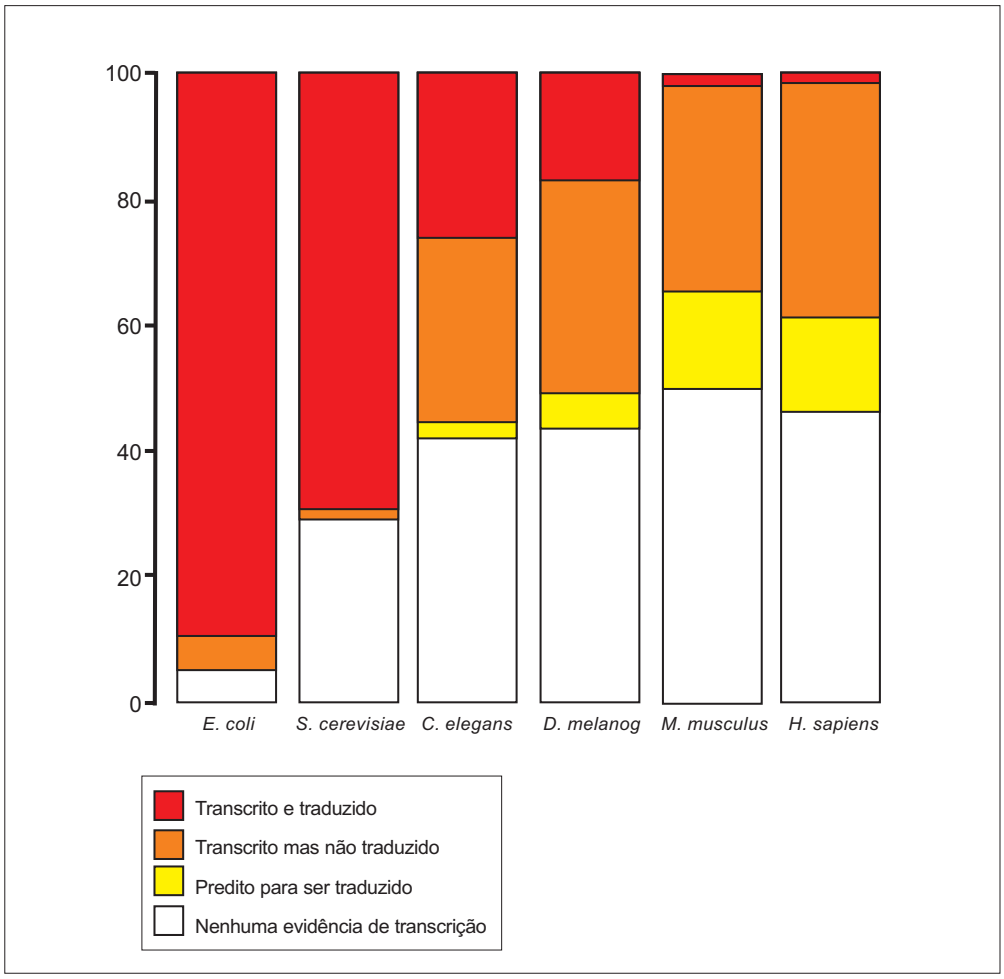

Figura 3.1: Espaço do genoma para descoberta de novos ncRNAs (Huttenhofer et al., 2005).

contém muitos tipos de snRNAs. A estrutura secundária desses RNAs é altamente conservada nos organismos. Alguns deles, conhecidos como U1, U2, U4, U5 e U6, são essenciais para o *splicing* do pre-mRNA (Seção 2.2.3)(Stryer et al., 2002).

Conforme abordado no Capítulo 2, os RNAs mensageiros irão carregar a informação genética do núcleo para a tradução de proteínas no citoplasma. Para isso, o pre-mRNA sofre modificações através da remoção dos íntrons e torna-se um mRNA maduro que será traduzido em proteína (Seção 2.2.3).

Os snRNAs estão associados com proteínas específicas para formar complexos denominados snRNPs (do inglês *small nuclear ribonucleoproteins*). Os snRNPs formam grandes arranjos que constituem os spliceossomos, cuja função está relacionada à remoção dos íntrons e ligação dos éxons. A maior parte dos íntrons é removida pelo spliceosomo, o qual inclui snRNAs U1, U2, U4, U5, e U6. Os snRNAs são codificados por genes moderadamente repetidos que mostram algumas variações dentro de uma mesma espécie (Mount et al., 2007).

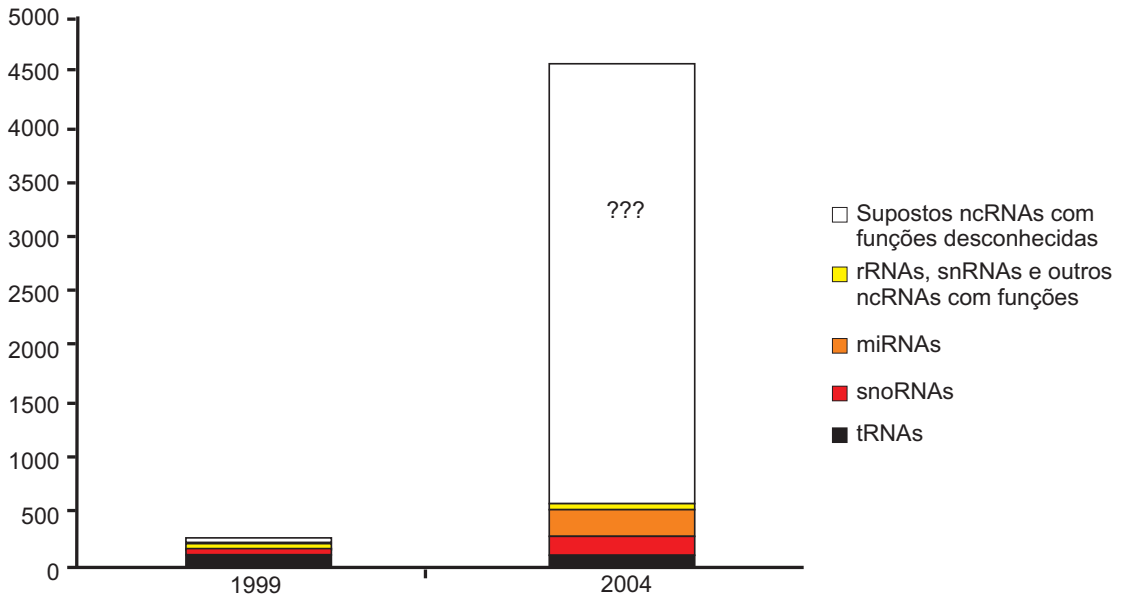

Figura 3.2: O rápido crescimento de números de ncRNAs e candidatos a ncRNA dos mamíferos de 1999 até 2004 (Huttenhofer et al., 2005).

3.3.2 RNA Pequeno do Nucléolo

Os RNAs pequenos do nucléolo (snoRNAs, do inglês *Small Nucleolar RNA*) são assim chamados pela sua localização na célula, no interior do nucléolo. O núcleolo é rico em snoRNAs, os quais possuem geralmente entre 60 e 400 nucleotídeos (Lowe, 1999).

A maioria dos snoRNAs dos mamíferos ocorrem em íntrons (Smith e Steitz, 1997). Alguns snoRNAs têm papéis na produção do RNA ribossômico, mas sua função principal é a modificação de rRNAs (Eliceiri, 1999). Os pequenos RNAs do núcleo estão envolvidos nas diferentes fases da biogênese do ribossomo eucariótico dentro do nucléolo. O RNA ribossômico sofre segmentações e dezenas de modificações nos nucleotídeos antes de unir-se com proteínas ribossomais em um ribossomo maduro.

Existem três tipos básicos de modificações encontradas em rRNA: metilação da base, metilação da ribose e pseudouridilação (Lowe, 1999). Em humanos acontece aproximadamente cerca de 100 a 110 modificações de cada tipo e em leveduras por volta de 50 de cada tipo (Eddy, 2001). Se forem os snoRNAs que guiam todas elas, existima-se que exista ainda um grande número dessas moléculas a ser descoberto.

Os snoRNAs podem ser divididos em dois grupos por divergência na seqüência e na estrutura secundária: C/D *Box* e H/ACA *Box* (Figura 3.3). A maioria dos snoRNAs do tipo C/D *box* estão envolvidos na metilação da ribose em rRNAs, enquanto os snoRNAs do tipo H/ACA são necessários para guiar a pseudouridilação do rRNA. Um pequeno número de snoRNAs em cada família estão envolvidos em outras etapas de processamento do rRNA (Lowe, 1999).

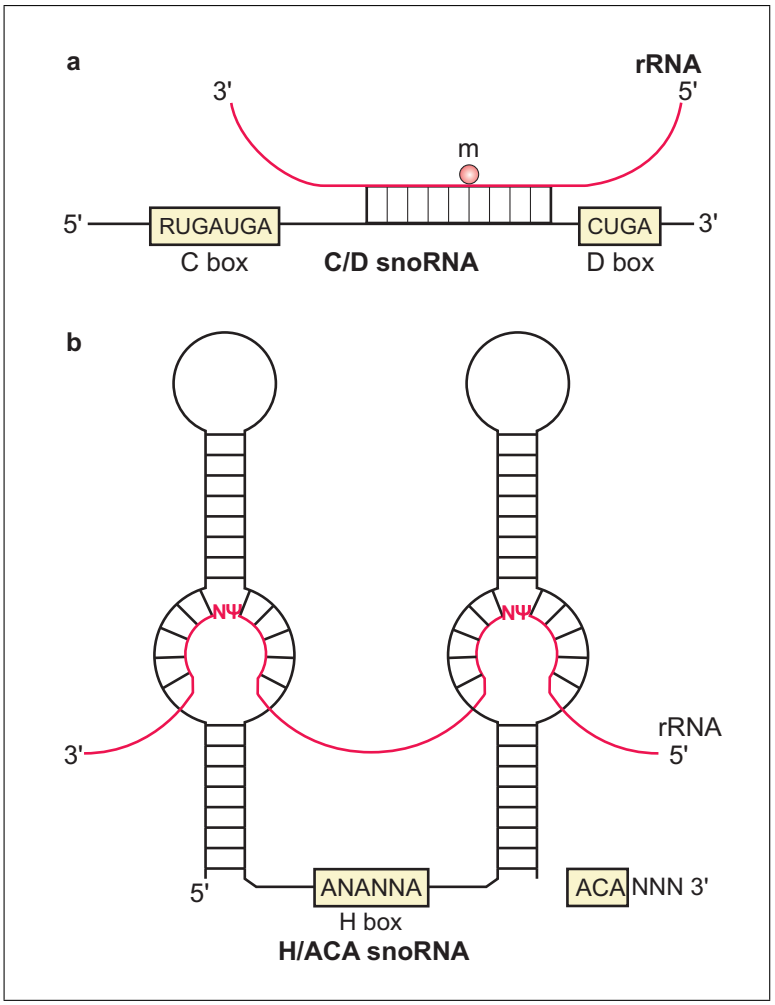

Figura 3.3: Diagramas de snoRNAs orientando modificações em bases de rRNAs. Em (a) tem-se snoRNA do tipo C/D box e em (b) snoRNA do tipo H/ACA box (Eddy, 2001).

Foram encontrados snoRNAs ao longo do DNA de eucariotos, incluindo exemplos específicos de mamíferos (humanos, roedores, porcos), outros vertebrados (galinhas, peixe, cobras), invertebrados metazoários (*Drosophila melanogaster*, *Caenorhabditis elegans*), plantas (*Arabidopsis thaliana*, arroz, milho, batata), leveduras (*Saccharomyces cerevisiae*, *Schizosaccharomyces pombe*), e protistas (tripanossomas, *Euglena gracilis* , *Chlamydomonas reinhardtii*, *Dictyostelium discoideum*) (Lowe, 1999). Técnicas computacionas revelaram 41 novos C/D snoRNA no genoma de leveduras (Lowe e Eddy, 1998) e mais de 60 novos C/D snoRNAs no genoma das *Arabidopsis thaliana* (Barneche et al., 2001; Liang-Hu et al., 2001).

3.3.3 microRNA

Os microRNAs (miRNAs) são pequenos fragmentos de RNA, possuindo um tamanho aproximado de 22 nucleotídeos. Essas moléculas desempenham importantes papéis na regulação da tradução, na regulação da expressão gênica, na inibição da tradução de proteínas e na degradação de RNAs mensageiros (Griffiths-Jones et al., 2006).

Muitos miRNAs vêm sendo identificados ultimamente. Em Eddy (2001) revela-se que 91 miRNAs diferentes foram identificados em três espécies distintas. Griffiths-Jones et al. (2006) citam que mais de 2000 seqüências de miRNA vêm sendo descritas em vertebrados, moscas, lombrigas, plantas e até mesmo em vírus, porém apenas alguns desses miRNAs foram determinados experimentalmente. Na Figura 3.4 são mostrados três diferentes exemplos de miRNAs. Também têm-se desenvolvido estudos que buscam relacionar miRNAs com o diagnóstico de alguns tipos de câncer (He et al., 2005; Lu et al., 2005).

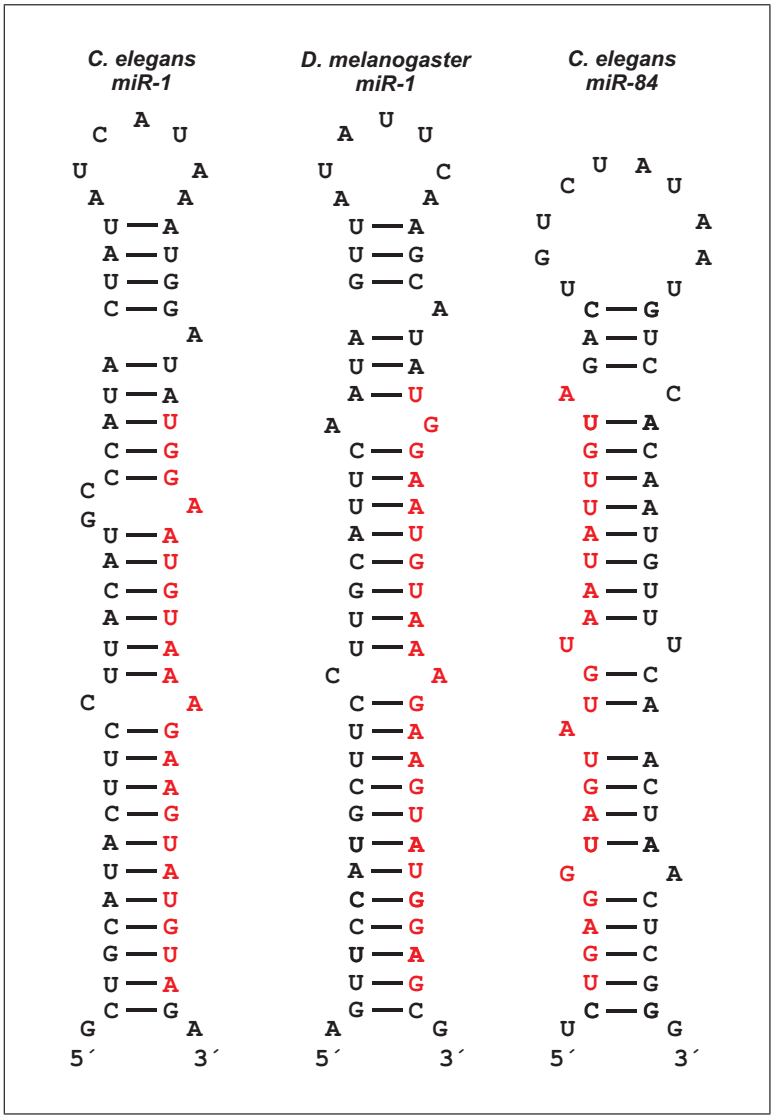

Figura 3.4: Três diferentes exemplos de miRNAs (Eddy, 2001).

3.3.4 RNA Pequeno do Corpo de Cajal

Corpos de Cajal são organelas subnucleares conservadas que estão presentes no nucleoplasma [1] de células, tanto de animais quanto de plantas (Darzacq et al., 2002). Sua forte conservação sugere

[1] Núcleo é preenchido com cromatina composto de DNA e proteínas, conhecido como nucleoplasma (Lodish et al., 2005).

que essa organela desempenha um importante papel, embora existam dificuldades em reconhecer quais funcões estes corpos executam (Ogg e Lamond, 2002).

Os corpos de Cajal são enriquecidos com ribonucleoproteínas spliceossomais e nucleolares (snRNPs e snoRNPs) e componentes da maquinaria de transcrição de bases. Essas organelas possuem funções ainda desconhecidas e foram chamadas assim em homenagem à Ramón e Cajal (Eddy, 2001). Existe grande possibilidade que estes corpos nucleares poderiam ter uma função na biogênese, troca ou armazenamento de snRNPs (Darzacq et al., 2002).

A função do RNA pequeno do corpo de Cajal (scaRNA, do inglês *Small Cajal body-specific*) é similar à dos snoRNAs: servem tanto como guia na metilação quanto na pseudouridilação (Darzacq et al., 2002). Na Figura 3.5 é mostrado que a estrutura dos scaRNAs é composta de ambas as características dos tipos de snoRNAs: *box* C/D e *box* H/ACA (Seção 3.3.2).

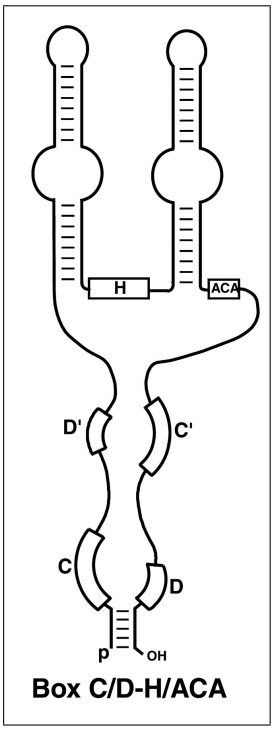

Figura 3.5: Os scaRNAs são freqüentemente compostos tanto de *box* C/D e *box* H/ACA, e eles podem ser guias tanto na metilação quanto na pseudouridilação de RNAs. As posições de *box* conservadas estão indicadas (Kiss et al., 2002).

3.4 Características e Desafios na Detecção de ncRNAs

Por ser um assunto novo e por muitas pesquisas estarem voltadas para esta área, constantes descobertas têm sido feitas em relação aos ncRNAs. Porém, os limites dessas moléculas ainda não foram totalmente desvendados. Dessa forma, ainda não se conhece a dimensão da importância dos ncRNAs.

Diferentemente dos genes que irão codificar proteínas, ainda não foi descoberta nos ncRNAs uma característica forte e presente em todos eles (Bernardo et al., 2003). Diante das dificuldades encontradas em detectar ncRNAs, imagina-se que um grande número de RNAs não codificantes ainda não foram descobertos.

Os ncRNAs carecem de sinais estatísticos comuns em suas seqüências primárias que poderiam ser explorados por algoritmos de detecção (Washietl et al., 2005). Entretanto, muitos RNAs funcionais, como também são chamados os ncRNAs, possuem uma estrutura secundária definida.

A conservação da estrutura secundária serve como evidência para identificação de ncRNAs, dessa forma, estudos comparativos parecem ser a abordagem mais promissora para detecção de RNAs funcionais. Washietl et al. (2005) comentam que estruturas secundárias de RNAs funcionais podem ser identificadas em alinhamentos múltiplos de seqüências com alta sensibilidade e alta especificidade.

Modelos probabilísticos formais, baseados parcialmente em métodos usados no reconhecimento de fala e lingüística computacional, têm sido aplicados para busca de características de consensos complicadas em seqüências biológicas (Durbin et al., 1998). Como concluem Lowe e Eddy (1999) em seu trabalho de identificação de snoRNAs do tipo C/D *box*, a utilização de métodos de modelagem probabilísticas começou a unir as ferramentas necessárias para identificação computacional de ncRNAs em seqüências gênicas.

3.5 Considerações Finais

Neste capítulo foram abordados os principais conceitos relacionados com RNAs não-codificantes. Os RNAs não-codificantes são RNAs que não irão codificar proteínas, possuindo as mais diversas funções biológicas.

As famílias de ncRNAs conhecidas até os anos 80 eram os tRNAs e rRNAs, com o passar do tempo foram descobertas novas classes de ncRNAs, como RNA pequeno do núcleo (snRNA), RNA pequeno do nucléolo (snoRNA), microRNA (miRNA), RNA pequeno do corpo de Cajal (scaRNA), dentre outras. A importância e a quantidade de novas classes de ncRNA vèm crescendo continuamente nos últimos anos devido a estudos e novas descobertas.

Ainda não foi descoberta nos ncRNAs uma característica comum a todos eles, com isso tem-se encontrado dificuldades na detecção de ncRNAs. Muitos modelos têm usado a conservação da estrutura secundária de determinada classe de ncRNA como evidência para sua identificação.

No Capítulo 4 são descritas algumas ferramentas com suas respectivas abordagens utilizadas na detecção ncRNAs.

CAPÍTULO

4

Trabalhos Relacionados

4.1 Considerações Iniciais

Neste capítulo são revistos alguns trabalhos que descrevem técnicas computacionais para identificação de ncRNAs ou trabalhos que descrevem o armazenamento de ncRNAs já identificados.

Na Seção 4.2 são apresentadas algumas ferramentas utilizadas na detecção de ncRNAs. A construção de uma técnica desse tipo se torna desafiadora devido às características dessas moléculas, como pode ser visto no Capítulo 3. Em seguida, na Seção 4.3, são mostrados alguns repositórios de ncRNAs disponíveis. Finalmente, na Seção 4.4 são apresentadas as considerações finais sobre este Capítulo.

