UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

LEONARDO VIANNA DO NASCIMENTO

Um Sistema Baseado em Agentes para Re-anotação de Genomas

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Profª Drª Ana Lúcia Bazzan
Orientadora

Porto Alegre, julho de 2005

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Nascimento, Leonardo Vianna do

Um Sistema Baseado em Agentes para Re-anotação de Genomas / Leonardo Vianna do Nascimento. – Porto Alegre: PPGC da UFRGS, 2005.

61 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2005. Orientadora: Ana Lúcia Bazzan.

1. Bioinformática. 2. Re-anotação de genomas. 3. Sistemas Baseados em Agentes. I. Bazzan, Ana Lúcia. II. Título.

AGRADECIMENTOS

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

BLAST	Basic Local Alignment Search Tool
COG	Cluster of Orthologous Groups
DDBJ	DNA Data Bank of Japan
DNA	Deoxyribonucleic Acid
EMBL	European Molecular Biology Laboratory
EST	Expressed Sequence Tag
FTP	File Transfer Protocol
GRAIL	Gene Recognition and Analysis Internet Link
HMM	Hidden Markov Model
HTTP	HyperText Transfer Protocol
IMM	Interpolated Markov Model
INSDC	International Nucleotide Sequence Database Collaboration
LNCC	Laboratório Nacional de Computação Científica
NCBI	National Center for Biotechnology Information
ORF	Open Reading Frame
PDB	Protein Data Bank
PIR	Protein Information Resource
RNA	RiboNucleic Acid
SABIA	System for Automated Bacterial Integrated Annotation
TrEMBL	Translated EMBL

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

A análise da informação contida em seqüências genéticas tem ganho cada vez mais importância nos dias atuais. A chamada *anotação genética* tem o objetivo de, a partir de uma ou mais seqüências, determinar suas características estruturais e funcionais. Muitos processos de anotação já foram realizados com êxito aumentando consideravelmente nosso conhecimento acerca do mecanismo genético de diversos organismos.

A *re-anotação genética* é um processo que visa revisar o resultado da anotação, em virtude da disponibilidade de novas informações. Neste trabalho foi desenvolvido um sistema de re-anotação automática, onde tarefas de análise repetitivas podem ser automatizadas e os dados na anotação re-analisados periodicamente, a fim de que possíveis modificações possam ser detectadas. O sistema é baseado na tecnologia de agentes. Cada agente é responsável pela execução de diferentes ferramentas de bioinformática. Ao final do processo, os resultados individuais são combinados a fim de atingir o objetivo da análise. O sistema demonstrou eficácia na análise realizada em organismos procariontes durante a fase de validação. Ambientes de re-anotação como este são ferramentas interessantes a serem futuramente integradas a sistemas de anotação existentes.

Palavras-chave: Bioinformática, Re-anotação de genomas, Sistemas Baseados em Agentes.

An Agent-Based System for Re-annotation of Genomes

ABSTRACT

The analysis of the information present in genetic sequences is gaining more importance nowadays. The *genetic annotation* aims to determine structural and functional characteristics of the sequences. Many annotation processes have already been carried out, improving our knowledge about the genetic mechanism of several organisms.

The *genetic re-annotation* is a process that aims to review the annotation results when new information is available. This work presents an automatic re-annotation system where repetitive analysis tasks can be automated and annotation data are periodically re-analysed in order to detect possible differences. The system is based on the agent technology and each agent must execute different bioinformatics tools and merge its results in order to reach the analysis goals. The system proved to be efficient on the analysis carried out in procariotic organisms in the validation process, becoming an interesting tool to be integrated in annotation systems in the future.

Keywords: bioinformatics,genome re-annotation, multi-agent systems.

1 INTRODUÇÃO

Muitas das descobertas mais significativas do último século se desenvolveram no campo da Biologia Molecular. As pesquisas realizadas sobre a estrutura química e molecular das estruturas genéticas responsáveis pelo gerenciamento das atividades celulares, e por conseguinte da própria vida, permitiram o surgimento de projetos audaciosos de investigação do chamado *genoma*. O mais conhecido deles, sem dúvida, foi o *Projeto Genoma Humano* (LANDER et al., 2001), mas muitos outros projetos relacionados a outros organismos estão em andamento ou já se encontram concluídos, muitos deles com dados completos disponíveis na Internet [1].

Desde que Ray Wu seqüênciou o primeiro segmento de DNA, em 1970, o número de informações biológicas disponíveis tem crescido bastante, a ponto de atingir taxas exponenciais nas últimas décadas. Os bancos de dados biológicos publicamente disponíveis, apresentam hoje dimensões que tornam a análise e comparação manuais destes dados impraticável.