4.2 Ferramentas para Detecção de ncRNAs

A detecção de genes é um importante desafio na bioinformática. Muitos estudos estão voltados para detecção de determinadas classes de ncRNAs ou ncRNAs de uma forma geral. Porém, a busca por ncRNAs é mais complexa quando comparada aos genes que codificam proteínas. As ferramentas que há muito tempo vêm sendo desenvolvidas para identificação de genes estão voltadas para genes que codificam proteínas. Como genes de ncRNAs e genes codificantes de proteínas não possuem características fortemente semelhantes, muitos genes de ncRNA não são detectados por essas ferramentas.

Várias técnicas vêm sendo criadas para identificar genes de ncRNAs. Dentre as abordagens pesquisadas, a maioria tem se baseado em sinais de conservação da estrutura secundária e modelos

estatísticos/probabilísticos para sua classificação. Essas ferramentas, em grande parte, utilizam alinhamentos de múltiplas seqüências a fim de identificar estruturas secundárias conservadas. A ferramenta QRNA (Seção 4.2.3) prediz estruturas secundárias de ncRNAs conservadas em alinhamentos de pares de seqüências utilizando uma abordagem probabilística para modelar a estrutura do RNA. Outros programas, como ddbRNA (Seção 4.2.4) e MSARi (Seção 4.2.6), usam alinhamentos múltiplos como entrada para detectar estruturas secundárias de RNAs conservadas. Em RNAz (Seção 4.2.8) uma medida para estabilidade termodinâmica é combinada com a análise de conservação da estrutura secundária. A GenoMiner (Seção 4.2.10) busca por regiões de similaridades entre um genoma ou uma seqüência de transcrito e um genoma completo. Já a ferramenta FastR (Seção 4.2.9) foi criada para busca de ncRNAs em banco de seqüências.

Outras ferramentas que têm sido desenvolvidas buscam detectar classes específicas de RNAs que não codificam proteínas. O tRNAScan-SE (Seção 4.2.1) é um programa desenvolvido para detecção de genes de RNA transportador, em que um modelo probabilístico é construído a partir do alinhamento de múltiplas seqüências e será utilizado para detectar genes de tRNAs. Outra ferramenta, snoscan (Seção 4.2.2), procura por genes de snoRNAs do tipo C/D *box* em uma seqüência gênica. Essa ferramenta também utiliza modelos probabilísticos formais para detectar snoRNAs C/D *box*. A ferramenta snoGPS (Seção 4.2.7) foi criada para detecção de snoRNAs do tipo H/ACA, utilizando também modelo de gene probabilístico. Por fim, a ProMir (Seção 4.2.11) é uma ferramenta criada para predição de possíveis microRNAs em uma seqüência consultada, utilizando um modelo de aprendizado probabilístico.

Algumas abordagens já foram criadas a fim de distinguir mRNA de ncRNA, como a CONC (Seção 4.2.12). Essa ferramenta utiliza uma máquina de vetor de suporte como classificador.

Atualmente, algumas técnicas diferentes identificaram um número surpreendentemente grande de novos genes de ncRNA. Essas ferramentas e abordagens utilizadas serão descritas nas Seções seguintes. Existem outras ferramentas que não foram detalhadas aqui por se distanciarem do interesse deste trabalho ou funcionarem de forma semelhante às ferramentas apresentadas.

4.2.1 tRNAScan-SE

O tRNAScan-SE[1] é um programa desenvolvido para detecção de genes de RNA transportador em seqüências de DNA, por Lowe e Eddy (1997) e Lowe (1999).

O tRNAscan-SE combina três métodos de busca de tRNA já existentes: o modelo de covariância de Eddy e Durbin (1994), o tRNAscan 1.3 de Fichant e Burks (1991) e o algoritmo de busca de tRNAs de Pavesi et al. (1994).

Eddy e Durbin (1994) desenvolveram um método de busca de similaridade de uma estrutura geral de RNA que envolve um perfil probabilístico da estrutura de um RNA ou modelos de covariância (Lowe e Eddy, 1997). Modelos de covariância podem capturar tanto padrões primários

[1]Disponível em `http://lowelab.cse.ucsc.edu/tRNAscan-SE/`.

quanto informações da estrutura secundária pelo uso de gramáticas livres de contexto estocásticas (SCFG, do inglês *stochastic context-free grammars*) (Eddy e Durbin, 1994). Esse modelo é construído a partir de um alinhamento de múltiplas seqüências. As seqüências são procuradas utilizando um modelo de covariância através de um algoritmo de programação dinâmica tridimensional (Lowe, 1999). Segundo Lowe (1999), esse tipo de busca, utilizando modelos de covariância de RNA, tem a vantagem de possuir alta sensibilidade, alta especificidade e aplicabilidade geral para qualquer seqüência de uma família de RNAs de interesse, eliminando a necessidade da criação de uma customização do *software* para cada família de RNA (Lowe, 1999). Porém, existe também uma desvantagem: esses algoritmos com programação dinâmica com modelos de covariância utilizam CPU intensivamente, sendo quase proibitivos (Lowe, 1999).

Eddy e Durbin (1994) relatam que um modelo de covariância identifica cerca de 99,09% de tRNAs verdadeiros e, na época, uma busca no genoma humano com este modelo levaria aproximadamente nove anos e meio de processamento de CPU (baseado em benchmarks de uma CPU SGI Indigo2 R4400/200, 140 SPECint92) (Lowe, 1999). Atualmente, a busca utilizando modelos de covarância ainda é considerada lenta, pois trata-se de um algoritmo de alta complexidade.

A versão 1.3 do tRNAscan por Fichant e Burks (1991) era talvez a mais utilizada para detecção de tRNAs. Essa versão identifica cerca de 97,5% de verdadeiros genes de tRNAs (Lowe, 1999). O algoritmo desse programa utiliza um sistema hierárquico baseado em regras, em que cada candidato a tRNA deve exceder um limite de similaridade determinado empiricamente entre dois promotores intragênicos (Lowe, 1999).

Pavesi et al. (1994) desenvolveram um algoritmo diferente para detecção tRNA. Esse algoritmo procura exclusivamente por sinais de seqüências lineares parecidos com promotores e terminadores de RNA polimerase III de células eucarióticas. A sensibilidade e seletividade deste algoritmo são comparáveis às do tRNAscan 1.3 na detecção de tRNAs em células eucarióticas. Entretanto, o algoritmo de busca Pavesi identifica tRNAs que não foram descobertos por tRNAscan 1.3 e vice-versa.

A entrada do programa tRNAScan-SE consiste em seqüências de DNA ou RNA no formato FASTA. O tRNAscan-SE controla o fluxo de informações entre três programas de predição de tRNA independentes, executa alguns pós-processamentos e gera o resultado (Figura 4.1).

O tRNAscan-SE funciona em três fases, descritas aqui simplificadamente. Na fase inicial o tRNAscan v. 1.4, uma versão otimizada da 1.3, e o algoritmo Pavesi (EufindtRNA) são rodados com a seqüência de entrada. Ambos os resultados são unidos em uma lista de candidatos a tRNAs. Na segunda fase, tRNAscan-SE extrai as subseqüências candidatas que são passadas pelo programa de busca pelo modelo de covariância (*covels*) (Lowe e Eddy, 1997). Por fim, a predição de tRNAs será confirmada com a pontuação da probabilidade do covels, ordenando o limite de tRNA para aqueles que foram preditos por covels e rodados no programa de alinhamento global da estrutura do modelo de covariância (coves) para obter a predição da estrutura secundária. O tRNA é predito através da identificação do anticódon dentro da saída da estrutura secundária do coves.

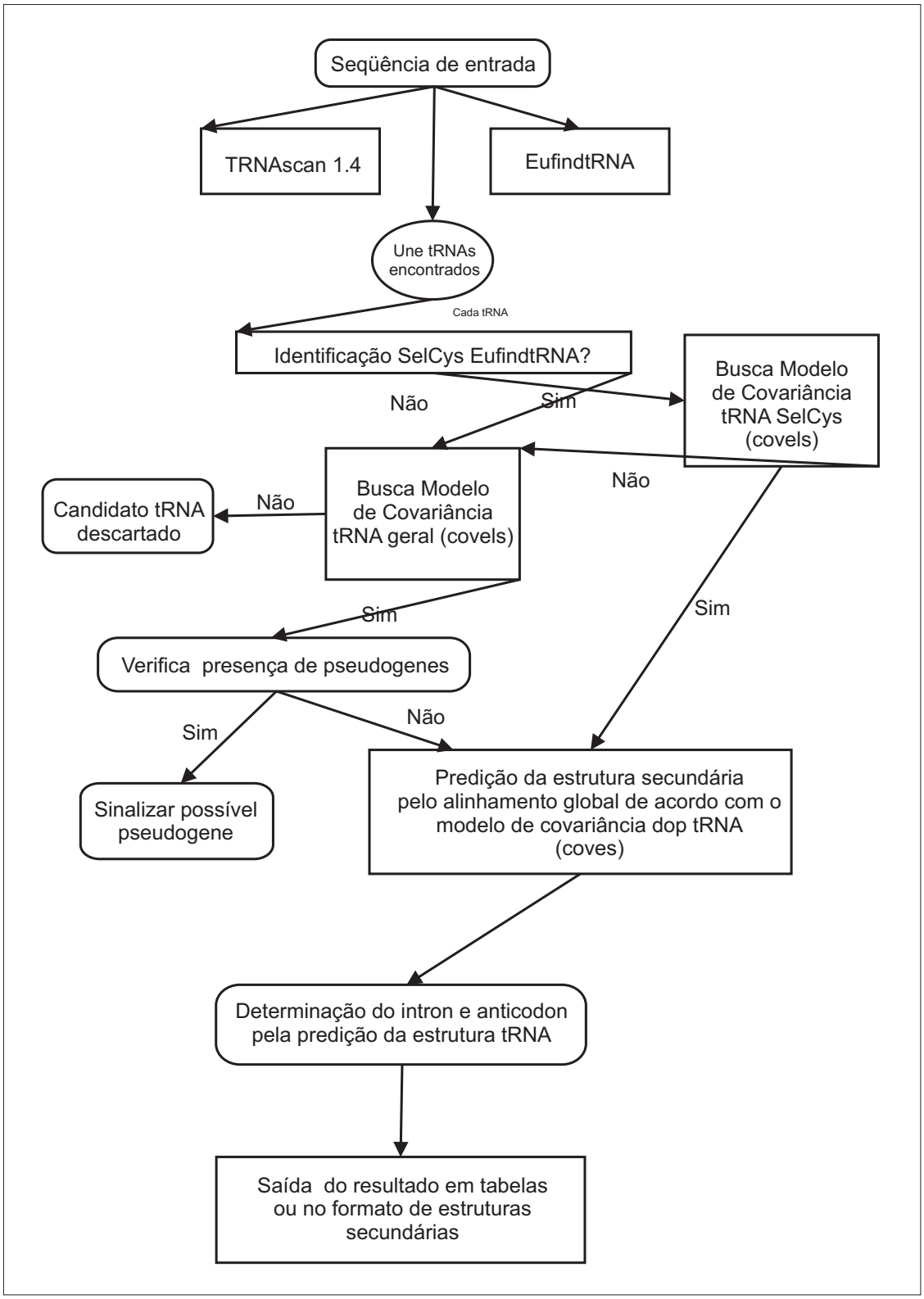

Figura 4.1: Diagrama esquemático do algoritmo do tRNAscan-SE (Lowe e Eddy, 1997).

Lowe e Eddy (1997) afirmam que o tRNAscan-SE consegue encontrar entre 99 e 100% dos genes tRNAs, com aproximadamente 1000 a 3000 vezes a velocidade da busca com o modelo de covariância de tRNA. Na Tabela 4.1 é mostrado um quadro comparativo de ferramentas de busca de

tRNAs. O tRNAscan-SE foi testado em genomas de *M. genitalium*, *H. influenzae*, *M. jannaschii*, *S. cerevisiae*, *S. pombe* e *C. elegans*.

Método de Busca	Positivos Verdadeiros	Falso Positivos	Velocidade da Busca
	(%)	(por Mbp)	(bp/seg)
tRNAscan 1.3	95,1	0,37	400
EufindtRNA	88,8	0,23	373000
Busca pelo modelo de covariância do tRNA	99,8	< 0,002	20
tRNAscan-SE	99,5	< 0,00007	30000

Tabela 4.1: Tabela comparativa dos resultados de tRNAscan 1.3, EufindtRNA, busca pelo modelo de covariância de tRNA e tRNAscan-SE (Lowe e Eddy, 1997).

Eddy (2002) comenta em seu trabalho que o tRNAscan é uma das poucas ferramentas que utilizam gramáticas estocásticas livres de contexto, pois estas são computacionalmente complexas e requerem muito mais tempo e memória do que algoritmos de alinhamento de seqüências primárias.

4.2.2 snoscan

O snoscan[2] procura por genes de snoRNAs do tipo C/D *box* em uma seqüência gênica. Essa ferramenta é baseada no algoritmo descrito por Lowe e Eddy (1999). Além da característica conservada do *box* C e *box* D, o snoRNA do tipo C/D *box*, que é envolvido na metilação da ribose, também contém uma seqüência guia interna que é capaz de se parear com um segmento específico do rRNA (Figura 4.2).

Modelos probabilísticos formais (Durbin et al., 1998) têm sido aplicados em programas que tentam detectar características complexas. Utilizando técnicas de modelagem probabilísticas, como modelos ocultos de Markov (HMM, do inglês *Hidden Markov models*) (Eddy, 1996) ou gramáticas livre de contexto estocásticas (Eddy e Durbin, 1994), em Lowe e Eddy (1999) é proposto um modelo integrado de snoRNA que é baseado na seqüência de características específicas (*features*) para genes dessa família de RNAs (Figura 4.3).

O algoritmo funciona da seguinte forma: inicialmente, para a rápida identificação de candidatos a snoRNAs guias da metilação, é feita uma busca gulosa na seqüência gênica para identificar seqüencialmente 6 características (Figura 4.2). Posteriormente, cada candidato é pontuado de acordo com uma tabela probabilística (Tabela 4.2). Os candidatos são classificados com base na pontuação final da probabilidade.

[2]Disponível em `http://lowelab.ucsc.edu/snoscan/`.

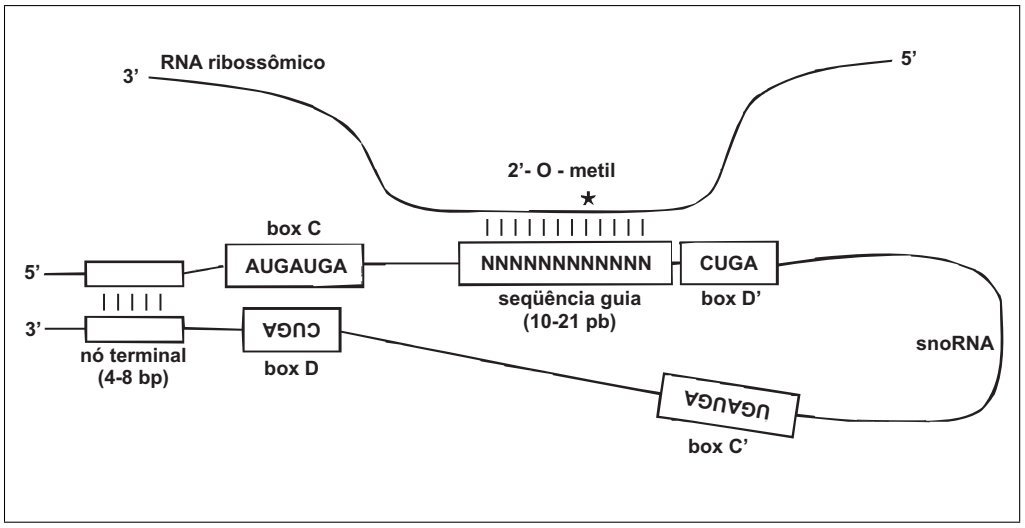

Figura 4.2: Características do snoRNAs do tipo C/D *box* (Lowe e Eddy, 1999).

Identificacão do Estado	*Features*	Modelo	Consenso	Pontução *features* (bits)		
				Melhor	Média	Pior
1	Nó terminal	SCFG, 4 a 8 pb	6 pb (quando presente)	7,60	3,09	0,35
2	*Box* C	7-pb *ungapped* HMM	AUGAUGA	12,73	11,63	5,84
3	*Gap*	Duração do modelo	Comprimento: 6 a 10 pb	-1,59	-2,09	-4,76
4	Seqüência guia	HMM	12-pb duplos	15,67	11,11	2,54
5	*Box* D'	4-pb *ungapped* HMM	CUGA	7,34	4,85	-3,74
6	Gap	Duração do modelo	Comprimento: 36 a 45 pb	-1.59	-2,43	-5,36
7	*Box* D	4-pb *ungapped* HMM	CUGA	8,05	7,92	5,43
8	*Gap*	Duração do modelo	Comprimento: 56 a 75 bp	-1,50	-2,10	-4,17
9	Seqüência guia	HMM	14-pb duplos	18,96	13,98	9,95

Tabela 4.2: Resumo de estados dentro do modelo probabilístico do snoRNA (Lowe e Eddy, 1999).

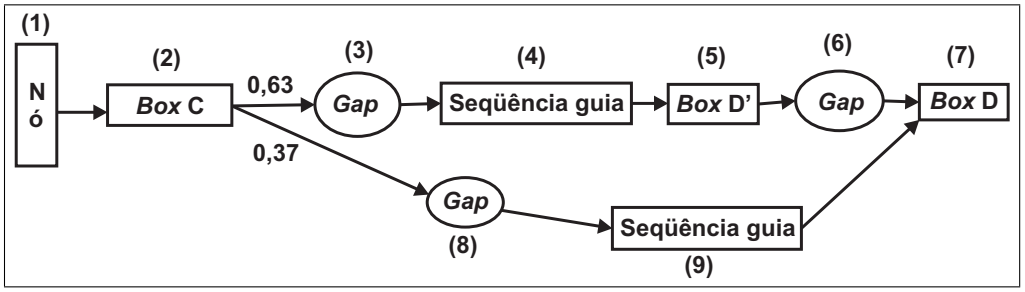

Figura 4.3: Diagrama esquemático do algoritmo do snoscan (Lowe e Eddy, 1999). Cada estado representa uma característica da seqüência com base no modelo probabilístico. As probabilidades das transições são iguais a 1,0, com exceção das transições 2→3 e 2→8, que contam a proporção de snRNAs em que a seqüência guia é adjacente ao *box* D' ou ao *box* D, respectivamente.

A identificação do estado na Tabela 4.2 corresponde à Figura 4.3. Os estados *Ungapped* HMM representam seqüências conservadas com comprimento fixo. Estados do tipo duração do modelo de *gaps* são estimados a partir de distribuições de comprimento. *Gaps* são aberturas introduzidas em um alinhamento de seqüências que buscam compensar tanto inserções como remoções nas

seqüências comparadas. Para cada estado, a característica mais comum (consenso) indica o padrão global procurado. A melhor, a média e a pior pontuação de *feature* são determinadas por 41 snoRNAs guias da metilação como uma indicação da contribuição relativa de cada estado para a informação global no modelo (Lowe e Eddy, 1999).

O snoscan foi utilizado com sucesso na detecção de snoRNAs do tipo C/D *box* em levedura (*Saccharomyces cerevisiae*).

4.2.3 QRNA

Uma nova abordagem para detecção de ncRNAs foi implementada em um programa chamado QRNA, descrito por Rivas e Eddy (2001), sendo considerado um protótipo estrutural para busca de genes de ncRNAs.

Segundo Rivas et al. (2001) a análise comparativa de genomas fornece um grande poder na detecção de ncRNAs. O QRNA faz um alinhamento da estrutura de dois ncRNAs que são suficientemente similares em sua seqüência primária, mas não o bastante para mostrar mudanças de base compensatórias que conservam a estrutura secundária.

O modelo se baseou no trabalho de Badger e Olsen (1999) que descreve uma abordagem que utiliza dois modelos para busca de genes de proteínas. Esse modelo, CRITICA (do inglês, *Coding Region Identification Tool Invoking Comparative Analysis*), é uma ferramenta para identificação de regiões codificantes utilizando análises comparativas. Badger e Olsen (1999) utilizam o programa BLASTN (Altschul et al., 1990) para localizar regiões com similaridades significantes entre duas espécies de bactérias, em seguida, o CRITICA analisa o padrão de mutação, alinhando regiões conservadas em busca de evidências de uma estrutura codificante.

Rivas e Eddy (2001) extenderam a idéia da abordagem de Badger e Olsen (1999). A idéia chave é produzir três modelos probabilísticos (RNA, COD e OTH) descrevendo diferentes limitações sobre o padrão de mutação observado em pares de seqüências alinhadas. A probabilidade do alinhamento dada pelo modelo OTH é apenas o produto das probabilidades das posições alinhadas individualmente. O modelo de COD assume que as seqüências alinhadas codificam proteínas homólogas, logo, em uma região de codificação, espera-se encontrar mutações que fazem substituições que conservam o aminoácido, ou seja, uma abundância de mutações sinônimas. O modelo de RNA assume que um padrão de mutação conserva significantemente uma estrutura secundária do RNA homóloga.

Uma gramática livre de contexto estocástica foi utilizada na construção do modelo RNA de forma semelhante ao trabalho do mesmo autor (Rivas e Eddy, 2000). A idéia fundamental utilizada nessa abordagem é explorar um padrão especial de mutação diferente de simples conservações. Na Figura 4.4 tem-se um exemplo de três alinhamentos distintos com três padrões de mutação diferentes e como eles podem ser pontuados com os três modelos diferentes. Nessa figura visualiza-se como três alinhamentos de pares de composições idênticas com um mesmo número de substituição de bases podem ser classificadas por diferentes padrões de mutação: a hipótese nula da posição

independente (acima), uma região codificantes (meio), ou um RNA estrutural (abaixo). Rivas e Eddy (2001) indicam como cada alinhamento é pontuado de acordo com o modelo que melhor ajusta o padrão de mutação: uma posição por vez para o OTH, um códon por vez para COD e como uma combinação de bases pareadas e posições únicas para RNA.

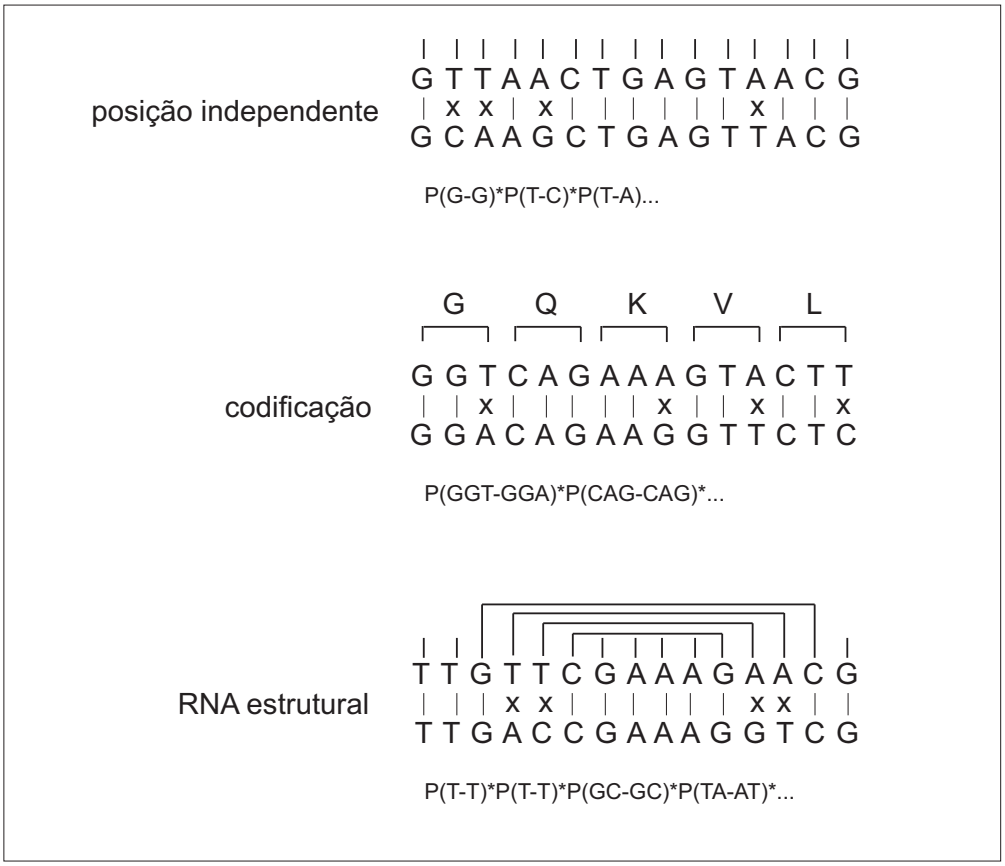

Figura 4.4: Idéia chave das técnicas utilizadas no QRNA (Rivas e Eddy, 2001).

Rivas et al. (2001) utilizaram esta abordagem, QRNA, para a identificação de ncRNAs em *Escherichia coli* (Blattner et al., 1997).

4.2.4 ddbRNA

O ddbRNA foi criado por Bernardo et al. (2003), que motivados pelas dificuldades na detecção de ncRNAs e em sua importância, desenvolveram um algoritmo que é capaz de detectar estruturas secundárias conservadas em alinhamentos múltiplos. Esse algoritmo gasta um tempo computacional proporcional ao quadrado do tamanho da seqüência.

O algoritmo utilizado pelo ddbRNA aplica uma técnica conhecida (Pace et al., 1989), procurando por mutações compensatórias que conservam sua estrutura em alinhamentos múltiplos. A quantidade de mutações compensatórias possíveis é computada e comparada com o número médio de mutações compensatórias obtidas pelo embaralhamento do alinhamento.

Para decidir se o alinhamento realmente contém uma estrutura secundária conservada, executa-se o algoritmo em S embaralhamentos diferentes do alinhamento original. Se o número de mutações compensatórias possíveis no alinhamento original é melhor que a média obtida com as mutações compensatórias com os alinhamentos embaralhados somada com um desvio-padrão K, então o alinhamento é classificado como contendo uma estrutura secundária conservada.

O algoritmo do ddbRNA não utiliza bases que são alinhadas por *gaps*, visto que elas não podem ser mutações compensatórias. Desse modo, alinhamentos usados como entrada para o algoritmo são preprocessados através da remoção dessas bases, que são alinhadas com um *gap* de toda a seqüência no alinhamento.

Existem algumas limitações no algoritmo implementado, citadas pelos próprios autores: nenhuma estrutura secundária é predita e nenhuma informação sobre o local da estrutura secundária dentro do alinhamento é provida. E devido à natureza aleatória do algoritmo pode-se obter classificações diferentes em diferentes execuções do algoritmo para o mesmo alinhamento.

4.2.5 RSearch

Klein e Eddy (2003) apresentam a ferramenta RSearch, que faz uma comparação de uma seqüência contra um banco de seqüências. Ou seja, dada um seqüência com uma estrutura secundária conhecida, busca em uma base de dados de nucleotídeos por RNAs similares com base tanto na estrutura primária quanto na secundária.

Para a busca feita pelo RSearch é necessária uma matriz de substituição para pontuar alinhamentos tanto de bases pareadas quanto de nucleotídeos. As matrizes da BLOSUM são tidas como as melhores para encontrar distância das relações homólogas (Henikoff e Henikoff, 1993). Dessa maneira, Klein e Eddy (2003) optaram por construir uma nova matriz a partir das BLOSUM.

O algoritmo para prover o alinhamento entre o RNA de entrada e a base de dados é baseado em gramáticas estocásticas livres de contexto (Eddy e Durbin, 1994; Sakakibara et al., 1994). Klein e Eddy (2003) utiliza o termo modelo de covariância para descrever o modelo do perfil da SCFG.

A idéia é determinar, para uma seqüência q, quais seqüências no banco de dados têm similaridade local de estrutura com q. Para isso, o RSearch constrói um modelo de covariância para a seqüência de entrada q. Um modelo de covariância contém estados, probabilidades de transição de estados e probabilidades de emissão de símbolos. O modelo é baseado em uma árvore ordenada que captura as interações entre pares de nucleotídeos na estrutura secundária de um RNA. A estrutura da árvore é uma descrição de uma coleção de RNAs relacionados pela estrutura secundária.

Um modelo de covariância representa uma seqüência nucleotídica. O modelo consiste de um conjunto de estados interligados. Os estados formam uma estrutura em árvore, com a raiz usualmente sendo traçada no topo. Em direção às folhas da árvore, os nucleotídeos estão preenchidos, tanto à esquerda quanto à direita até se reunirem no meio da seqüência (Klein e Eddy, 2003).

Cada estado pode emitir ou não nucleotídeos, um nucleotídeo do lado esquerdo, um nucleotídeo do lado direito, ou um par de nucleótidos, um de cada lado. Bifurcações resultam em uma divisão

na seqüência, em que cada metade é preenchida a partir de ambos os lados, ao longo de um a dois ramos bifurcados. O modelo é atravessado por uma série de transições de um estado para o seguinte depois de cada emissão. Cada transição é regida por uma pontuação e apenas um conjunto limitado de transições é permitido. Dado um modelo de covariância parametrizado, algoritmos são usados para pesquisar uma base de dados por seqüências homólogas e alinhar o modelo de acertos encontrados no banco de dados (Eddy e Durbin, 1994; Sakakibara et al., 1994).

O algoritmo de alinhamento usa programação dinâmica em três dimensões. Cada célula da matriz contém a probablidade de ocorrência de um alinhamento de subseqüências começando em um certo estado. O algoritmo produz um índice de significância estatística para os alinhamentos, o que é bastante desejável em aplicações biológicas (Klein e Eddy, 2003).

A sensibilidade do algoritmo na identificação de RNAs não-codificantes é boa, mas seu desempenho é ruim, principalmente em termos do tempo de CPU. A complexidade do tempo no algoritmo de busca no pior caso é O (nm^3), para uma entrada de comprimento *m* e banco de dados de comprimento *n* (Klein e Eddy, 2003).

Segundo Klein e Eddy (2003) existem três áreas em que futuros esforços deveriam ser focados. Dentre essas áreas, está a melhoria do tempo gasto pela ferramenta. A RSearch é bastante lenta e é muito importante que o uso de novas heurísticas melhorem sua velocidade.

4.2.6 MSARi

Coventry et al. (2004) propõem um método para detecção da estrutura secundária de RNA usando grandes alinhamentos de múltiplas seqüências (MSA, do inglês *multiple sequence alignments*). Segundo Coventry et al. (2004) a evidência estatística de estruturas secundárias de RNAs conservadas em inúmeras seqüências é geralmente forte. Modelos estatísticos robustos podem ser usados para detectá-las.

A abordagem utilizada em MSARi[3] é baseada em computar o significado estatístico de possíveis regiões de estruturas secundárias que são conservadas entre candidatos ortólogos, permitindo pequenas variações entre o pareamento de bases desses ortólogos.

O algoritmo usado por MSARi é composto por dois passos:

- Primeiramente, o algoritmo procura por colunas do alinhamento que tenham *gaps* ou bases discordantes, mais que sejam significativas estatisticamente. O MSARi pode tolerar falhas em alinhamentos no pareamento de ortólogos até uma distância de dois caracteres. Ele precisa apenas encontrar uma pequena significância nas regiões pareadas para identificá-las como estruturas secundárias conservadas.

- Em segundo lugar, a abordagem de MSARi estima a significância da variação em seqüências altamente redundantes, baseada na determinação de que porção da seqüência dentro

[3]MSARI está disponível em `http://theory.csail.mit.edu/MSARi/`.

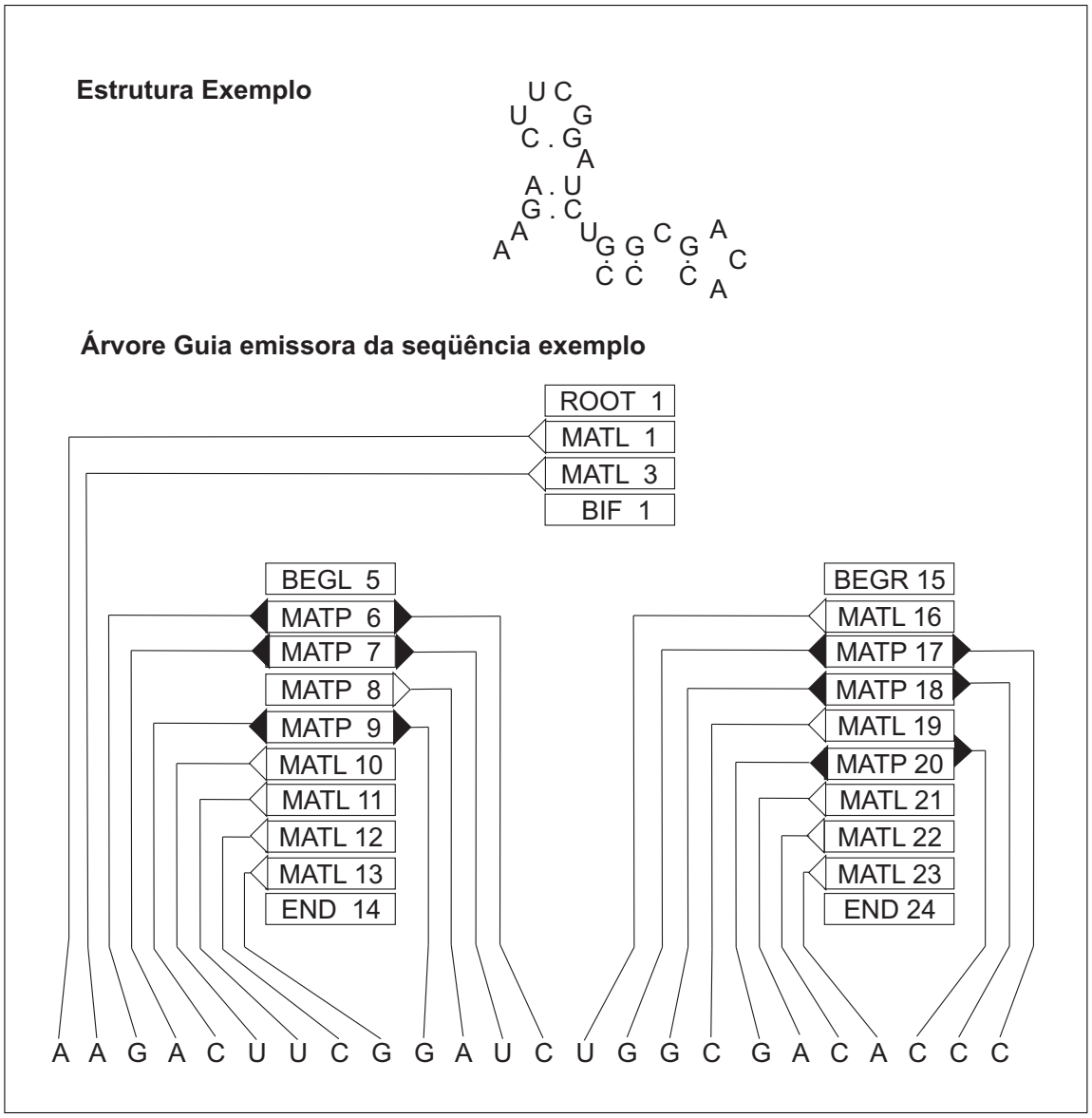

Figura 4.5: Um exemplo da arquitetura SCFG. A seqüência no topo mostra a estrutura secundária. Abaixo é mostrada a arquitetura do modelo que irá produzir esta seqüência. Os triângulos escuros representam nós que emitem pares de bases e apontam para as bases que eles emitem. Os triângulos claros representam os nós emissores de um único nucleotídeo e apontam para o nucleotídeo que eles emitem (Klein e Eddy, 2003).

dos MSAs deve ser tratado como evoluções de outras seqüências e de que porções são tão diferentes que elas devem ser tratadas como posições independentes.

Coventry et al. (2004) utilizaram MSARi para rastrear a base de dados de *The Institute for Genomic Research Eukaryotic Gene Orthologs* (TIGR EGO) por ortólogos com estruturas secundárias de RNAs conservadas.

4.2.7 snoGPS

O snoGPS[4] foi criado para detecção de snoRNAs do tipo H/ACA. Os princípios utilizados no snoGPS estão descritos no trabalho desenvolvido por Schattner et al. (2004). Em contraste com o snoRNA do tipo C/D *box*, os snoRNAs H/ACA possuem pequenos e menos conservados *motifs*[5] em suas seqüências primárias (Figura 4.6) (Schattner et al., 2004).

O programa snoGPS utiliza um algoritmo de busca determinístico e um modelo probabilístico de gene (Durbin et al., 1998) para procurar por genes de RNA com seqüências primárias fracamente conservadas e *motifs* de estruturas secundárias. Segundo Schattner et al. (2004) a natureza híbrida do programa busca combinar tanto a eficiência dos algorítimos determinísticos quanto a sensibilidade dos modelos probabilísticos na detecção de *motifs* fracamente conservados.

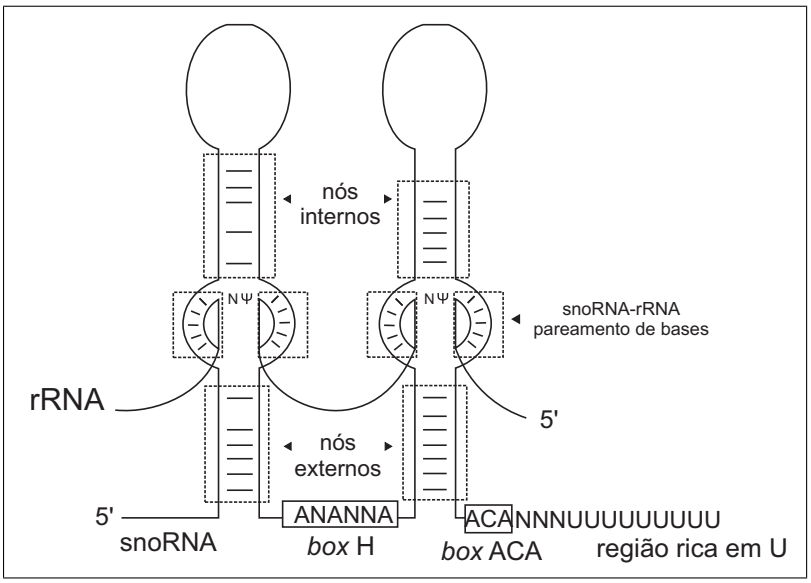

Figura 4.6: Diagrama esquemático de um modelo de snoRNA H/ACA. As seqüências *motifs* do H/ACA snoRNA são indicadas, incluindo as seqüências guia da esquerda e da direita, os *boxes* H e o ACA, os nós 5' e 3', e a região rica em U (Schattner et al., 2004).

Na prática o programa funciona em duas fases:

- Na primeira fase executa uma série inicial de testes determinísticos que limitam o possível espaço de busca, enumerando todas as possíveis características (*features*) para um determinado candidato.
- A segunda fase consiste em rotinas de pontuação que medem quão similar as *features* identificadas são quando comparadas às de um conjunto de treinamento de RNAs já conhecidos.

Somando-se a pontuação dos componentes com o modelo probabilístico, obtem-se uma pontuação final usada para classificar os candidatos.

[4] O programa snoGPS está disponível em `http://lowelab.ucsc.edu/snoGPS/`.
[5] *Motifs* são combinações regulares (padrões) da estrutura de uma molécula.

Esta ferramenta identificou snoRNAs do tipo H/ACA *box* em leveduras (*Saccharomyces cerevisiae* (Schattner et al., 2004)).

4.2.8 RNAz

Washietl et al. (2005) utilizam uma abordagem alternativa para detectar ncRNAs, chamada RNAz[6]. Em RNAz uma medida para a estabilidade termodinâmica é combinada com uma medida para a conservação da estrutura. Utilizando a combinação de ambas as pontuações, fica-se apto a descobrir RNAs funcionais em alinhamento de múltiplas seqüências (Washietl et al., 2005).

O RNAz é baseado no algoritmo de predição de estruturas utilizando energia mínima livre (MFE, do inglês *minimum free energy*) (Zuker e Stiegler, 1981; Hofacker et al., 1994). Esse trabalho se baseia no fato que RNAs estruturais tem duas características (*features*) (Washietl, 2006):

- Estabilidade termodinâmica pouco comum. O RNAz calcula uma medida normalizada da estabilidade termodinâmica e, em seguida, uma pontuação (*score z*) é calculada.
- Conservação da estrutura secundária. O RNAz faz a predição de uma estrutura secundária consenso de um alinhamento utilizando a abordagem RNAalifold, por Hofacker et al. (2002). Mutações compensatórias e consistentes proporcionam um aumento na energia, enquanto mutações inconsistentes produzem uma penalidade. O RNAz calcula o índice de conservação da estrutura (SCI, do inglês *structure conservation index*).

Duas *features* independentes caracterizam a estrutura do ncRNA: *z-score* e SCI. Essas medidas são usadas para classificar o alinhamento como RNA estrutural ou outros. Posteriormente, o RNAz utliza um algoritmo de aprendizado com máquina de vetor de suporte (SVM, do inglês *support vector machine*) que foi treinado em um grande conjunto de ncRNAs já conhecidos (Washietl, 2006). O RNAz detecta estruturas secundárias de RNAs conservadas e estáveis termodinamicamente em alinhamentos de múltiplas seqüências e, em seguida, os candidatos a ncRNAs são filtrados.

Aplicou-se o RNAz em *Saccharomyces cerevisiae* como descrito por Washietl (2006).

4.2.9 FastR

O FastR, por Zhang et al. (2005), é uma ferramenta criada para busca de ncRNAs em banco de dados. Segundo os autores, a criação do FastR objetivou resolver o seguinte problema: dada uma seqüência de RNA com estrutura secundária conhecida, detectar de forma eficiente todos os homólogos estruturais em um banco de seqüências gênicas.

A abordagem do FastR é baseada em filtros estruturais, a fim de eliminar uma grande porção do banco de dados e reter os verdadeiros homólogos. Essa ferramenta executa em duas fases (Zhang et al., 2005):

[6]Disponível em `http://www.tbi.univie.ac.at/~wash/RNAz/`.

- Inicialmente, o banco de dados é filtrado para identificar subseqüências que possuam características (*features*) estruturais similares com a entrada que está sendo consultada.

- Na segunda fase, as subseqüências selecionadas são alinhadas com a consultada utilizando um alinhamento estrutural da seqüência.

FastR utiliza um algoritmo com programação dinâmica para gerar automaticamente opções de filtros com uma alta especificidade (Zhang et al., 2005). O *software* permite que o usuário ajuste os parâmetros computados para obter a sensibilidade desejada mantendo sua especificidade.

Após a realização do filtro, a ferramenta irá alinhar as regiões com a seqüência de entrada. Dentre os tipos de alinhamento de RNA existentes, FastR faz o alinhamento estrutural da seqüência de RNA, em que alinha a seqüência com uma estrutura secundária ou um perfil da estrutura (Bafna et al.; Durbin et al.; Lenhof et al. *apud* Zhang et al. (2005)).

FastR foi aplicado na detecção de *riboswitches*, que constituem uma classe de RNA encontrado em uma região que não sofre tradução, constituindo elementos de ncRNAs.

4.2.10 GenoMiner

O GenoMiner[7], por Castrignano et al. (2005), é um *software* que busca por regiões de similaridades entre um genoma ou uma seqüência de transcrito submetida e um genoma completo especificado pelo usuário.

Esse programa identifica regiões conservadas (CSTs, do inglês *conserved sequence tags*) e provê uma predição de sua natureza codificante ou não-codificante.

A análise feita pelo GenoMiner é realizada em três passos:

- O primeiro passo consiste na definição da região da seqüência similar com a seqüência consultada no genoma. O melhor alinhamento local entre a seqüência consultada e o genoma selecionado é identificado usando o BLAT (Kent, 2002) e esses alinhamentos definem uma ou mais regiões similares no genoma selecionado.