Os computadores têm desempenhado um papel de destaque na montagem e subseqüente análise de seqüências genéticas. A capacidade de processar grandes quantidades de dados em um curto espaço de tempo e a disponibilidade de acesso público a bases eletrônicas de dados biológicos representaram um grande avanço quanto aos antigos métodos de análise manual. A grande importância das ferramentas computacionais na análise de seqüências genéticas possibilitou a criação de uma nova área interdisciplinar denominada *Bioinformática*. Durante as últimas décadas, inúmeras ferramentas computacionais contribuíram de forma significativa na melhoria de tarefas comuns na análise genética (como comparação de seqüências, busca, anotação de genes, etc), bem como na integração destas em sistemas mais complexos.

Ferramentas computacionais têm se destacado principalmente em tarefas relacionadas à *Anotação Genética* (STEIN, 2001). A anotação genética têm o objetivo de obter informações estruturais e/ou funcionais sobre uma ou várias seqüências relacionadas a um determinado genoma. Tarefas comuns à anotação genética incluem a busca por seqüências semelhantes (homólogos ou ortólogos), busca por sítios funcionais dentro da seqüência (motivos ou domínios), inferência da estrutura espacial de proteínas, etc. Todas estas tarefas encontram ferramentas capazes de auxiliar o especialista humano de forma significativa, reduzindo tempo e esforço necessários para a conclusão de um processo de anotação.

Em muitos casos, os resultados obtidos por um processo de anotação podem sofrer alterações. Novas informações provenientes de comparações com novas seqüências e sítios funcionais podem fornecer indícios que levem ao acréscimo de novas características

[1] http://www.ncbi.nlm.nih.gov

ou até mesmo à eliminação de regiões funcionais cuja existência não foi totalmente comprovada durante a anotação (proteínas hipotéticas). Logo, é desejável que os resultados de anotações genéticas possam ser re-analisados de forma sistemática em um processo denominado *re-anotação*.

Até o momento nenhum sistema integrado de re-anotação está disponível para uso pela comunidade científica. A metodologia adotada atualmente nestes processos é a execução manual de diversas ferramentas computacionais sobre as seqüências desejadas e a subseqüente integração manual dos resultados. Este processo de integração manual é uma tarefa muitas vezes custosa, e que envolve ações repetitivas.

A carência de sistemas computacionais integrados de re-anotação foi a motivação para o desenvolvimento, neste trabalho, de uma ferramenta de re-anotação automática, baseada na tecnologia de agentes, capaz de efetuar análises periódicas sobre o resultado obtido durante o processo de anotação. Novas informações encontradas são reportadas à especialistas humanos, que podem analisar a informação e decidir se as modificações sugeridas são adequadas ou não. Os diversos agentes de software responsáveis pela obtenção de informações são capazes de agir cooperativamente a fim de resolver conflitos e maximizar a performance do sistema.

Os próximos capítulos abordam diversos temas relacionados ao desenvolvimento deste trabalho, bem como o sistema de re-anotação desenvolvido. O capítulo 2 apresenta uma breve introdução aos processos e conceitos biológicos envolvidos neste trabalho, bem como à área de bioinformática e as principais ferramentas computacionais relacionadas ao trabalho. Já o capítulo 3 descreve os processos de anotação e re-anotação em mais detalhe. A organização, implementação e validação do sistema são discutidos nos capítulos 4 e 5. O capítulo 6 apresenta uma breve conclusão bem como perspectivas de trabalhos futuros relacionados ao sistema desenvolvido.

2 BIOLOGIA MOLECULAR E BIOINFORMÁTICA

O interesse na análise de dados genéticos é cada vez maior. Com os constantes avanços da ciência no campo da Biologia Molecular, uma vasta quantidade de dados sobre a estrutura e funcionalidade do genoma de diversos organismos se tornou disponível. Muitas informações podem ser facilmente acessadas através de bancos de dados públicos disponíveis na Internet, como o *Genbank* (BENSON et al., 2004). O acesso a estas informações é um fator de vital importância para os processos de análise que utilizam a similaridade com estruturas já catalogadas para inferir certas informações acerca de novos genomas.

2.1 Conceitos Básicos de Biologia Molecular

Alguns conceitos biológicos são de extrema importância para a compreensão das técnicas de análise apresentadas neste trabalho. Dentre estes conceitos, são apresentados aqui os de DNA e RNA bem como o processo bioquímico conhecido como o *Dogma Central da Biologia Molecular*.

2.1.1 DNA e RNA

Todo o material genético de um ser vivo está localizado em suas células, codificado em longas cadeias de uma molécula chamada *DNA* (*DeoxyriboNucleic Acid* - Ácido DeoxiriboNucleico). O conhecimento sobre sua composição, função e estrutura foi resultado de diversas pesquisas sobre a natureza química do material genético, desenvolvidas a partir de meados do século passado.