- Em seguida, é feita a identificação de CSTs por um alinhamento mais sensível, similar ao BLAST (Altschul et al., 1997), entre a consulta e as regiões similares do genoma selecionado.

- Por fim, no terceiro passo, é feita uma estimativa da natureza codificante ou não-codificante dos CSTs detectados. Nesse sentido, é feita a computação de uma pontuação adequada para possíveis codificantes (CPS). O CPS (do inglês *coding potential score*) é determinado como descrito por Castrignano et al. (2004).

[7] O GenoMiner está disponível na web em `http://www.caspur.it/GenoMiner/`.

Castrignano et al. (2004) apresentam um ferramenta, a CSTminer[8], com um algoritmo que, comparando seqüências gênicas submetidas pelo usuário, identifica blocos conservados e estima sua natureza codificante ou não-codificante através do cálculo do CPS.

Segundo Castrignano et al. (2005), o GenoMiner permite ao usuário procurar a seqüência consultada em inúmeros genomas de vertebrados em uma única execução, provendo uma saída com uma interface amigável.

4.2.11 ProMiR

O ProMir[9] é uma ferramenta para predição de possíveis microRNAs em uma seqüência, utilizando um modelo de aprendizado probabilístico (Nam et al., 2005). Os autores sugerem um modelo de aprendizado probabilístico baseado em modelos ocultos de Markov. Esse modelo foi usado para implementar um método geral de predição de microRNA que identifica homólogos próximos, bem como homólogos distantes.

Segundo Nam et al. (2005) a informação estatística de genes de microRNAs é insuficiente para identifica-los. Dessa forma, a predição de microRNAs torna-se difícil utilizando métodos computacionais. Entretanto, utilizando de forma simultânea tanto a informação estrutural quanto as seqüências de precursores de microRNAs é possível desenvolver um método computacional com o uso de modelos de Markov ocultos. Esses modelos são adequados para o aprendizado da informação estrutural e seqüencial ao mesmo tempo.

O método utilizado na criação do ProMir combina tanto características da seqüência quanto características da estrutura de genes de microRNAs em um *framework* probabilístico e, simultaneamente, decide se um gene de microRNA e se uma região de microRNA maduro estão presentes.

Nam et al. (2006), posteriormente, introduziram um método melhorado para identificar microRNAs conservados e não conservados próximos a microRNAs conhecidos ou candidatos a microRNAs. Essa nova versão foi chamada de ProMir II[10].

4.2.12 CONC

CONC, do inglês *coding or non-coding*, por Liu et al. (2006), é um método que busca distinguir RNAs que codificam proteínas de RNAs não-codificantes. Essa ferramenta utiliza uma metodologia baseada em máquinas de vetor de suporte e busca classificar transcritos a partir de características (*features*) que esses devam possuir se forem codificar proteínas. As SVMs são utilizadas para combinar diferentes características importantes escolhidas inicialmente em grande parte pela intuição.

[8] A CSTminer está disponível em `http://t.caspur.it/CSTminer/`.
[9] ProMir está disponível em `http://bi.snu.ac.kr/ProMiR/`.
[10] A nova versão, ProMir II, está disponível em `http://cbit.snu.ac.kr/~ProMiR2/`.

Para a classificação, o CONC utiliza características como comprimento do peptídeo, composição do aminoácido, conteúdo da estrutura secundária, porcentagem predita de resíduos expostos, composição da entropia, entropia do alinhamento, dentre outras.

As SVMs, de forma semelhante a outros algoritmos de aprendizado de máquina supervisionados, tentam aprender a partir de dados de entrada já classificados, que para esse problema são rotulados em RNAs que codificam proteínas e RNAs que não codificam proteínas (Liu et al., 2006). Essas regras "aprendidas" serão utilizadas para classificar um novo dado.

Liu et al. (2006) treinaram as máquinas de vetores de suporte utilizando ncRNAs de eucariotos dos banco de dados RNAdb e NONCODE (Seção 4.3), obtendo bons resultados na distinção de RNAs codificantes de ncRNAs.

4.2.13 Busca utilizando estatísticas de composição de base

Em seu trabalho, Schattner (2002) investiga a viabilidade da utilização da estatísticas da composição de bases para distinguir entre regiões ricas em ncRNAs e regiões pobres em ncRNA do genoma. O objetivo é separar uma região genômica em dois componentes, um componente com alta probabilidade de conter ncRNAs e o outro com baixa probabilidade de conter ncRNAs.

Schattner (2002) baseia-se na idéia de que, para alguns genomas, a percentagem de determinados nucleotídeos pode servir como um filtro para regiões ricas em ncRNAs. Diante de um banco de seqüências, para cada seqüência foram feitos os seguintes cálculos estatísticos: freqüências observadas de bases CG, $(C + G)\%$, diferença da base G e C, $(G - C)\%$, A e T, $(A - T)\%$.

A Tabela 4.3 mostra resultados da estatística de composição de bases para alguns genomas.

	No. de seqüências usadas	(G+C)%	ρ(CG)	(G-C)% diferença	(A-T)% diferença
(A) Média da estatística da composição de bases do RNA					
M. jannaschii	48	63,1 (7,3)	0,75 (0,24)	8,1 (9,7)	-3,3 (12,9)
Plasmodium	59	32,1 (7,2)	0,94 (0,56)	12,7 (6,3)	-1,6 (4,1)
C.elegans	59	53,5 (8,2)	0,96 (0,23)	6,8 (10,1)	-9,6 (11,4)
H.sapiens	186	48,7 (9,1)	0,60 (0,41)	7,5 (11,8)	-5,8 (13,0)
(B) Estatístca da composição de bases do Genoma					
M. jannaschii		31,4 (6,9)	0,34 (0,47)	1,4 (36,9)	-0,34 (18,8)
P.falciparum Chr. II		20,0 (8,4)	0,75 (1,3)	0,73 (34,5)	-1,7 (24,0)
C.elegans Chr. I		35,9 (8,8)	1,03 (0,68)	0,65 (25,0)	-0,61 (19,6)

Tabela 4.3: Estatística da Composição de Base para RNAs e genomas (Schattner, 2002). Onde $p(CG)$ é a freqüência com que ocorre o dinucleotídeo CG.

4.3 Base de Dados de ncRNAs

Diante de estudos tanto experimentais como computacionais, diversos ncRNAs vêm sendo identificados e repositórios deles já estão disponíveis.

NONCODE (Liu et al., 2005) é uma base de dados de conhecimento integrado dedicada a ncRNAs. Todos os ncRNAs do NONCODE foram filtrados automaticamente da literatura e do GenBank[11] e, em seguida, tratados manualmente. O NONCODE inclui quase todos os tipos de ncRNAs, menos tRNAs e rRNAs. Mais de 80% das entradas do NONCODE estão baseadas em dados experimentais. A primeira versão de NONCODE (v1.0) contém 5339 seqüências de 861 organismos (Liu et al., 2005).

RNAdb[12] é uma base de dados de ncRNAs de mamíferos que contém seqüências e anotações de milhares de ncRNAs, alguns com suas funções documentadas, mas a maioria com sua importância não esclarecida.

Outro repositório é o fRNAdb[13] (do inglês, *functional RNA database*) (Kin et al., 2006). O fRNAdb é uma base de dados que integra um conjunto de outras base de dados como NONCODE e RNAdb. Essa base de dados provê uma interface eficiente para ajudar os usuários a filtrar determinados transcritos utilizando seus próprios critérios para classificar a saída dos candidatos a RNAs funcionais.

A ASRP[14] (do inglês *Arabidopsis thaliana small RNA project*) é uma base de dados de sRNAs de *Arabidopsis thaliana*.

A miRBase[15] é uma base de dados de microRNAs (Griffiths-Jones et al., 2006). A base de dados snoRNAbase[16] é uma base de dados de snoRNAs humanos do tipo H/ACA e C/D *box* e a snoRNA *Database*[17] também é uma base de dados de snoRNAs. Tem-se também uma base de dados de snoRNAs de plantas: *Plant snoRNA Database*[18], e uma base de dados de snoRNAs de leveduras (*Saccharomyces cerevisiae*): *Yeast snoRNA Database*[19].

O banco de dados Rfam[20] (Griffiths-Jones et al., 2003, 2005) é um conjunto de alinhamentos de múltiplas seqüências e modelos de covariância que representam famílias de ncRNAs.

[11] Disponível em `http://www.ncbi.nlm.nih.gov/Genbank/`.
[12] RNAdb está disponível em `http://research.imb.uq.edu.au/rnadb/`.
[13] O fRNAdb está disponível em `http://www.ncrna.org/`.
[14] A ASRP está disponível em `http://asrp.cgrb.oregonstate.edu/`.
[15] A base de dados miRBase está disponível em `http://microrna.sanger.ac.uk/`.
[16] Disponível em `http://www-snorna.biotoul.fr/`.
[17] A snoRNA *Database* está disponível em `http://lowelab.ucsc.edu/snoRNAdb/`.
[18] Disponível em `http://bioinf.scri.sari.ac.uk/cgi-bin/plant_snorna/home`.
[19] Disponível em `http://people.biochem.umass.edu/fournierlab/snornadb/main.php`.
[20] Rfam está disponível em `http://www.sanger.ac.uk/Software/Rfam/`.

4.4 Considerações Finais

Neste capítulo foram apresentadas algumas abordagens utilizadas na detecção de ncRNAs bem como repositórios de ncRNAs disponíveis.

Muitas técnicas vêm sendo criadas a fim de identificar ncRNAs. Muitas abordagens baseiam-se na conservação da estrutura secundária e modelos probabilísticos para sua classificação. Essas técnicas identificaram um grande número de novos genes de ncRNA apesar de muitas ainda precisarem de refinamentos.

Devido aos estudos e pesquisas feitos na detecção por ncRNAs, repositórios desses RNAs identificados vêm sendo disponibilizados.

CAPÍTULO

5

Modelo Proposto para Busca de ncRNAs

5.1 Considerações Iniciais

A finalidade deste capítulo é apresentar o modelo proposto para detecção de ncRNAs. Na Seção 5.2 é apresentada a motivação para a realização deste trabalho. Na Seção 5.3 é apresentado o modelo proposto para busca de RNAs não-codificantes. Finalmente, na Seção 5.4 são descritos os experimentos realizados, bem como seus resultados e desempenhos.

5.2 Motivação

A Bioinformática utiliza conhecimentos da computação a fim de processar e solucinar problemas biológicos. Muitos desses problemas são simples, porém se tornam complicados devido ao enorme volume de dados utilizados em sua solução. Outros são de natureza combinatória complexa, mesmo para poucos dados. Há ainda aqueles que são complexos em função da ausência de um modelo biológico abrangente. Nesse contexto, a fim de extrair informações desses dados, é indispensável o uso de técnicas computacionais, que serão utilizadas para possibilitar ou auxiliar a tentativa de resolução de problemas na área de maneira mais rápida e eficiente.

Na década de 90 confirmou-se que vários tipos de moléculas de RNA que não são traduzidas estão presentes em muitos organismos diferentes e afetam uma variedade grande de processos (Liu

et al., 2005). Os ncRNAs (Capítulo 3) têm sido alvo de inúmeras pesquisas recentes. Tem-se observado que os ncRNAs são responsáveis por uma gama notável de reações e processos biológicos.

Apesar de sua importância funcional, os métodos biológicos e computacionais para a detecção e caracterização de RNAs não-codificantes ainda são imprecisos e incompletos. Imprecisos no sentido de serem incapazes de detectar um grande número de ncRNAs. Isso se deve em parte ao fato de que a natureza biológica dessas moléculas ainda é pouco conhecida. E incompletos no sentido de não explorarem múltiplos aspectos combinatórios do problema. Isso se deve principalmente à complexidade das alternativas existentes para a identificação de ncRNAs.

Mesmo que atualmente muitos estudos tenham se voltado para busca de ncRNAs, isto se torna um desafio pois os genes de ncRNAs são tipicamente curtos, não têm um padrão de seqüência bem comportado e são caracterizados mais pela estrutura secundária do que pela seqüência primária (Huttenhofer et al., 2005). Do ponto de vista experimental, os ncRNAs são caracterizados pela ausência de tradução, o que dificulta a confirmação em laboratório de hipóteses computacionais.

Por ser uma área em recente e constantes descobertas, a busca de novos ncRNAs e formas eficientes para sua identificação são muito importantes para uma compreensão clara das funções e da complexidade dos organismo.

5.3 Modelo Proposto para Busca de ncRNAs

O objetivo deste trabalho é obter uma ferramenta melhor para a comparação de uma seqüência de RNA não-codificante contra um banco de seqüências, ou seja, possuir um tempo de execução menor que as alternativas existentes.

A estratégia proposta neste trabalho de mestrado para uma melhoria na busca por RNAs não-codificantes teve como base a ferramenta Infernal[1], objetivando diretamente a melhoria do seu tempo de execução. Essa ferramenta funciona de forma similar à RSearch (Seção 4.2.5), que inclusive utilizou a Infernal como base.

O Infernal é uma abordagem baseada em Gramáticas Estocásticas Livres de Contexto (SCFGs, do inglês *Stochastic Context-Free Grammars*) (Eddy e Durbin, 1994; Sakakibara et al., 1994). A ferramenta Infernal constrói um RNA consenso que é chamado de Modelo de Covariância (CM, do inglês *Covariance Model*). O modelo de covariância é um caso especial de SCFGs concebido para modelagem da seqüência e estrutura do RNA consenso. O Infernal usa esse modelo para buscar por RNAs homólogos em bases de dados de seqüências de ácidos nucléicos.

O modelo de covariância é construído a partir do alinhamento de múltiplas seqüências (ou uma seqüência única de RNA) com a estrutura secundária consenso marcando em que posições o alinhamento é único e marcando os pareamentos de bases. Os modelos de covariância criados pelo Infernal atribuem pontuações específicas para determinadas posições. Essas pontuações são

[1]Disponível em `http://Infernal.janelia.org/`.

obtidas a partir da contagem de resíduos, pareamento de bases, inserções e deleções na entrada do alinhamento combinadas com informações previamente obtidas a partir do alinhamento estrutural do RNA ribossômico.

A ferramenta Infernal é composta de programas que são utilizados em combinação na busca de ncRNAS:

1. **Construção do Modelo de Covariância a partir de uma entrada utilizando *cmbuild*.** O *cmbuild* tem como entrada um alinhamento estrutural de múltiplos RNAs no formato estocolmo (do inglês *Stockholm format*) (Figura 5.1) (Eddy, 2003) e cria um arquivo que contém o modelo de covariância e que será utilizado por outras funções do Infernal.

2. **Pesquisa bases de dados por possíveis homólogos utilizando *cmsearch***. Dado um arquivo contendo o modelo de covariância obtido com o *cmbuild* e uma base de dados como entrada, o *cmsearch* busca na base de dados por *hits* com alta pontuação e resulta em alinhamentos de cada *hit* em um formato similar à estrutura BLAST (Altschul et al., 1997).

3. **Alinhamento de possíveis homólogos utilizando *cmalign***. A partir de um arquivo de modelo de covariância e outro arquivo que contenha possíveis homólogos, o *cmalign* alinha seqüências de acordo com o modelo, criando um alinhamento de múltiplas estruturas no formato Estocolmo. Esse alinhamento poderá ser utilizado como entrada na construção de um modelo de covariância pelo *cmbuild*.

No formato estocolmo os pareamentos de bases são identificados pelos símbolos () (parênteses), [] (colchetes) e {} (chaves), assim como as bases pareadas, esses símbolos sempre devem estar formando pares. Bases únicas e resíduos são representados pelos símbolos _ (sublinhado), - (hífen), , (vírgula), : (dois pontos), . (ponto) e ˜ (til). A escolha desses símbolos não tem um significado especial. Eddy (2003) descreve esse formato com maiores detalhes.

Conforme ilustrado na Figura 5.2, para efetuar uma busca por ncRNAs homólogos em uma base de dados usando a ferramenta Infernal, deve-se primeiramente construir o modelo de covariância utilizando *cmbuild* a partir do arquivo de entrada. Em seguida, com o arquivo contendo o modelo de covariância será possível utilizar o *cmsearch* para efetuar a busca por RNAs homólogos na base de dados.

A ferramenta Infernal foi escolhida para este trabalho por ser considerada a ferramenta mais sensível e específica para pesquisa de RNAs homólogos (Freyhult et al., 2007) e, além disso, a ferramenta requer melhorias pois é lenta e gasta uma grande quantidade de tempo para processamento (Eddy, 2003). Adicionalmente, o Infernal é uma ferramenta de código aberto implementada em ANSI C. Essa ferramenta foi projetada para ser compilada e utilizada em plataformas UNIX. A versão do Infernal utilizada foi a 0.81.

A partir de uma análise mais detalhada da ferramenta Infernal foram identificados possíveis pontos para melhoria do tempo de execução do algoritmo. Dessa forma, este trabalho objetivou

```
# STOCKHOLM 1.0
#=GF AU    Infernal 0.81

tRNA1             GCGGAUUUAGCUCAGUuGGG.AGAGCGCCAGACUGAAGAUCUGGAGGUCC
tRNA2             UCCGAUAUAGUGUAAC.GGCuAUCACAUCACGCUUUCACCGUGGAGA-CC
tRNA3             UCCGUGAUAGUUUAAU.GGUcAGAAUGGGCGCUUGUCGCGUGCCAGA-UC
tRNA4             GCUCGUAUGGCGCAGU.GGU.AGCGCAGCAGAUUGCAAAUCUGUUGGUCC
tRNA5             GGGCACAUGGCGCAGUuGGU.AGCGCGCUUCCCUUGCAAGGAAGAGGUCA
tRNA6             GCGGAUUUAGCUCAGUuGGG.AGAGCGC------CAGAC----GAGGUCC
#=GC SS_cons      ((((((,,<<<<___.___._>>>>,<<<<<_______>>>>>,,,,,<
#=GC RF           gccgacaUaGcgcAgu.GGu.AgcgCgccagcuUgaaaagcuggAGgucc

tRNA1             UGUGUUCGAUCCACAGAAUUCGCA
tRNA2             GGGGUUCGACUCCCCGUAUCGGAG
tRNA3             GGGGUUCAAUUCCCCGUCGCGGAG
tRNA4             UUAGUUCGAUCCUGAGUGCGAGCU
tRNA5             UCGGUUCGAUUCCGGUUGCGUCCA
tRNA6             UGUGUUCGAUCCACAGAAUUCGCA
#=GC SS_cons      <<<<_______>>>>>)))))))):
#=GC RF           gggGUUCgAuuCcccgugucggca
//
```

Figura 5.1: Exemplo do arquivo de entrada utilizado pelo Infernal.

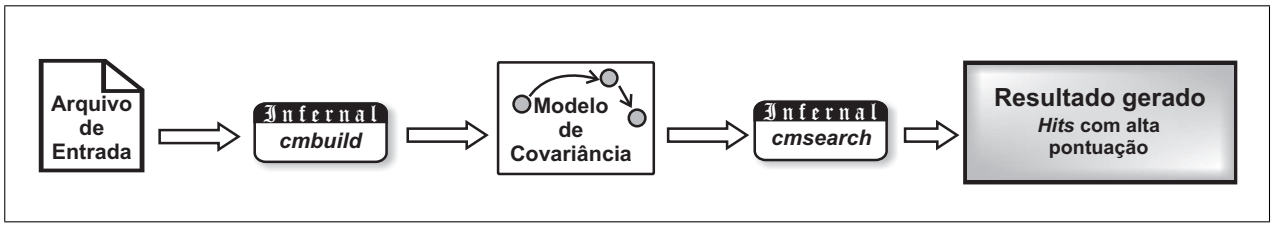

Figura 5.2: Busca utilizando o Infernal.

efetuar uma modificação na busca por ncRNAs homólogos usando a ferramenta Infernal, com a finalidade de melhorar o seu tempo de execução, como pode ser visto na Figura 5.3.

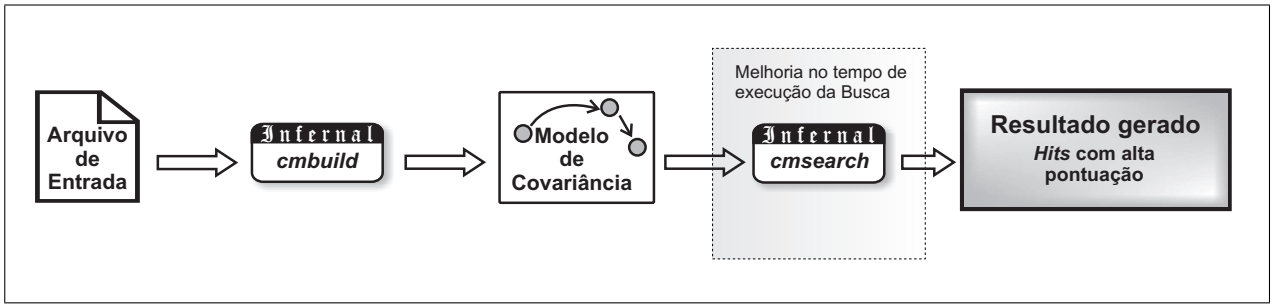

Figura 5.3: Melhoria na busca por ncRNAs homólogos utilizando o Infernal.

A idéia central focou-se na criação de uma pré-busca. Nesse sentido, o módulo auxiliar funcionaria como um filtro. Esse filtro deveria gerar um arquivo contendo uma lista de seqüências de RNAs com tamanho menor ou igual quando comparado à base de dados de entrada do Infernal.

Posteriormente, o Infernal utilizaria a saída do filtro (base de dados filtrada) como a base de entrada para a busca utilizando o *cmsearch*. A Figura 5.4 mostra o protótipo inicial do modelo proposto.

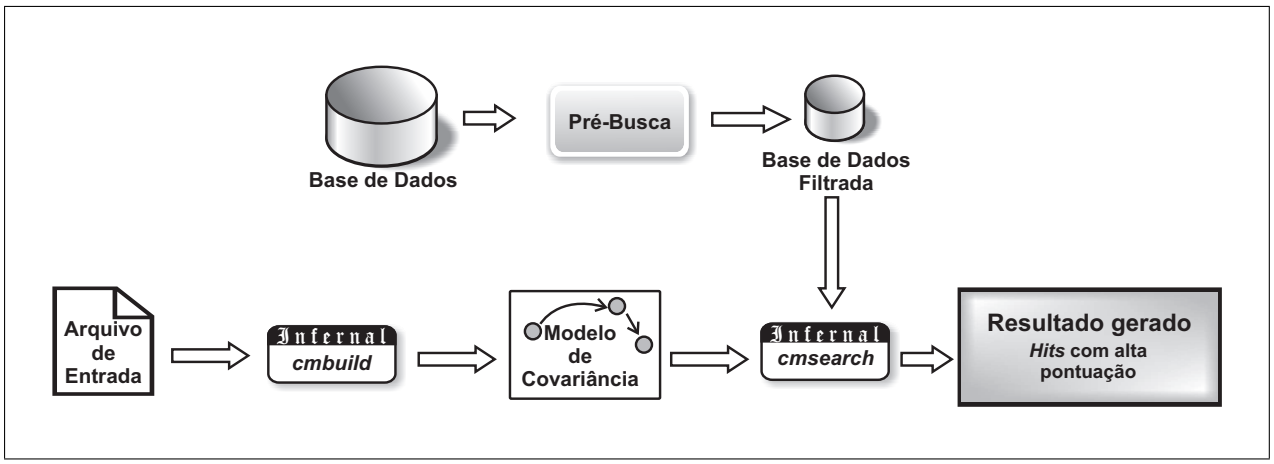

Figura 5.4: Protótipo do modelo de busca proposto utilizando a ferramenta Infernal.

Para a construção do filtro, como também é chamada a pré-busca construída, foi necessário analisar um meio para a comparação entre a seqüência de entrada e a base de dados.

Como descrito na Seção 3.4, os ncRNAs carecem de sinais estatísticos comuns em suas seqüências primárias que poderiam ser explorados por algoritmos de detecção (Washietl et al., 2005). Entretanto, muitos RNAs funcionais, como também são chamados os ncRNAs, possuem uma estrutura secundária definida. A conservação da estrutura secundária serve como evidência para identificação de ncRNAs, dessa forma, estudos comparativos parecem ser a abordagem mais promissora para detecção de RNAs funcionais. Washietl et al. (2005) comentam que estruturas secundárias de RNAs funcionais podem ser identificadas em alinhamentos múltiplos de seqüências com alta sensibilidade e alta especificidade.

Nesse contexto, optou-se por uma pré-busca que analisasse características extraídas da estrutura secundária e que fosse de tempo linear. Para alcançar esse objetivo, a base de dados precisou ser processada para que contivesse características extraídas de cada seqüência contida nela. Para isso, o modelo proposto é constituído de um passo chamado pré-processamento. É necessário que esse passo seja efetuado uma única vez para cada base de dados, pois ela poderá ser reutilizada em outras buscas. Nessa etapa a base de dados será adicionada de características e resultará em uma base de dados processada, como ilustra a Figura 5.5.

Figura 5.5: Pré-Processamento.

A fim de obter uma base de dados pré-processada foi desenvolvido um programa em linguagem C utilizando as bibliotecas do pacote Viena[2]. A versão utilizada para a criação desse programa foi a 1.6 do Viena.

O pacote Viena consiste de uma biblioteca de código C incluindo programa para a predição da estrutura secundária do RNA. A predição da estrutura secundária do RNA através da minimização de energia é a mais utilizada pelo Viena (Zuker e Stiegler, 1981).

No *website* do Viena são descritas algumas das técnicas disponíveis no pacote, como:

- ***RNAfold***. Predição da estrutura secundária e pareamento de bases baseado na energia mínima livre.
- ***RNAeval***. Avaliação da energia de uma estrutura secundaria.
- ***RNAdistance***. Cálculo da distância entre duas estruturas secundárias.
- ***RNApdist***. Comparação das probabilidades de pareamentos de bases.
- ***RNAplot***. Desenho da estrutura secundária.
- ***RNAalifold***. Predição da estrutura consenso através de seqüências alinhadas.

A biblioteca utilizada nesse trabalho baseia-se na técnica da energia mínima livre para predição da estrutura secundária de RNAs, gerando uma única estrutura ótima. O *RNAfold* lê uma seqüência de RNA e calcula sua energia mínima livre (mfe, do *inglês minimum free energy*) e imprime como saída a estrutura com mfe. As seqüências são lidas em um formato de texto simples, onde cada seqüência ocupa uma única linha.

A partir da predição da estrutura secundária que faz uso do cálculo da energia mínima livre foi criado o programa que efetua o pré-processamento da base de dados. O funcionamento desse programa, bem como as características extraídas da estrutura secundária gerada são descritas na Seção 5.3.1.

Para que fosse efetuada a comparação entre o arquivo de entrada, o mesmo utilizado pelo Infernal, e a base de dados processada foi necessária a criação de um programa que tratasse também o arquivo de entrada. Para tanto, no modelo proposto foi criada uma etapa de extração de características da entrada. Foi desenvolvido um programa em linguagem C que seleciona a estrutura secundária (como exemplo as linhas 10 e 19 da Figura 5.1), extrai suas características e salva em um novo arquivo.

Na Figura 5.6 é ilustrado o passo de extração de características da entrada. O funcionamento do programa desenvolvido para extração de características do arquivo de entrada, bem como as características extraídas são descritos na Seção 5.3.2.

[2]O código fonte do pacote e sua documentação estão disponíveis em `http://www.tbi.univie.ac.at/~ivo/RNA/`.

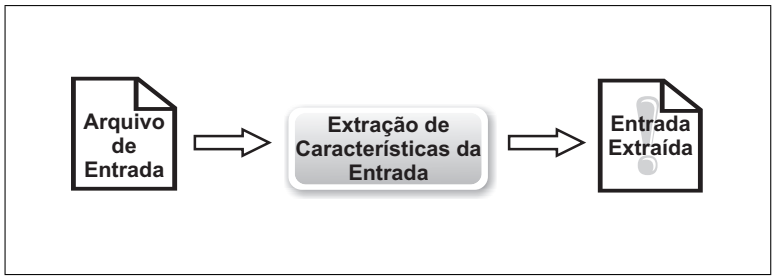

Figura 5.6: Extração de características do arquivo de entrada.

Por fim, foi desenvolvido o programa relacionado com a pré-busca. Esse programa foi desenvolvido em linguagem Perl. Essa linguagem é muito utilizada na bioinformática devido ao seu ponto forte em manipular caracteres e que, além disso, possui módulos específicos para aplicações em bioinformática.

Na Figura 5.7 tem-se o fluxo completo do modelo proposto para detecção de ncRNAs homólogos contra um banco de seqüências. O funcionamento do filtro desenvolvido para coletar apenas seqüências que tenham alguma similaridade com o arquivo de entrada é descrito na Seção 5.3.3.

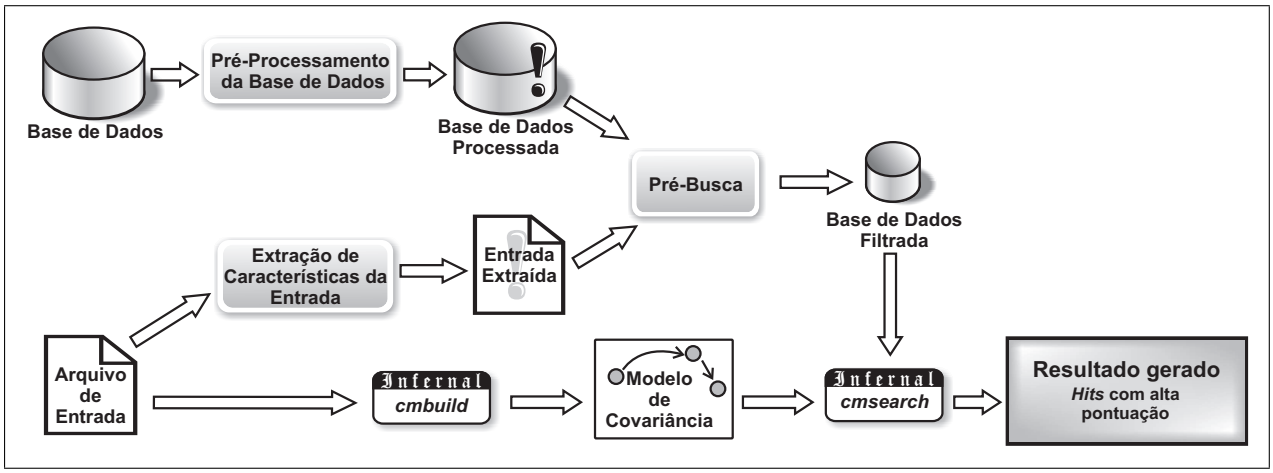

Figura 5.7: Modelo completo da busca incluindo o filtro

Como pode ser visto na Figura 5.7, primeiramente, a base de dados sofre um pré-processamento, conforme é descrito na Seção 5.3.1. Com isso, a base de dados processada irá conter a estrutura secundária das seqüências que a compõem, bem como caractéristicas que foram extraídas de cada seqüência. Adicionalmente, é necessário efetuar a extração de características do arquivo de entrada (Seção 5.3.2). O arquivo de entrada que foi extraído servirá como entrada no filtro (ou pré-busca) da base de dados processada. O filtro irá comparar características encontradas no arquivo de entrada extraído com as encontradas nas seqüências que compõem a base de dados processada e, com isso, irá selecionar as seqüências consideradas próximas ou menos distantes (Seção 5.3.3). Essas seqüências selecionadas são guardadas em um arquivo de base de dados filtrada.

Para finalizar a busca por ncRNAs homólogos, será construído o modelo de covariância utilizando *cmbuild* a partir do arquivo de entrada. Em seguida, com o arquivo contendo o modelo

de covariância, será possível utilizar o *cmsearch* para efetuar a busca na base de dados filtrada por RNAs homólogos.

O modelo apresentado para busca de ncRNAs homólogos foi projetado para ser utilizado em plataformas UNIX.

Nas sessões seguintes são descritas as etapas criadas no modelo proposto para busca de ncRNAs: pré-processamento, extração de características do arquivo de entrada e pré-busca.

5.3.1 Pré-Processamento

A fase de pré-processamento consiste em transformar uma base de dados em uma base de dados processada e que posteriormente será utilizada pela pré-busca (Seção 5.3.3).

Para esse fim, foi desenvolvido um programa em linguagem C. O programa recebe dois parâmetros por meio da linha de comando: o nome do arquivo com a base de dados de entrada e o nome do arquivo em que a base de dados processada será gravada.

A base de dados de entrada para o programa desenvolvido é semelhante ao formato FASTA exceto que seqüências longas não devem ser interrompidas por quebras de linha (Figura 5.8).

```
1  >tRNA01.
2  AGGAGCGACAGGUAGUCGCGGCUGCUAUGACACAGCAGUUAAGAGGGGUUCAAAUCCCCCCCCGAACCGGAGGGUUAUCCGGCCCA
3  >tRNA02.
4  AGGUGAGAGUAGUUUCUCUCGGUCAUCAUCACACGAUGAUCCUGUGGGGUGCAAAUCCCCCCCCUUAACUUGAGGGAAAUCAAGCCC
5  >tRNA03.
6  GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC
7  >tRNA04.
8  AGAUGCGACGAGGUGUCGCCGUUCAUAAGACACAUGGACGCUGUGGGGUGCAAAUCCCCCCCCUAACUUGAGGGAAAUCAAGCCC
9  >tRNA05.
10 GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC
11 >tRNA06.
12 GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC
```

Figura 5.8: Estrutura do arquivo de entrada para que seja efetuado o pré-processamento.

O programa irá processar cada seqüência do arquivo de entrada e gravar um registro contendo a descrição e a seqüência que já eram encontradas na base de dados de entrada, a estrutura secundária gerada pelo pacote Viena e as características extraídas.

Como ilustra a Figura 5.9, o formato do arquivo de saída é similar ao formato FASTA. A partir da estrutura secundária gerada são extraídas as características: quantidade de subcadeias, quantidade de pareamentos, quantidade de bases não-pareadas e vetor de níveis. Cada uma dessas características é descrita em detalhes a seguir.

Caso o pré-processamento extraia todas as características citadas acima, o arquivo de base de dados processada é gerado no seguinte formato:

<Descrição> <Seqüência> <Estrutura Secundária> <Quantidade de Subcadeias> <Lista de Subcadeias> <Quantidade de Pareamentos> <Quantidade de Bases Não-Pareadas> <Função do Vetor de Níveis> <Quantidade de Níveis> <Vetor de Níveis>

onde a descrição e a seqüência são os mesmos conteúdos encontrados na base de dados de entrada; a estrutura secundária é a estrutura gerada utilizando a predição da estrutura secundária baseada na energia mínima livre do Viena; a quantidade de subcadeias, lista de subcadeias, quantidade de pareamentos, quantidade de bases não-pareadas, função do vetor de níveis, quantidade de níveis e vetor de níveis estão relacionados às características definidas acima.

Um exemplo de um arquivo de base de dados processada a partir do arquivo da Figura 5.8 é mostrado na Figura 5.9.

```
1  >tRNA01.
   AGGAGCGACAGGUAGUCGCGGCUGCUAUGACACAGCAGUUAAGAGGGGUUCAAAUCCCCCCCCGAACCGGAGGGUUAUCCGGCCCA..
   ..(((((.....))))).((((((.......)))))).((.((((........)))).)).(((((.....))))).....
   4   (((((.....))))) ((((((.......)))))) (.((((........)))).)    (((((.....)))))
   21  16  66  6   4   4   4   4   4   1
2  >tRNA02.
   AGGUGAGAGUAGUUUCUCUCGGUCAUCAUCACACGAUGAUCCUGUGGGGGUGCAAAUCCCCCCCCUUAACUUGAGGGAAAUCAAGCCC
   .(((..((....(..(((..(((((((.......)))))))....(((.......))))..........))))..)..)))..)))..
   1
   (((..((....(..(((..(((((((.......)))))))....(((.......))))..........))))..)..)))..)))
   20 3   182 16  1   1   1   1   1   1   1   1   1   2   2   2   2   1   1   1
3  >tRNA03.
   GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC...
   .((((((.....))))))((((((.......))))))......((((((..((((.........)))).))))))........
   3   ((((((.....))))))   ((((((.......)))))) ((((((..((((.........)))).))))))
   22   17  97  10  3   3   3   3   3   3   1   1   1   1
4  >tRNA04.
   AGAUGCGACGAGGUGUCGCCGUUCAUAAGACACAUGGACGCUGUGGGGGUGCAAAUCCCCCCCCUAACUUGAGGGAAAUCAAGCCC...
   .(((.(.(.(((((..........)))))).).).)))...(((((....))))).((((......)))).............
   3   (((.(.(.(((((..........)))))).).).)))   (((((....)))))  ((((......))))
   19   20  80  10  3   3   3   3   2   1   1   1   1   1
5  >tRNA05.
   GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC...
   .((((((.....))))))((((((.......))))))......((((((..((((.........)))).))))))........
   3   ((((((.....))))))   ((((((.......)))))) ((((((..((((.........)))).))))))    22
   17  97  10  3   3   3   3   3   3   1   1   1   1
6  >tRNA06.
   GAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACC...
   .((((((.....))))))((((((.......))))))......((((((..((((.........)))).))))))........
   3   ((((((.....))))))   ((((((.......)))))) ((((((..((((.........)))).))))))
   22   17  97  10  3   3   3   3   3   3   1   1   1   1
```

Figura 5.9: Exemplo de estrutura do arquivo de saída do pré-processamento, ou seja, a base de dados processada, contendo todas características extraídas.

Quantidade de Subcadeias

Nesse trabalho uma subcadeia é definida como uma seqüência de um ou mais caracteres consecutivos em uma cadeia que representa a estrutura secundária de uma molécula. Uma subcadeia deve iniciar-se com um pareamento e não pode conter outras subcadeias. A quantidade de subcadeias é definida com o número de subcadeias encontrada na estrutura.

Por exemplo, na cadeia . . (. ((.) .)) . ((.) .) . que representa uma estrutura secundária no formato estocolmo são encontradas duas subcadeias: (. ((.) .)) e ((.) .) .

Quantidade de Pareamentos

A estrutura secundaria do RNA é em grande parte gerada por pareamentos de bases nas regiões complementares de sua fita simples, que dobra-se gerando estruturas tridimensionais. A quantidade de pareamentos é definida como a quantidade de bases pareadas na estrutura secundária de uma molécula.

Por exemplo, na cadeia . . (. ((.) .)) . ((.) .) . são encontradas cinco pares de bases pareadas. Para obter o número de bases pareadas, basta contar o número de parênteses abertos, que deverá ser igual ao número de parênteses fechados.

Quantidade de Bases Não-Pareadas

Uma base não pareada é uma base única ou resíduo que é representada, por exemplo, pelo símbolo ponto no formato estocolmo. A quantidade de bases não-pareadas é definida como a quantidade de bases que não estão pareadas na estrutura secundária.

Para o exemplo da cadeia . . (. ((.) .)) . ((.) .) . são encontradas nove bases não-pareadas. Para obter o número de bases não-pareadas, basta contar o número de pontos que contém na cadeia que representa a estrutura secundária.

A soma da quantidade de bases pareadas com a quantidade de bases não-pareadas é igual ao tamanho total da seqüência.

Vetor de Níveis e sua Função

O vetor de níveis é um vetor definido para a seqüência representante da estrutura secundária e que irá conter a quantidade de bases por nível. Um pareamento externo a todos os outros é definido como nível um, já para um pareamento interno o nível é maior que um e igual ao número de pareamentos que são externos a eles.

No exemplo da cadeia . . (. ((.) .)) . ((.) .) . , são encontrados três níveis de pareamentos: dois pareamentos são de nível um, dois de nível dois e um de nível três.

Uma função foi definida para calcular a soma ponderada dos níveis de uma estrutura secundária. Supondo um vetor de níveis V, que possui N níveis e que a quantidade em cada nível i é definida por Q_i, a função para o cálculo do vetor de níveis é dada por:

$$\sum_{i=1}^{N} i \times Q_i$$

5.3.2 Extração de Características da Entrada

A fase de extração de características consiste em extrair características do arquivo de entrada e gerar um novo arquivo que será utilizado na pré-busca (Seção 5.3.3). Para facilitar a entrada do usuário, manteve-se o mesmo arquivo de entrada utilizando pela ferramenta Infernal.

Para isso, foi criado um programa em linguagem C que efetua a extração do arquivo de entrada. Esse programa lê o arquivo de entrada e extrai suas características, gerando um arquivo que contém tanto a estrutura secundária quanto as características extraídas.

Se o programa de extração processar e gravar todas as características descritas na Seção 5.3.1, o arquivo de saída gerado segue o seguinte formato:

<Estrutura Secundária> <Quantidade de Subcadeias> <Lista de Subcadeias> <Quantidade de Pareamentos> <Quantidade de Bases Não-Pareadas> <Função do Vetor de Níveis> <Quantidade de Níveis> <Vetor de Níveis>

Um exemplo de um arquivo de entrada extraído a partir do arquivo da Figura 5.1 é mostrado na Figura 5.10. Na figura que contém o arquivo de entrada, utilizamos diretamente a informação da estrutura secundária consenso, na linha 10 continuando na linha 19. O arquivo gerado, conforme visto na segunda figura, possui, além da estrutura secundária consenso, as características extraídas dessa estrutura.

```
1  .....(((((......)))))..((((((.......))))))......((.(((.......))))........((((.......)
   ))))....   4   (((((......)))))    ((((((.......)))))) ((.(((.......))))
   (((((.......)))))   20  25  61  6   4   4   4   4   3   1
```

Figura 5.10: Exemplo de estrutura do arquivo de saída da extração do arquivo de entrada.

5.3.3 Pré-Busca

As características descritas na Seção 5.3.1 que estão contidas tando na base de dados processada quanto no arquivo de entrada extraído são utilizados nessa fase. A pré-busca funciona como um filtro, ela irá selecionar seqüências da base de dados processada que sejam aprovadas pelo filtro definido e irá salvá-las em um arquivo contendo a base de dados filtrada (Figura 5.11).

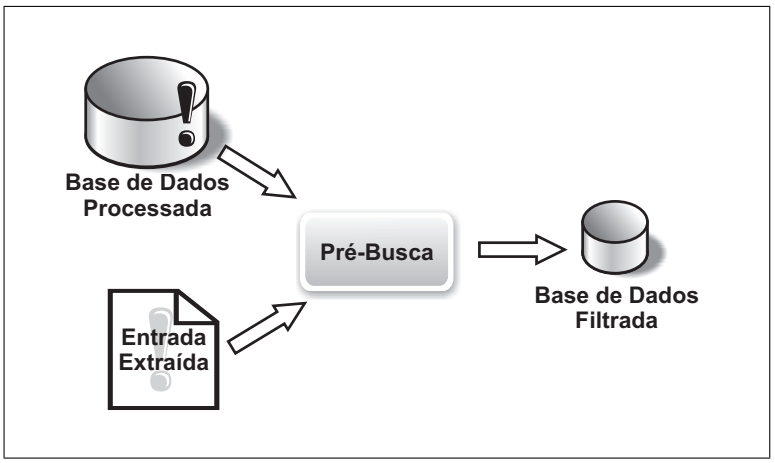

Figura 5.11: Funcionamento do filtro tendo como entrada a base de dados processada e o arquivo de entrada extraído.

O arquivo contendo a base de dados filtrada servirá como entrada para o programa de busca $cmsearch$ do Infernal, conforme descrito na Seção 5.3.