O DNA tem a forma de uma hélice dupla, formada por cadeias de ácidos nucleicos chamados *nucleotídeos* (Figura 2.1). Existem quatro tipos de nucleotídeos encontrados no DNA: *Adenina*(A), *Citosina*(C), *Guanina*(G) e *Timina*(T). Cada nucleotídeo em uma das cadeias da hélice dupla é ligado a outro nucleotídeo na cadeia oposta, da seguinte forma: Adenina é ligada a Timina(A-T) e Guanina é ligada a Citosina(C-G), e vice versa. Assim, uma molécula de DNA completa possui duas cadeias de nucleotídeos que são complementares. Tais nucleotídeos são muitas vezes chamados *bases* e uma cadeia de DNA é freqüentemente medida em milhares de bases, abreviadamente *kb*.

Outra molécula que desempenha um papel importante nos processos estudados na biologia molecular é o *RNA* (*RiboNucleic Acid* - Ácido RiboNucleico). Sua estrutura é semelhante ao DNA, mas difere na forma e composição. A molécula de RNA possui a forma de uma hélice simples, composta também por quatro nucleotídeos: Adenina, Citosina, Guanina, como no DNA, e a *Uracil* (U) substitui a Timina.

Existe uma variedade de tipos diferentes de moléculas de RNA presentes em uma cé-

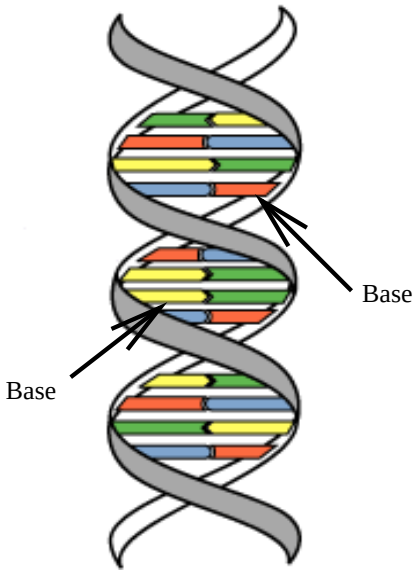

Figura 2.1: Representação simplificada de uma molécula de DNA.

lula. O RNA *mensageiro* (mRNA) é transcrito a partir do DNA, tranportando a informação genética até uma organela presente no citoplasma celular chamada *ribossomo*, responsável pela síntese de proteínas. O RNA *transportador* (tRNA) é responsável pelo transporte de moléculas de amino-ácidos até o ribossomo. Já o RNA *ribossômico* é responsável por catalizar algumas etapas do processo de síntese de proteínas.

2.1.2 Proteínas

Grande parte do processo químico celular é controlado por longas moléculas de proteínas. O estudo sobre a composição e função das proteínas é um passo essencial para a compreensão do mecanismo funcional de uma célula.

Uma proteína é formada por uma longa cadeia de *amino-ácidos*. Existem cerca de 20 amino-ácidos que são utilizados por todas as espécies vivas conhecidas. A Tabela 2.1 apresenta a listagem completa destes amino-ácidos.

A seqüência química da proteína (contendo o conjunto de amino-ácidos pertencentes a ela) é chamada *estrutura primária*. A estrutura primária por si só não é suficiente para a determinação exata da função da proteína. O modo como a proteína está disposta espacialmente é um fator decisivo na função desempenhada pela molécula. As estruturas secundária e terciária da proteína especificam como os componentes da estrutura primária se organizam em elementos estruturalmente mais complexos, como *hélices alfa* (Figura 2.2).

As proteínas podem desempenhar funções na célula, tanto na ativação de algum processo importante, como na composição estrutural das células e na catalização de reações químicas (enzimas). É comum que diversas proteínas produzidas pela célula executem uma reação química de forma conjunta e em uma seqüência definida, formando as chamadas *vias metabólicas*.

Duas proteínas diferentes são consideradas *ortólogas* quando desempenham a mesma função mas em organismos diferentes. Já proteínas que desempenham funções diferentes, mas relacionadas, dentro do mesmo organismo são conhecidas como *parálogas*. A análise da ocorrência de proteínas parálogas e ortólogas nos organismos estudados é utilizada na obtenção de relacionamentos entre proteínas em diferentes espécies, bem como na

Figura 2.2: Uma representação gráfica simplificada da estrutura de uma hélice alfa.

Figura 2.3: Organização dos genes em organismos procariontes.

compreensão do mecanismo evolucionário existente.