Para filtrar a base de dados processada foi desenvolvido um programa em linguagem Perl que recebe como parâmetros de entrada a base de dados processada e o arquivo de entrada com as características extraídas e tem como saída a base de dados filtrada.

Inicialmente o programa lê o arquivo de entrada extraído e cada seqüência da base de dados processada. Para cada característica vista na Seção 5.3.1 e que será utilizada como critério de filtragem, é verificado se seu valor está dentro de um intervalo aceito. Para a definição do intervalo aceito, inicialmente é definido um valor para o erro. Esse erro pode variar entre 0 (zero) e 1 (um). O intervalo de aceitação do filtro é calculado utilizando o erro definido. Quanto maior o valor do erro menos sensível será o filtro.

Para melhor compreensão do funcionamento do filtro, a cadeia . . (. ((.) .)) . ((.) .) . será utilizada como exemplo. Nessa cadeia são encontradas cinco bases pareadas. Supondo que o erro tenha sido definido em 0,4, seqüências que tenham entre três e sete pares de bases pareadas são selecionadas para compor a base de dados filtrada.

Seguindo essa idéia, foram criados filtros para as seguintes características: quantidade de subcadeias, quantidade de pareamentos, quantidade de bases não-pareadas, função do vetor de níveis, quantidade de níveis e vetor de níveis.

Na Seção 5.4 são discutidos os experimentos realizados e seus resultados utilizando esses filtros de modo independente e quando combinados.

5.4 Experimentos e Resultados

Experimentos foram realizados para avaliar o modelo proposto. Para cada filtro descrito na Seção 5.3.3 foram feitos experimentos isolados e combinando os filtros.

Para a realização dos experimentos, primeiramente foi definida uma base de dados na qual seria efetuada a busca por ncRNAs homólogos (Seção 5.4.1). Em seguida, definiu-se o arquivo de entrada, conforme descrito na Seção 5.4.2. A fim de avaliar os resultados obtidos pelos filtros, definiu-se uma função de avaliação para medir o desempenho dos filtros (Seção 5.4.3). E, por fim, são descritos os experimentos realizados (Seção 5.4.4).

5.4.1 Base de Dados

Para a realização dos experimentos foi necessária a criação de uma base de dados de seqüências. Definiu-se que a base de dados utilizada deveria conter seqüências de diferentes classes de ncRNAs, RNAs codificantes e seqüências desconhecidas. Foram coletadas 879 seqüências de RNAs para formar a base de dados.

Seqüências de ncRNAs

Foram selecionadas seqüências de diferentes classes de ncRNAs do banco de dados Rfam (Seção 4.3). O Rfam é um repositório que contém um conjunto de dados de diferentes famílias de ncRNAs (Griffiths-Jones et al., 2003, 2005).

Para compor a base de dados foram coletados um total 300 seqüências de diferentes classes de ncRNAs. Na Tabela 5.1 pode ser vista a relação de seqüências selecionadas e sua quantidade.

ncRNA	Total de Seqüências Selecionadas
miRNA	15
tRNA	50
snoRNA	50
Histone	50
scaRNA	55
Rnase	80

Tabela 5.1: Tabela mostrando as classes de ncRNAs e as respectivas quantidades de seqüências que compõem a base de dados, em um total de 300 seqüências.

RNAs Codificantes

As seqüências que representam os RNAs codificantes foram selecionados de uma base de ESTs (do inglês, *Expressed Sequences Tags*). Os ESTs representam regiões do DNA que irão codificar proteínas (genes). Para compor a base de dados foram selecionadas 379 seqüências consenso de contigs montados pelo CAP3 a partir das seqüências de ESTs da cana de açúcar originadas do projeto SUCEST, depois de filtradas para remover artefatos e seqüências de tRNA e rRNA (Vettore et al., 2003).

Seqüências Desconhecidas

A fim de simular seqüências desconhecidas foram geradas seqüências de RNAs baseando-se no trabalho de Schattner (2002), conforme descrito na Seção 4.2.13.

A geração foi feita usando dados da parte A da Tabela 4.3 que fornece a média do percentual de $G + C$, $G - C$ e $A - T$. Para tal, foi implementado um programa em linguagem C que, para cada seqüência gerada de uma espécie, respeita o percentual de $G + C$, $G - C$ e $A - T$ com seu respectivo desvio padrão.

Na Tabela 5.2 pode ser vista a relação da quantidade de seqüências de diferentes tamanhos geradas baseadas em cada espécie vista na Tabela 4.3.

	Tamanhos das Seqüências Gerada	Total de Seqüências Gerada
M. jannaschii	15, 20, 40, 60, 100	50
Plasmodium	15, 20, 40, 60, 100	50
C.elegans	15, 20, 40, 60, 100	50
H.sapiens	15, 20, 40, 60, 100	50

Tabela 5.2: Tabela mostrando os tamanhos gerados e a quantidade total para cada espécie. Para cada espécie foram geradas seqüências de tamanho 15, 20, 40, 60 e 100, em um total de 50 seqüências igualmente distribuídas. A base de dados contém um total de 200 seqüências consideradas desconhecidas.

5.4.2 Arquivo de Entrada

Para os experimentos realizados, o arquivo de entrada é formado por cinco seqüências de tRNAs no formato estocolmo encontradas também no banco de dados Rfam (Seção 4.3).

A Figura 5.1, utlizada para mostrar o formato estocolmo, também representa o arquivo de entrada que foi utilizado nos experimentos.

5.4.3 Função de Avaliação

Para facilitar a comparação entre os filtros e a avaliação de seu resultado é necessária uma forma de medir seu desempenho. Nesse sentido, foi criada uma função de avaliação do filtro.

Para a definição dessa função, partiu-se do princípio de que o filtro ideal é aquele em que sua saída é idêntica à saída utilizando o Infernal, que será chamada de saída esperada. As seqüências que fazem parte da saída esperada serão chamadas de seqüências esperadas. Nesse sentido, a função de avaliação teria que, de alguma forma, comparar a saída do filtro à saída esperada.

Deste ponto em diante, o termo descartada será usada para designar uma seqüência que não passou pelo filtro e o termo não-descartada será usado para designar uma seqüência que passou pelo filtro. Assim, as seqüências não-decartadas são aquelas que farão parte da base de dados filtrada.

Seja T o conjunto formado pelas seqüências que compõem a base de dados e seja I o conjunto formado pelo resultado obtido ao efetuar uma busca por ncRNAs utilizando puramente o Infernal. Seja F o conjunto formado pelas seqüências da base de dados filtrada resultante da aplicação do filtro à base composta por T. O ideal seria então, que F fosse igual a I.

Porém existem outras situações que devem ser avaliadas. O resultado do filtro poderá conter outras seqüências além do resultado esperado. Nesse caso, I será um subconjunto de F (Figura 5.12), ou seja, $I \subset F$ e $|F| > |I|$.

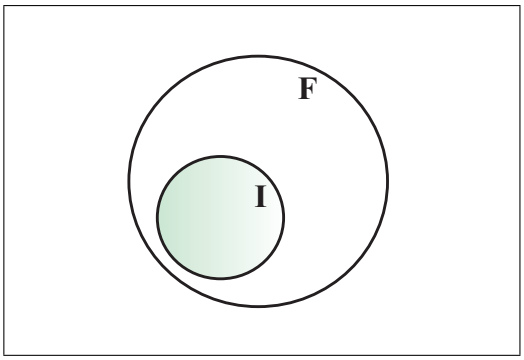

Figura 5.12: O conjunto de seqüências encontradas pelo Infernal (I) está contido no conjunto de seqüências não-descartadas pelo filtro (F).

Nessa situação, a pontuação do filtro deve ser penalizado por não ter barrado as seqüências extras. Essa penalização é feita sobre a quantidade de seqüências da base de dados T, conforme a Equação 5.1,

$$\frac{|F - H|}{|T|} \tag{5.1}$$

onde H é o conjunto de seqüências esperadas não-descartadas pelo filtro , ou seja $H = I \cap F$. Logo $|F - H|$ é a quantidade de seqüências não-descartadas pelo filtro e que não foram selecionadas pelo Infernal. As seqüências do conjunto H serão chamadas de seqüências preservadas.

Outra situação que deve ser penalizada é o caso em que o resultado do filtro contenha apenas uma parte dos resultados esperados. Para esse caso, F será um subconjunto de I (Figura 5.13), ou seja, $I \supset F$ e $|F| < |I|$.

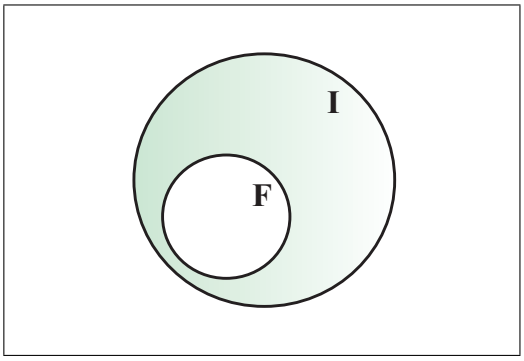

Figura 5.13: O conjunto de seqüências encontradas pelo Infernal (I) contém o conjunto de seqüências não-descartadas pelo filtro (F) e sua quantidade é maior que em F.

Nesse caso o filtro barrou seqüências que não deveria, logo, a penalização deve ter um peso ainda maior e é feita sobre a quantidade de seqüências esperadas I de acordo com a Equação 5.2,

$$\frac{|I - H|}{|I|} \tag{5.2}$$

onde $|I - H|$ é a quantidade de seqüências que o filtro descartou incorretamente.

Adicionalmente, o resultado do filtro F poderá conter tanto seqüências em comum com I, quanto seqüências distintas, como ilustrado na Figura 5.14. Nesse caso, $F \cap I \neq \emptyset$, $I \not\subset F$ e $F \not\subset I$.

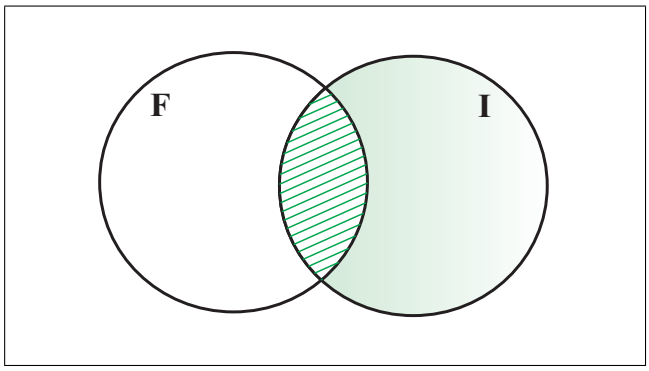

Figura 5.14: O conjunto de seqüências encontradas pelo Infernal (I) e o conjunto das seqüências não-descartadas pelo filtro (F) têm intersecção mas nenhum está contido no outro.

Nesse caso, o filtro selecionou algumas seqüências esperadas, outras não esperadas e, além disso, barrou seqüências que deviam ser selecionadas. Logo, deve ser penalizado de acordo com as Equações 5.1 e 5.2, de acordo com a Equação 5.3,

$$\frac{|F - I|}{|T|} + \frac{|I - H|}{|I|} \tag{5.3}$$

A partir das equações definidas acima, definiu-se a Equação 5.4 limitada a zero como função de avaliação do desempenho dos filtros.

$$1 - (\frac{|F - I|}{|T|} + \frac{|I - H|}{|I|}) \tag{5.4}$$

Para essa função de avaliação temos:

- Se F é igual a I então tem-se concordância máxima e o valor é 1.
- Se F e I são disjuntos então não tem-se concordância e o valor é 0.
- Se F contém e é maior que I, então F concorda com I, porém foi pouco exigente e e sofre penalização.

- Se F está contido e é menor que I, então F concorda com I, porém foi muito exigente e sofre penalização que será maior ou igual à penalização acima.

- Se F não contém nem está contido em I e $F \cap I \neq \emptyset$, então F concorda parcialmente com I e sofre as penalizações descritas nos dois itens anteriores.

5.4.4 Experimentos

Nessa seção são descritos os experimentos efetuados utilizando a base de dados formada por um total de 879 seqüências definida na Seção 5.4.1 e o arquivo de entrada definido na Seção 5.4.2. Para cada filtro descrito na Seção 5.3.1 foram feitos experimentos isolados e combinando-os.

Para aplicação da função de avaliação, primeiramente foi feita a busca de ncRNAs utilizando o Infernal. Um critério utilizado no Infernal para significância estatística é o *E-value*. O *E-value* é calculado a partir da pontuação, e quanto menor seu valor maior a probabilidade que a seqüência seja um ncRNA. Eddy (2009) explica que um *E-value* de 0,1, por exemplo, significa que há apenas 10% de chance de que haja falsos positivos e que confia nos resultados de buscas com *E-value* menores que 0,1 e examina manualmente os hits que tem um *E-value* ≤ 10.

Nesse sentido, foi feita uma busca na base de dados utilizando o Infernal com o $E \leq 10$. Na Tabela 5.3 são descritas as seqüências encontradas e seus respectivos *E-values*.

Descrição ncRNA	E-value	Descrição ncRNA	E-value	Descrição ncRNA	E-Value
tRNA 01	2,15E-07	tRNA 19	0,0001791	tRNA 37	0,0001382
tRNA 02	2,33E-05	tRNA 20	9,98E-02	tRNA 38	8,90E-06
tRNA 03	7,96E-08	tRNA 21	3,24E-02	tRNA 39	1,48E-02
tRNA 04	1,50E-04	tRNA 22	1,48E-02	tRNA 40	0,003074
tRNA 05	7,96E-08	tRNA 23	0,0002426	tRNA 41	3,34E-02
tRNA 06	7,96E-08	tRNA 24	7,90E-02	tRNA 42	0,2436
tRNA 07	4,49E-05	tRNA 25	0,0003211	tRNA 43	0,0002171
tRNA 08	1,23E-08	tRNA 26	0,0001618	tRNA 44	8,63E-03
tRNA 09	1,37E-05	tRNA 27	0,0001297	tRNA 45	0,5443
tRNA 10	7,96E-08	tRNA 28	8,90E-06	tRNA 46	0,0001618
tRNA 11	1,05E-05	tRNA 29	8,90E-06	tRNA 47	0,0001618
tRNA 12	7,96E-08	tRNA 30	0,1679	tRNA 48	3,34E-02
tRNA13	7,96E-08	tRNA 31	8,90E-06	tRNA 49	8,90E-06
tRNA 14	0,0008437	tRNA 32	0,0003372	tRNA 50	8,90E-06
tRNA 15	0,001222	tRNA 33	3,34E-02	EST 38	2,497
tRNA 16	0,0005811	tRNA 34	8,90E-06	EST 172	3,647
tRNA 17	3,34E-02	tRNA 35	3,34E-02	EST 221	4,399
tRNA 18	7,61E-02	tRNA 36	8,90E-06	EST 335	7,767

Tabela 5.3: Conjunto de seqüências retornadas na busca utilizando o Infernal e seus respectivos *E-values*.

Pode-se observar que foram encontradas 54 seqüências, dentre elas 50 seqüências de tRNAs previamente conhecidas e 4 seqüências de ESTs, que são falsos positivos. Essas seqüências resultantes foram utilizadas como forma de avaliação dos filtros criados, inclusive os falsos positivos, levando em consideração que se o Infernal selecionou essas seqüências existem semelhanças em suas estruturas secundárias e composição com a seqüência de entrada.

A seguir são descritos os experimentos utilizando os filtros.

Quantidade de Pareamentos

Foi feito o experimento utilizando unicamente o parâmetro de quantidade de pareamentos. Nesse experimento, variou-se o erro e foi utilizada a função de avaliação (Seção 5.4.3).

Na Tabela 5.4 são mostrados os resultados obtidos.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0,05	34	11	0,177537606
0,065	42	15	0,247061054
0,07	42	15	0,247061054
0,075	95	47	0,815762862
0,08	95	47	0,815762862
0,085	95	47	0,815762862
0,09	95	47	0,815762862
0,095	95	47	0,815762862
0,1	95	47	0,815762862
0,11	101	48	0,828593098
0,12	101	48	0,828593098
0,13	108	50	0,859941853
0,14	108	50	0,859941853
0,15	108	50	0,859941853
0,2	121	50	0,84515232
0,3	163	50	0,79737075
0,4	177	50	0,78144356
0,5	194	50	0,7621034
0,7	274	50	0,671090886
0,8	291	50	0,651750727
0,9	302	50	0,639236506
1	373	50	0,5584629

Tabela 5.4: Tabela mostrando os valores de erro usados no teste, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

Vale ressaltar que as 4 seqüências que o filtro descartou foram os falsos positivos encontrados pelo Infernal (ESTs 38, 172, 221 e 335 da Tabela 5.3). Destaca-se também que para um erro de 0,13 o filtro reduziu em aproximadamente oito vezes o tamanho da base de dados, tendo um desempenho de aproximadamente 0,85, o qual decai adiante. Adicionalmente, evidencia-se também um crescimento acentuado na variação do erro de 0,07 a 0,075.

Na Figura 5.15 é ilustrado um gráfico mostrando o comportamento da função de avaliação (pontuação) ao variar o erro.

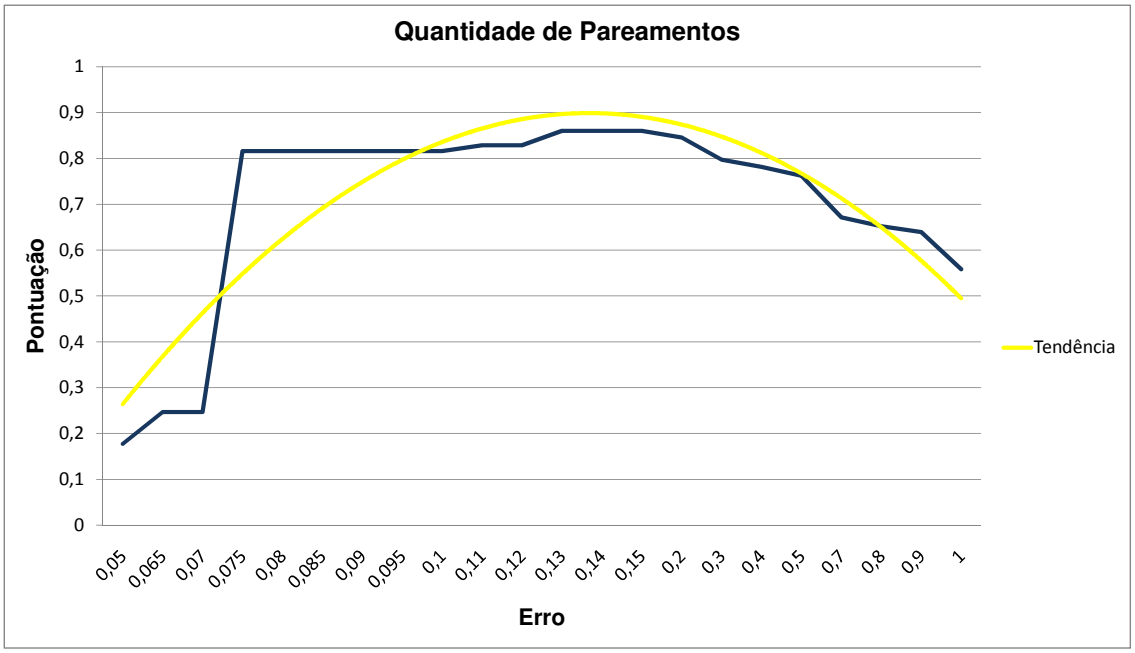

Figura 5.15: Gráfico demostrando o comportamento da função de avaliação ao variar o erro para o filtro utilizando unicamente quantidade de pareamentos como parâmetro.

Quantidade de Não-Pareados

Outro experimento foi efetuado utilizando unicamente o parâmetro de quantidade de não-pareados e utilizando a função de avaliação (Seção 5.4.3) à medida que varia-se o erro.

Na Tabela 5.5 são mostrados os resultados obtidos.

Nesse experimento o melhor desempenho ocorreu com um erro de 0,4, onde obteve-se uma redução de aproximadamente metade do tamanho da base de dados e os falsos positivos não foram encontrados, porém, para esse valor de erro, não foram encontrados 5 dos tRNAs homólogos.

Na Figura 5.16 é ilustrado um gráfico mostrando o comportamento da função de avaliação ao variar o erro pra um filtro que utiliza unicamente a quantidade de não-pareados como parâmetro.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0,1	93	6	0,012135002
0,2	192	8	0
0,3	298	28	0,211351283
0,4	432	45	0,393060296
0,5	523	46	0,309189736
0,6	623	47	0,215080268
0,7	730	52	0,191631905
0,75	751	53	0,187397295
0,8	784	53	0,149854633
0,81	802	53	0,129376817
0,82	804	53	0,127101504
0,83	804	53	0,127101504
0,84	804	53	0,127101504
0,85	840	54	0,105802048
0,9	863	54	0,07963595
0,95	868	54	0,073947668

Tabela 5.5: Tabela mostrando os valores de erro usados no teste, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

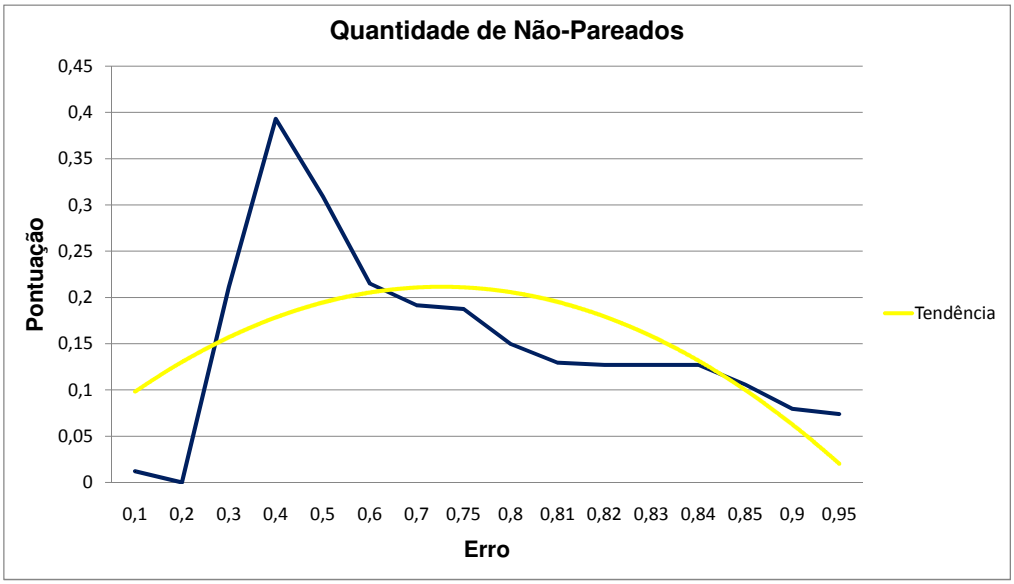

Figura 5.16: Gráfico demostrando o comportamento do desempenho filtro.

Quantidade de Subcadeias

Nesta Seção é descrito o experimento efetuado utilizando unicamente o parâmetro de quantidade de subcadeias. Para isso, variou-se o valor do erro e foi utilizada a função de avaliação (Seção 5.4.3) para medir seu desempenho.

Na Tabela 5.6 são mostrados os resultados obtidos.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0,01	159	45	0,703640501
0,1	159	45	0,703640501
0,2	165	46	0,716470737
0,3	440	49	0,462583744
0,4	441	49	0,461446088
0,5	441	49	0,461446088
0,51	810	54	0,139931741
0,6	810	54	0,139931741
0,7	810	54	0,139931741
0,8	879	54	0,061433447
0,9	879	54	0,061433447

Tabela 5.6: Tabela mostrando os valores de erro usados no teste, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

No experimento verifica-se que o maior desempenho é com um erro de 0,2 que barra 9 seqüências que deviam ser selecionadas. Ao aumentar o valor do erro, essas seqüências são selecionadas, porém muitas outras também, fazendo com que a quantidade de seqüências não-descartadas aumente substancialmente.

Na Figura 5.17 é mostrado o gráfico que reflete o desempenho ao variar o erro para o filtro que utiliza unicamente quantidade de subcadeias como parâmetro.

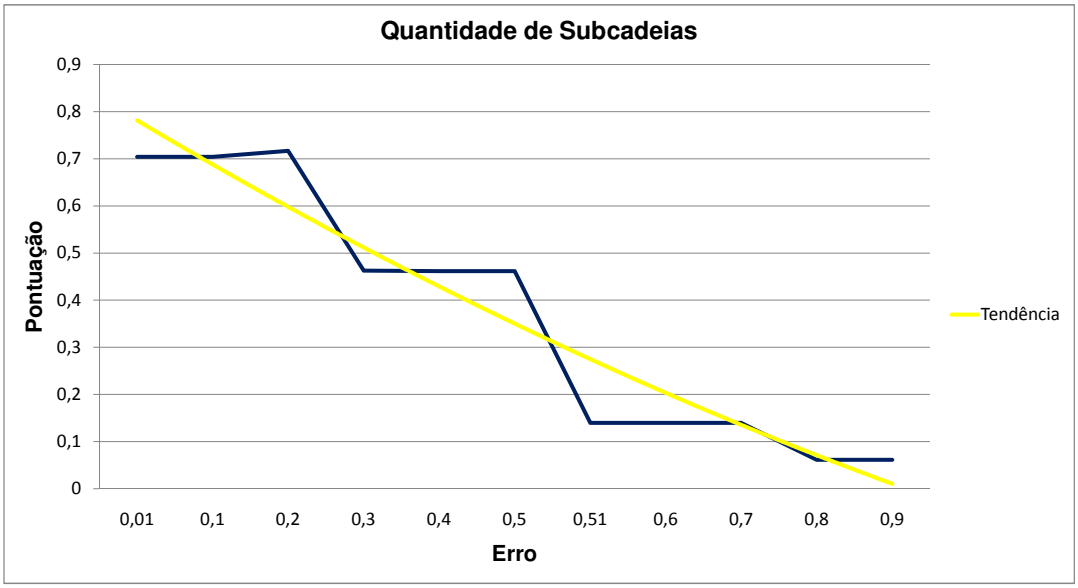

Figura 5.17: Gráfico mostrando o desempenho do filtro através da função de avaliação.

No Gráfico da Figura 5.17 fica evidenciado que para valores de erro maiores que 0,2, o desempenho decai substancialmente.

Função do Vetor de Nível

Foi efetuado outro experimento utilizando unicamente o parâmetro do valor da função do vetor de nível, também descrito conforme descrito na Seção 5.3.1. Nesse experimento também foi utilizada a função de avaliação (Seção 5.4.3) como medida para o desempenho.

Os resultados são mostrados na Tabela 5.7.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0,1	22	7	0,112564783
0,15	24	7	0,11028947
0,2	46	28	0,498040703
0,25	52	28	0,491214764
0,3	58	30	0,523701176
0,35	68	31	0,531980786
0,4	76	31	0,522879535
0,45	77	32	0,541398053
0,5	81	32	0,536847428
0,55	98	32	0,517507268
0,6	106	40	0,665655417
0,65	183	40	0,578055872
0,7	186	40	0,574642902
0,75	232	40	0,522310707
0,8	234	40	0,520035394
0,85	248	43	0,563076729
0,9	258	43	0,551700164
0,95	259	43	0,550562508
1	299	45	0,544368601
1,5	276	45	0,570534699
2	294	49	0,628681583
2,4	312	49	0,608203767
2,5	333	50	0,603969157

Tabela 5.7: Tabela mostrando os valores de erro usados no teste, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

A Figura 5.18 ilustra o gráfico com o comportamento da função de avaliação para o filtro que utiliza unicamente a quantidade de pareamentos como parâmetro ao variar o valor do erro.

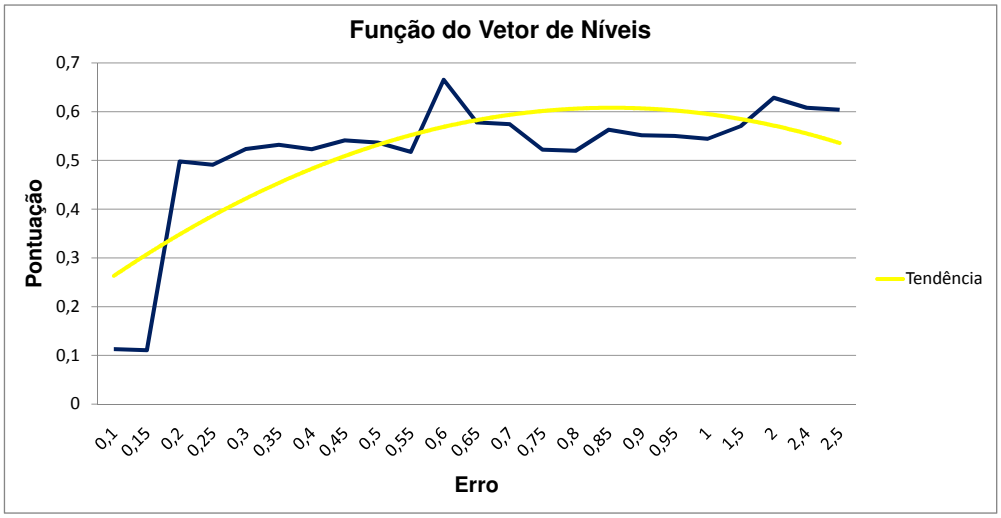

Figura 5.18: Gráfico mostrando o desempenho do filtro através da função de avaliação ao variar o valor do erro.

Para esse filtro pode-se notar que, com a variação do erro, a quantidade de seqüências encontradas cresceu mais gradativamente. No gráfico da Figura 5.18 evidencia-se que o melhor desempenho é para um erro de 0,6 e, ao aumentar o valor do erro, o desempenho irá variar, mas não alcançando um valor superior ou igual a 0,66. Para esse desempenho 14 seqüências esperadas não foram selecionadas.

Vale ressaltar que as 4 seqüências que o filtro não selecionou com um erro de 2,5 são os falsos positivos encontrados pelo Infernal (ESTs 38, 172, 221 e 335 da Tabela 5.3).

Vetor de Nível

Além do experimento da função de nível, foi efetuado outro experimento analisando o vetor de nível propriamente dito. Nesse experimento também foi utilizada a função de avaliação (Seção 5.4.3) como medida de desempenho. Entretanto, nesse experimento não houve uma variação do erro como nos casos anteriores. O valor do erro foi fixado em 0,2 e variou-se a quantidade máxima de níveis fora do intervalo requerido.

Na Tabela 5.8 são mostrados os resultados obtidos.

Pode-se observar que o melhor desempenho ocorreu para um erro em que a quantidade máxima de níveis fora do limite é 5. Para valores maiores que 5, observa-se que o desempenho decai gradativamente.

Pode ser visto na Figura 5.19 o gráfico que mostra o comportamento da função de avaliação para o filtro que utiliza unicamente o vetor de nível como parâmetro.

Quantidade Permitida de Níveis Fora do Limite	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0	1	1	0,018518519
1	23	23	0,425925926
2	29	28	0,517380862
3	32	31	0,572936418
4	34	32	0,59031728
5	97	40	0,675894324
6	215	40	0,541650866
7	221	40	0,534824927
8	233	43	0,580141575
9	245	43	0,566489698
10	254	43	0,55625079

Tabela 5.8: Tabela mostrando a quantidade permitida de níveis fora do limite, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

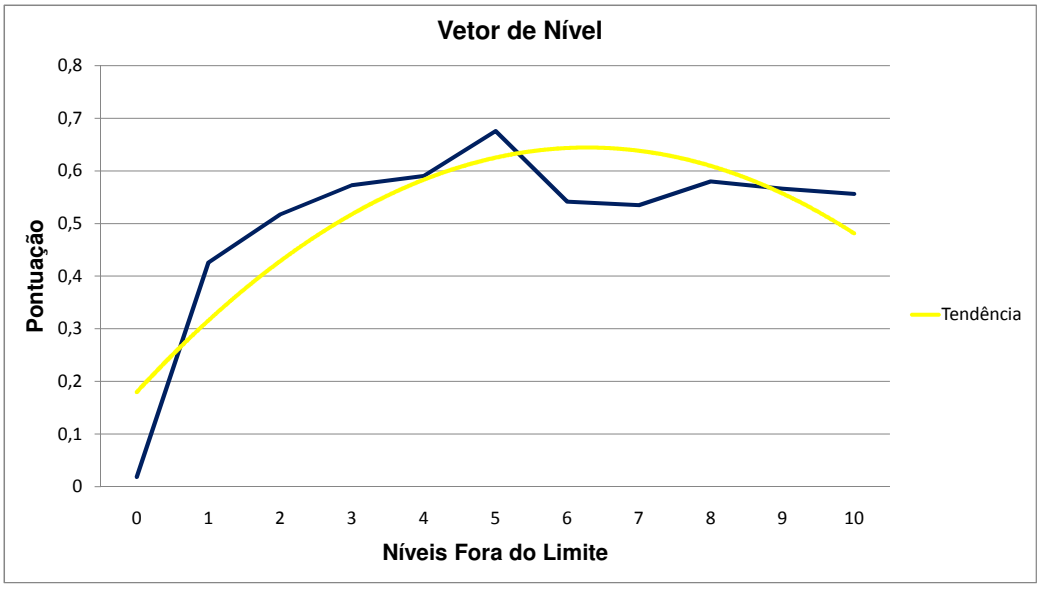

Figura 5.19: Gráfico representando desempenho do filtro que utiliza unicamente o vetor de níveis.

Votação Unânime

Além dos experimentos descritos acima foi feito um experimento combinando os parâmetros vetor de nível, quantidade de pareamentos, quantidade de não pareados, quantidade de subcadeias e o valor da função de níveis.

Para cada seqüência analisada o filtro leva em consideração a "opinião" dos parâmetros descritos acima, e a seqüência será descartada se e somente se todos os parâmetros a descartarem,

por isso é chamado de votação unânime. Nesse experimento também foi utilizada a função de avaliação descrita na Seção 5.4.3 como medida de desempenho.

Pode ser visto na Tabela 5.9 o desempenho apresentado por esse conjunto de filtros.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação
0,01	181	47	0,717924409
0,03	208	50	0,74617621
0,05	225	50	0,726836051
0,1	289	52	0,69333839
0,15	329	52	0,647832132
0,2	362	52	0,61028947
0,25	420	53	0,563961572
0,3	602	54	0,376564278
0,35	667	54	0,30261661
0,4	681	54	0,28668942
0,45	689	54	0,277588168
0,5	702	54	0,262798635

Tabela 5.9: Tabela mostrando os valores de erro usados no teste, a quantidade de seqüências não-descartadas pelo filtro (F), a quantidade de seqüências preservadas (H) e por fim, o resultado da função de avaliação para o erro.

A Figura 5.20 ilustra o gráfico com o comportamento da função de avaliação ao variar o erro.

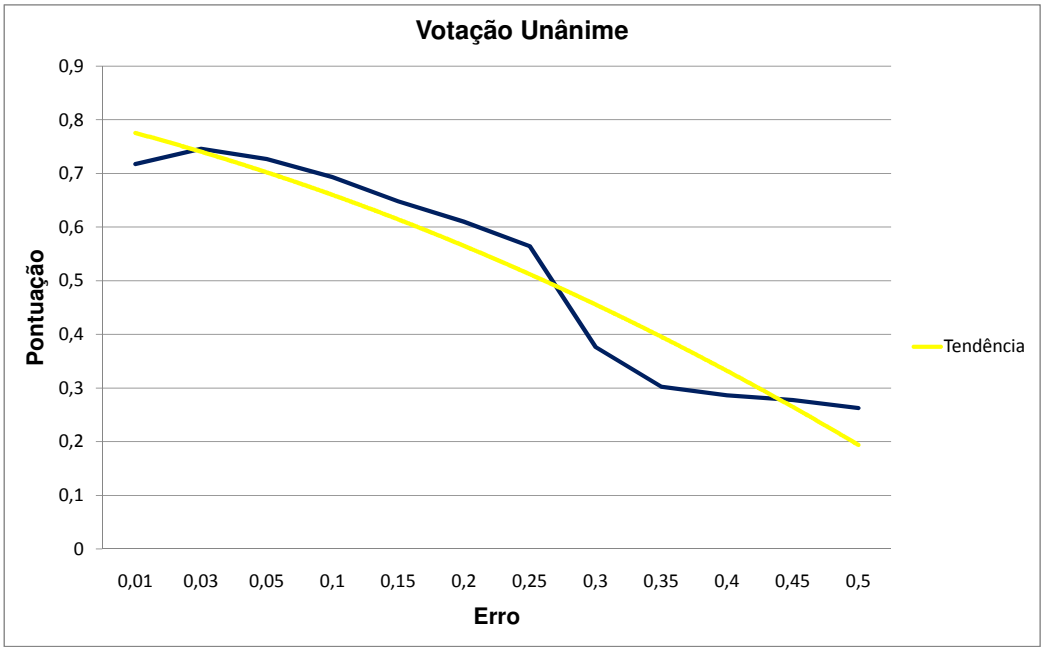

Figura 5.20: Gráfico representando desempenho do filtro que utiliza a votação unânime.

Para o valor de erro 0,1 as duas seqüências faltantes são as falso positivas encontradas pelo Infernal. Para 0,03 e 0,05 duas seqüências de tRNAs e duas seqüências falso positivas foram descartadas.

Na Seção seguinte são mostrados o tempo de execução gasto para a busca no Infernal e das etapas do modelo proposto e descrito na Seção 5.3.

5.4.5 Tempo de Execução

Como descrito na Seção 5.3 o objetivo deste trabalho é obter uma ferramenta com menor tempo de execução para a comparação de uma seqüência de RNA não-codificante contra um banco de seqüências quando comparado ao Infernal.

Para o experimento descrito neste Capítulo observou-se o tempo gasto para executar algumas etapas da busca. Nos experimentos foi utilizado um computador com processador Intel Core Duo 1.66 Ghz (667 MHz FSB, 2 MB L2 cache), memória DDR2 2Gb 667 Mhz, com sistema operacional GNU Linux, Ubuntu 8.0.4 [3].

Na Tabela 5.10 estão listados os tempos gastos para a execução da ferramenta Infernal e para cada etapa descrita na Seção 5.3.

Descrição	Tempo de Execução
Construção do Modelo de Covariância - *cmbuild*	0,183s
Busca Infernal - *cmsearch*	14m33s
Pré-Processamento	24m10s
Extração de Características da Entrada	0,002s
Pré-Busca	0,243s

Tabela 5.10: Tempo de execução do Infernal e de algumas fases do modelo proposto.

Na primeira linha pode-se ver que o tempo gasto para gerar o modelo de covariância utilizando *cmbuild* durou cerca de 0,2 segundos, na segunda linha tem-se o tempo utilizando unicamente o Infernal como ferramenta de busca na base de dados definida na Seção 5.4.1, 14 minutos e 33 segundos. Nas linhas seguintes estão relacionadas as etapas relacionadas ao modelo proposto (Seção 5.3): o pré-processamento (Seção 5.3.1) que durou cerca de 24 minutos e 10 segundos, a extração de características do arquivo de entrada (Seção 5.3.2) descrito na Seção 5.4.2, que levou 0,002 segundos; e a pré-busca (seção 5.3.3), na qual foi calculada uma média dos experimentos descritos na Seção 5.4.4, obtendo um média de 0,243 segundos.

[3]Comunidade do Ubuntu no Brasil `http://www.ubuntu-br.org/`.

A Tabela 5.11 reúne o tempo gasto para a busca utilizando o Infernal da base de bados filtrada a partir de um filtro que utiliza a quantidade de pareamento como único parâmetro, mostrado na Tabela 5.4.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação	Tempo de Execução Infernal
0,075	95	47	0,81576286	2m35s
0,11	101	48	0,8285931	2m40s
0,13	108	50	0,85994185	2m41s

Tabela 5.11: Tempo de execução na busca utilizando o Infernal tendo a base de dados filtrada a partir do filtro que possui a quantidade de pareamento com único parâmetro.

É mostrado também na Tabela 5.12 o tempo de execução da base de dados filtrada por votação unânime, visto na Tabela 5.9.

Erro	Seqüências Não-Descartadas (F)	Seqüências Preservadas (H)	Função de Avaliação	Tempo de Execução Infernal
0,01	181	47	0,717924409	4m50s
0,03	208	50	0,74617621	5m10s
0,05	225	50	0,726836051	5m30s
0,1	289	52	0,69333839	5m58s
0,15	329	52	0,647832132	6m30s
0,2	362	52	0,61028947	6m50s

Tabela 5.12: Tempo de execução na busca utilizando o Infernal tendo a base de dados filtrada a partir do filtro por votação unânime.

Pode-se observar que houve uma redução substancial nas buscas utilizando o Infernal nas bases de dados filtradas quando comparadas à busca na base de dados sem a presença do filtro.

5.5 Considerações Finais

Neste capítulo foi mostrado o modelo proposto para a busca de RNAs não codificantes, tendo como base a ferramenta Infernal. Para isso, essa ferramenta foi a apresentada, bem como o pacote Viena utilizado na implementação do modelo.

Foram também descritos os experimentos realizados para avaliar o modelo proposto. Para alguns dos experimentos descritos observou-se o tempo gasto para executar algumas etapas da busca e identificou-se as melhorias no seu tempo de execução.

CAPÍTULO

6

Conclusão

Nesta dissertação foi apresentado um modelo para a busca de RNAs não-codificantes. Para tal, foram inicialmente recuperados os principais conceitos da biologia molecular, com ênfase nos ncRNAs. Além desses conceitos, foram apresentadas pesquisas sobre a busca de ncRNAs. Identificou-se a possibilidade de melhoramento do tempo de execução da ferramenta Infernal. A melhoria foi feita com o apoio do pacote Viena, o que tornou possível a inclusão de novas etapas no mecanismo do Infernal, permitindo que tais buscas possam ser efetuadas em um tempo de execução menor. Foram definidas novas etapas na busca por ncRNAs homólogos: pré-processamento, extração de características da entrada e pré-busca.

O modelo proposto permite a reutilização da base de dados processada, processo mais custoso desse novo modelo. Neste trabalho também foram apresentados experimentos envolvendo a busca por homólogos de tRNAs. Após o pré-processamento da base de dados e da extração de características do arquivo de entrada utilizando os programas implementados, foi possível realizar experimentos utilizando filtros e suas combinações. Os resultados desses experimentos mostraram que é possível pré-selecionar as seqüências homólogas em tempo linear.

6.1 Contribuições

A principal contribuição deste trabalho é um modelo para busca de RNAs não-codificantes contra um banco de seqüências. O modelo permite a execução de uma pré-busca de complexidade linear. Essa pré-busca permite que a busca feita pelo Infernal, mais complexa, seja realizada sobre uma base de dados com tamanho reduzido quando comparada à base de dados inicial. Para o experimento quantidade de pareamentos, por exemplo, o filtro reduziu em mais de 80% o tamanho da

base de dados, sem descartar todas as seqüências esperadas e que são homólogas. Além disso, não selecionou seqüências esperadas falso-positivas. Essa melhoria fez com que o tempo de execução para a busca fosse reduzido em mais de 5 vezes quando comparado à busca utilizando apenas o Infernal. A pré-busca pode utilizar diferentes parâmetros disponíveis e suas combinações a fim de refinar esse filtro.

Uma grande vantagem do modelo proposto é o pequeno número de parâmetros livres do filtro. Embora o trabalho não preveja uma forma automática de fixar os parâmetros, o pequeno número de parâmetros favorece o uso de estratégias *ad-hoc* para a seleção de parâmetros.

6.2 Trabalhos Futuros

Como trabalho futuro é possível realizar mais experimentos, utilizando diferentes classes de RNAs não-codificantes. Para esses novos experimentos podem ser feitas análises com diferentes combinações de parâmetros para o filtro. Esses novos testes teriam a finalidade de avaliar a eficiência da abordagem proposta em situações mais complexas. Vale ressaltar que os experimentos realizados nesse trabalho de mestrado não são suficientes para garantir um desempenho similar do processo em qualquer situação.

Outros possíveis trabalhos futuros seriam a análise das intersecções entre os filtros como uma forma de melhorar o desempenho obtido ou a análise da utilização de filtros mais complexos. Na versão atual foram utilizados apenas filtros lineares. Entretanto, outros filtros podem ser idealizados usando combinações não-lineares dos valores de características e seu desempenho e eficiência analisados. Outras características podem ser extraídas das moléculas, eventualmente recorrendo a algoritmos mais elaborados, embora não tão rápidos. Visto que a descoberta de características mais complexas é uma atividade difícil e extensa, pode-se investigar novos processos e ferramentas que auxiliem na criação, melhoria e identificação de possíveis filtros.

Referências Bibliográficas

ALTSCHUL, S. F.; GISH, W.; MILLER, W.; MYERS, E. W.; LIPMAN, D. J. Basic local alignment search tool. *J. Mol. Biol*, v. 215, p. 403–410, 1990.

ALTSCHUL, S. F.; MADDEN, T. L.; SCHAFFER, A. A.; ET AL. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, v. 25, p. 3389–3402, 1997.

BADGER, J. H.; OLSEN, G. J. Critica: Coding region identification tool invoking comparative analysis. *Mol Biol Evol*, v. 16, p. 512–524, 1999.

BARNECHE, F.; GASPIN, C.; GUYOT, R.; ECHEVERRIA, M. Identification of 66 box c/d snornas in arabidopsis thaliana: extensive gene duplications generated multiple isoforms predicting new ribosomal rna 2'-o-methylation sites. *J. Mol. Biol.*, v. 311, p. 57–73, 2001.

BERNARDO, D.; DOWN, T.; HUBBARD, T. ddbrna: detection of conserved secondary structures in multiple alignments. *BIOINFORMATICS*, v. 19, n. 13, p. 1606–1611, 2003.

BLATTNER, F. R.; PLUNKETT, G.; BLOCH, C. A.; PERNA, N. T.; BURLAND, V.; RILEY, M. The complete genome sequence of escherichia coli k-12. *Science*, v. 277, p. 1453–1474, 1997.

CASTRIGNANO, T.; CANALI, A.; GRILLO, G.; LIUNI, S.; MIGNONE, F.; PESOLE, G. Cst-miner: a web tool for the identification of coding and noncoding conserved sequence tags through cross-species genome comparison. *Nucleic Acids Res.*, v. 32 (Web Server issue), p. W624–W627, 2004.

CASTRIGNANO, T.; MEO, P. D.; GRILLO, G.; LIUNI, S.; MIGNONE, F.; TALAMO, I. G.; PESOLE, G. Genominer: a tool for genome-wide search of coding and non-coding conserved sequence tags. *BIOINFORMATICS*, v. 22, n. 4, p. 497–499, 2005.

COVENTRY, A.; KLEITMAN, D. J.; BERGER, B. Msari: Multiple sequence alignments for statistical detection of rna secondary structure. *PNAS*, v. 101, n. 33, p. 12102–12107, 2004.

CRICK, F. On protein synthesis. In: *ICSE '02: Proceedings of the 24th International Conference on Software EngineeringThe Symposia of the Society for Experimental Biology 12*, 1958, p. 138–163.

DARZACQ, X.; JÁDY, B. E.; VERHEGGEN, C.; KISS, A. M.; BERTRAND, E.; KISS, T. Cajal body-specific small nuclear rnas: a novel class of 2'-o-methylation and pseudouridylation guide rnas. *The EMBO Journal*, v. 21, p. 2746–2756, 2002.

DURBIN, R.; EDDY, S.; KROGH, A.; MITCHISON, G. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge:Cambridge University Press, 1998.

EDDY, S. R. Hidden markov models. *Curr. Opin. Struct. Biol.*, v. 6, p. 361–365, 1996.

EDDY, S. R. Non-coding rna genes and the modern rna world. *Nature Reviews Genetics*, v. 2, p. 919–929, 2001.

EDDY, S. R. Computacional genomics of noncoding rna genes. *Cell*, v. 109, p. 137–140, 2002.

EDDY, S. R. The infernal users guide. *Disponível em* `http://infernal.janelia.org/`., 2003.

EDDY, S. R. The infernal users guide. *Disponível em* `http://infernal.janelia.org/`., 2009.

EDDY, S. R.; DURBIN, R. Rna sequence analysis using covariance model. *Nucleic Acids Research*, v. 22, n. 11, p. 2079–2088, 1994.

ELICEIRI, G. L. Small nucleolar rnas. *Cell. Mol. Life Sci.*, v. 56, p. 22–31, 1999.

FICHANT, G. A.; BURKS, C. Identifying potential trna genes in genomic dna sequences. *J. Mol. Biol.*, v. 220, n. 03, p. 659–671, 1991.

FREYHULT, E. K.; BOLLBACK, J. P.; GARDNER, P. P. The infernal users guide: Exploring genomic dark matter: A critical assessment of the performance of homology search methods on noncoding rna. *Genome Res.*, v. 17, p. 117–125, 2007.

GRIFFITHS-JONES, S.; BATEMAN, A.; MARSHALL, M.; KHANNA, A.; R.EDDY, S. Rfam: an rna family database. *Nucleic Acids Research*, v. 31, n. 1, p. 439–441, 2003.

GRIFFITHS-JONES, S.; GROCOCK, R. J.; DONGEN, S.; BATEMAN, A.; ENRIGHT, A. J. mirbase: microrna sequences, targets and gene nomenclature. *Nucleic Acids Research*, v. 34, p. D140–D144, 2006.

GRIFFITHS-JONES, S.; MOXON, S.; MARSHALL, M.; KHANNA, A.; EDDY, S. R.; BATEMAN, A. Searching genomes for noncoding rna using fastr. *Nucleic Acids Research*, v. 33, p. D121–D124, 2005.

GUTELL, R. R.; LARSEN, N.; WOESE, C. R. Lessons from an evolving rrna: 16s and 23s rrna structures from a comparative perspective. *Microbiological Reviews*, v. 58, n. 1, p. 10–26, 1994.

HE, L.; THOMSON, J. M.; HEMANN, M. T.; MU, E. H.-M. D.; GOODSON, S.; POWERS, S.; CORDON-CARDO, C.; LOWE, S. W.; HANNON, G. J.; HAMMOND, S. M. A microrna polycistron as a potential human oncogene. *Nature*, v. 435, n. 7043, p. 828–833, 2005.

HENIKOFF, S.; HENIKOFF, J. G. Performance evaluation of amino acid substitution matrices. *Proteins*, v. 17, p. 49–61, 1993.

HOFACKER, I. L.; FEKETE, M.; STADLER, P. F. Secondary structure prediction for aligned rna sequences. *J Mol Biol*, v. 319, p. 1059–1066, 2002.

HOFACKER, I. L.; FONTANA, W.; STADLER, P. F.; BONHOEFFER, L. S.; TACKER, M.; SCHUSTER, P. Fast folding and comparison of rna secondary structures. *Monatsh Chem*, v. 125, p. 167–188, 1994.

HUTTENHOFER, A.; SCHATTNER, P.; POLACEK, N. Non-coding rnas: hope or hype? *Science Direct*, v. 21, p. 289–297, 2005.

KENT, W. J. Blat - the blast-like alignment tool. *Genome Res.*, v. 12, p. 656–664, 2002.

KIN, T.; YAMADA, K.; TERAI, G.; OKIDA, H.; YOSHINARI, Y.; ONO, Y.; KOJIMA, A.; KIMURA, Y.; KOMORI, T.; ASAI, K. frnadb: a platform for mining/annotating functional rna candidates from non-coding rna sequences. *Nucleic Acids Research*, v. 00, p. D1–D4, 2006.

KISS, A. M.; JÁDY, B. E.; DARZACQ, X.; VERHEGGEN, C.; BERTRAND, E.; KISS, T. A cajal body-specific pseudouridylation guide rna is composed of two box h/aca snorna-like domains. *Nucleic Acids Research*, v. 30, n. 21, p. 4643–4649, 2002.

KLEIN, R. J.; EDDY, S. R. RSEARCH: Finding homologs of single structured RNA sequences. *BMC Bioinformatics*, v. 4, p. 44, 2003.

KOOLMAN, J.; ROEHM, K. H. *Color atlas of biochemistry*. Georg Thieme Verlag, 477 p., 2005.

LAFONTAINE, D. L. J.; TOLLERVEY, D. Ribosomal rna. *Encyclopedia of Life Sciences*, 2001.

LIANG-HU, Q.; QING, M.; HUI, Z.; YUE-QIN, C. Identification of 10 novel snorna gene clusters from arabidopsis thaliana. *Nucleic Acids Res.*, v. 29, p. 1623–1630, 2001.

LIU, C.; BAI, B.; SKOGERBO, G.; CAI, L.; DENG, W.; ZHANG, Y.; BU, D.; ZHAO, Y.; CHEN, R. Noncode: an integrated knowledge database of non-coding rnas. *Nucleic Acids Researc*, v. 33, p. 112–115, 2005.

LIU, J.; GOUGH, J.; ROST, B. Distinguishing protein-coding from non-coding rnas through support vector machines. *PLOS Genetics*, v. 2, n. 4, p. 529–536, 2006.

LODISH, H.; BERK, A.; MATSUDAIRA, P. *Molecular cell biology*. W. H. Freeman & Co, 968 p., 2005.

LOPES, S. *Introdução à biologia e origem da vida, citologia, reprodução e embriologia, histologia*. Saraiva, 379 p., 1998.

LOWE, T. M.; EDDY, S. R. trnascan-se: a program for improved detection of transfer rna genes in genomic sequence. *Nucleic Acids Research*, v. 25, p. 955–964, 1997.

LOWE, T. M.; EDDY, S. R. A computational screen for methylation guide snornas in yeast. *Science*, v. 283, p. 1168–1171, 1998.

LOWE, T. M.; EDDY, S. R. A computational screen for methylation guide snornas in yeast. *Science*, v. 283, p. 1168–1171, 1999.

LOWE, T. M. J. *Combining new computational and traditional experimental methods to identify trna and snorna gene families*. Tese de doutoramento, Division of Biology and Biomedical Sciences/Molecular Genetics Program/WASHINGTON UNIVERSITY, Saint Louis, Missouri, 1999.

LU, J.; GETZ, G.; MISKA, E. A.; ALVAREZ-SAAVEDRA, E.; LAMB, J.; PECK, D.; SWEET-CORDERO, A.; EBERT, B. L.; MAK, R. H.; FERRANDO, A. A.; DOWNING, J. R.; JACKS, T.; HORVITZ, H. R.; GOLUB, T. R. Microrna expression profiles classify human cancers. *Nature*, v. 435, n. 7043, p. 834–838, 2005.

LUNDBLAD, R. L. *Biochemistry and molecular biology compendium*. 1ed ed. Taylor & Francis Group, 424 p., 2007.

MATTICK, J. S. Non-coding rnas: the architects of eukaryotic complexity. *EMBO Reports*, v. 2, p. 986–991, 2001.

MICHALAK, P. Rna world - the dark matter of evolutionary genomics. *Journal Compilation*, p. 1768–1774, 2006.

MOUNT, S. M.; GOTEA, V.; LIN, C.; HERNANDEZ, K.; MAKALOWSKI, W. Spliceosomal small nuclear rna genes in 11 insect genomes. *RNA Journal*, v. 13, p. 5–14, 2007.

NAM, J.; KIM, J.; KIM, S.; ZHANG, B. Promir ii: a web server for the probabilistic prediction of clustered, nonclustered, conserved and nonconserved micrornas. *Nucleic Acids Research*, v. 34 (Web Server issue), p. W455–W458, 2006.

NAM, J.; SHIN, K.; HAN, J.; LEE, Y.; KIM, V. N.; ZHANG, B. Human microrna prediction through a probabilistic co-learning model of sequence and structure. *Nucleic Acids Researc*, v. 33, n. 11, p. 3570–3581, 2005.

NELSON, D. L.; COX, M. M. *Lehninger principles of biochemistry*. 4ed ed. W. H. Freeman & Co, 1124 p., 2004.

OGG, S. C.; LAMOND, A. I. Cajal bodies and coilin-moving towards function. *The Journal of Cell Biology*, v. 159, n. 1, p. 17–21, 2002.

PACE, N.; SMITHM, D.; OLSEN, G.; B.JAMES Phylogenetic comparative analysis and the secondary structure of riboclease: a review. *Gene*, v. 82, p. 65–75, 1989.

PAVESI, A.; CONTERIO, F.; BOLCHI, A.; DIECI, G.; OTTONELLO, S. Identification of new eukaryotic trna genes in genomic dna databases by a multistep weight matrix analysis of transcriptional control regions. *Nucleic Acids Research*, v. 22, n. 7, p. 1247–1256, 1994.

RIVAS, E.; EDDY, S. R. Secondary structure alone is generally not statistically significant for the detection of noncoding rnas. *Bioinformatics*, v. 16, n. 7, p. 583–605, 2000.

RIVAS, E.; EDDY, S. R. Noncoding rna gene detection using comparative sequence analysis. *BMC Bioinformatics*, v. 2, 2001.

RIVAS, E.; KLEIN, R. J.; JONES, T. A.; EDDY, S. R. Computational identification of noncoding rnas in e. coli by comparative genomics. *Current Biology*, v. 11, p. 1369–1373, 2001.

SAKAKIBARA, Y.; BROWN, M.; HUGHEY, R.; MIAN, I. S.; SJOLANDER, K.; UNDERWOOD, R. C.; HAUSSLER, D. Stochastic context-free grammars for trna modeling. *Nucleic Acids Research*, v. 22, p. 5112–5120, 1994.

SCHATTNER, P. *Searching for rna genes using base-composition statistics*, v. 30. Nucleic Acids Research, 2076–2082 p., 2002.

SCHATTNER, P.; DECATUR, W. A.; DAVIS, C. A.; JR, M. A.; FOURNIER, M. J.; LOWE, T. M. Genome-wide searching for pseudouridylation guide snornas: analysis of the saccharomyces cerevisiae genome. *Nucleic Acids Research*, v. 32, n. 14, p. 4281–4296, 2004.

SILVA, N. P.; ANDRADE, L. E. C. Noções básicas de biologia molecular. *Revista Brasileira de Reumatologia*, v. 41, p. 83–94, 2001.

SMITH, C. M.; STEITZ, J. A. Sno storm in the nucleolus: New roles for myriad small rnps. *Cell*, v. 89, p. 669–672, 1997.

SPRINZL, M.; HORN, C.; BROWN, M.; IOUDOVITCH, A.; STEINBERG, S. Compilation of trna sequences and sequences of trna genes. *Nucleic Acids Research*, v. 26, n. 1, p. 148–153, 1997.

STRYER, L.; BERG, L.; TYMOCZCO, J. L. *Biochemistry*. 5ed ed. W. H. Freeman & Co, 1514 p., 2002.

VETTORE, A. L.; SILVA, F. R.; KEMPER, E. L.; ET AL. Analysis and functional annotation of an expressed sequence tag collection for tropical crop sugarcane. *Genome Res.*, v. 13, p. 2725–2735, 2003.

VOET, D.; VOET, J. G. *Biochemistry*. 2nd ed. Wiley, 1995.

WASHIETL, S. *Rnaz: Predicting structural noncoding rnas*. Department for Theoretical Chemistry/University Vienna, 2006.

WASHIETL, S.; HOFACKER, I. L.; STADLER, P. F. Fast and reliable prediction of noncoding rnas. *Proc. Natl Acad. Sci. USA*, v. 102, p. 2454–2459, 2005.

ZHANG, S.; HAAS, B.; ESKIN, E.; BAFNA, V. Rfam: annotating non-coding rnas in complete genomes. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, v. 2, n. 4, p. 366–379, 2005.

ZUKER, M.; STIEGLER, P. Optimal computer folding of large rna sequences using thermodynamics and auxiliary information. *Nucleic Acids Res*, v. 9, p. 133–148, 1981.

APÊNDICE

A

Script do Modelo Proposto

A.1 Disponibilização

O modelo proposto para detecção de ncRNAs está hospedado no seguinte *website*: `http://rnanc.googlecode.com`. Ao entrar na url indicada, basta clicar na aba *downloads*, onde podem ser encontradas as diferentes versões do modelo.

Atualmente, o modelo se encontra na versão 2.0 e está disponível para o sistema operacional Linux.

A.2 Dependências

Para utilização do *script* criado, primeiramente deve-se providenciar a instalação da ferramenta Infernal, que está disponível para *download* juntamente com sua documentação em `http://infernal.janelia.org/`. A versão do Infernal utilizada na criação do modelo foi a 0.81.

Em seguida, deve-se obter o pacote Viena. Vale ressaltar que a versão utilizada para a criação do programa de pré-processamento foi a 1.6. O código fonte do pacote e sua documentação estão disponíveis em `http://www.tbi.univie.ac.at/~ivo/RNA/`.

A.3 Script

Para facilitar a utilização do modelo, foi criado um *script* que fornece uma interface gráfica. Para tal, foi utilizado o *Zenity*, mecanismo que permite a criação de uma interação gráfica entre o usuário e os comandos do *script*.

Primeiramente deve-se efetuar o *download*, salvando o arquivo que contém a ferramenta em uma pasta conhecida. É necessário descompactar o arquivo salvo. Por fim, basta acessar o diretório descompactado "*script*" e rodar o script `ncRNA`. Nesse momento, o modelo já está pronto para ser utilizado, como mostra a Figura Figura A.1.

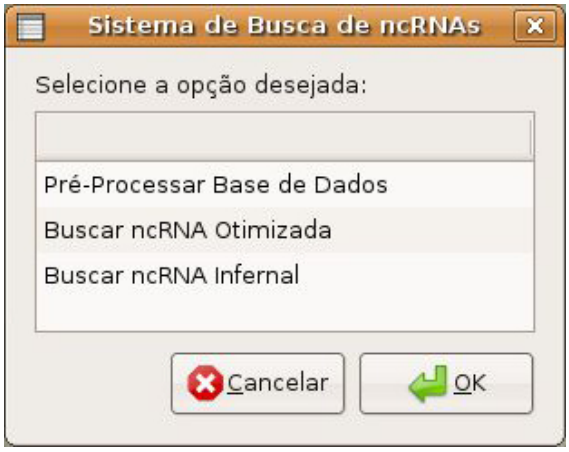

Figura A.1: Interface do Modelo Proposto.

O *script* fornece de três opções:

- **Pré-Processar Base de Dados**. Efetua o pré-processamento de uma base de dados, ou seja, deixá-la no formato que possibilita a busca otimizada com filtro criado.
- **Buscar ncRNA Otimizada**. Efetua a busca por determinado ncRNA de entrada em uma base de dados previamente processada. Essa busca realiza a detecção de ncRNAs utilizando o modelo proposto neste trabalho de mestrado.
- **Buscar ncRNA Infernal**. Efetua busca de determinado ncRNA utilizando apenas o Sistema de Busca Infernal com parâmetros *default*.

Ao selecionar a primeira opção "Pré-Processar Base de Dados", deve-se escolher o arquivo que contém a base de dados a ser processada (Figura A.2).

Enquanto ocorre o pré-processamento é mostrada uma barra de progresso. Ao finalizar, é gerado o arquivo contendo a base de dados processada, na mesma pasta que contém a base de dados original e é exibida novamente a tela inicial.

Figura A.2: Seleção de Base de Dados a ser pré-processada.

Para efetuar a busca otimizada de um ncRNA, deve-se selecionar a opção "Buscar ncRNA Otimizada". Em seguida deve-se selecionar o arquivo de entrada que possua o ncRNA a ser buscado (no formato do modelo de covariância) e, posteriormente, selecionar o arquivo que contenha a base de dados processada. Novamente, enquanto ocorre a busca é exibida a barra de progresso.

Para utilizar diferentes filtros deve-se alterar a chamada para o filtro requerido na função que efetua o filtro da base de dados para otimização da busca. No fim da execução da busca, a saída do Infernal é exibida, como resultado (Figura A.3).

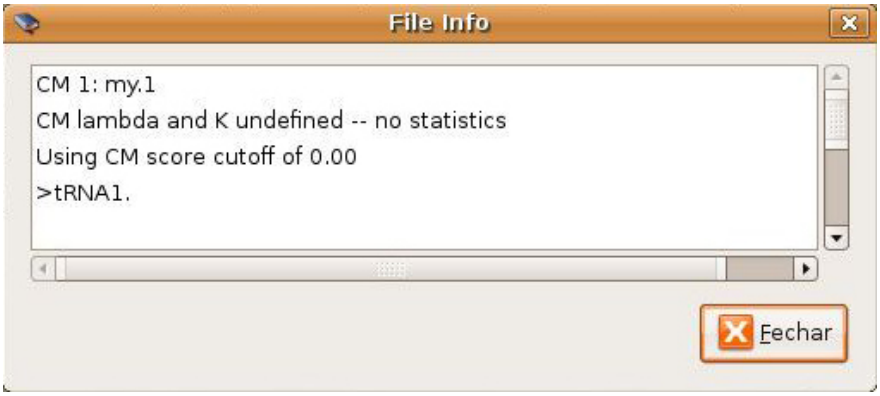

Figura A.3: Saída da Busca Otimizada.

O *script* pergunta ao usuário se ele deseja efetuar uma nova busca utilizando a mesma entrada, porém em outra base, evitando, assim, uma nova extração de características do mesmo arquivo de entrada. Caso cancele, irá voltar à tela inicial.

A busca utilizando unicamente o Infernal, funciona de forma semelhante à descrita para a busca otimizada. Durante a execução do *script* é registrado em log no *shell* o que for ocorrendo, inclusive o tempo gasto para efetuar a busca pelo ncRNA.