Outra definição importante a ser destacada sobre proteínas é a de *homologia*. Duas proteínas são ditas homólogas se estão relacionadas por divergência a partir de um ancestral comum na cadeia evolucionária. A divergência entre ambas ocorre a partir de mutações sucessivas ocorridas em ambas as seqüências, alterando sua composição e possivelmente sua função no organismo.

2.1.3 Organização do Genoma

A seqüência completa de DNA que codifica um ser vivo é chamada *genoma*. O genoma, entretanto, não funciona como uma única e longa seqüência. Seu conteúdo é dividido em genes individuais. Um *gene* é uma pequena seção da seqüência genética, com um propósito único e específico.

Existem três classes de genes. Os genes codificadores de proteínas são utilizados como modelos para construção de proteínas. Outros genes presentes no genoma são responsáveis pela produção de moléculas de RNA. Finalmente, existem genes que são responsáveis pela regulação da atividade de outros genes. Esse genes são os chamados *genes reguladores*.

Os genes apresentam estruturas diferentes em organismos *Procariontes* [1] e em organismos *Eucariontes* [2]. Organismos procariontes apresentam genes estruturados conforme ilustrado na Figura 2.3. Regiões promotoras (*promoters*) são seqüências responsáveis por promover a transcrição dos genes. *Start codons* são triplas de bases que marcam o início de uma região codificadora. Já os *Stop codons* são triplas de bases que especificam o final da seqüência codificadora de um gene.

Organismos eucariontes apresentam uma estrutura genética mais complexa (Figura 2.4). A região codificadora de cada gene, nestes organismos, encontra-se distribuída em diversas regiões menores denominadas *exons*. Exons estão separados dentro do gene por

[1] Organismos que não possuem núcleo definido (Ex.: bactérias).
[2] Organismos que possuem núcleo celular delimitado (Ex.: seres humanos).

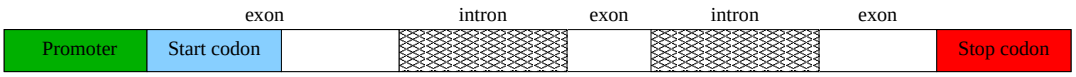

Figura 2.4: Organização dos genes em organismos eucariontes.

Tabela 2.1: Codificação de amino-ácidos

	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	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

regiões chamadas *íntrons*. Logo, para que a região codificadora de um gene eucariota possa ser construída, é necessário remover os íntrons e unir os exons de forma a obter uma única seqüência. Este processo, denominado *intron splicing* será contextualizado na próxima seção.

2.1.4 O Dogma Central da Biologia Molecular

O material genético pode ser visto como um livro de receitas de proteínas. Genes específicos dentro do genoma produzem proteínas que por sua vez desempenham funções específicas dentro da célula. O processo de produzir proteínas a partir de seqüências de genes presentes em uma molécula de DNA é conhecido como o *Dogma Central da Biologia Molecular*.

O processo inicia com a *transcrição* do gene desejado em uma molécula de RNA mensageiro. Esta molécula de RNA conterá a seqüência complementar ao gene (contendo inclusive os íntrons, no caso de organismos eucariontes). O próximo passo é transportar o mRNA até uma organela denominada *ribossomo*, responsável por *traduzir* a molécula de mRNA em uma proteína funcional. Em organismos eucariontes, um passo extra de *intron splicing* é realizado antes da tradução, no momento em que a molécula de mRNA é retirada do núcleo celular. Nesta etapa adicional os íntrons são retirados, de forma que apenas as bases pertencentes a *exons* sejam mantidas. Íntrons não são utilizados na contrução de proteínas, e logo não precisam estar presentes durante a etapa de tradução.

Cada amino-ácido é codificado na cadeia de mRNA através de uma seqüência de três nucleotídeos chamada *códon*. A Tabela 2.1 apresenta o conjunto completo de códigos utilizados pela grande maioria dos seres vivos. Esta tabela está dividida em quatro colunas

Tabela 2.2: Descrição dos principais amino-ácidos existentes

Acrônimo	Nome do Amino-ácido	Símbolo
Ala	Alanina	A
Arg	Arginina	R
Asn	Asparagina	N
Asp	Aspartato (Ácido aspártico)	D
Cys	Cisteína	C
Gln	Glutamina (Glutamida)	Q
Glu	Glutamato (Ácido glutâmico)	E
Gly	Glicina	G
His	Histidina	H
Ile	Isoleucina	I
Leu	Leucina	L
Lys	Lisina	K
Met	Metionina	M
Phe	Fenilalanina	F
Pro	Prolina	P
Ser	Serina	S
Thr	Treonina	T
Trp	Triptofano (Triptofana)	W
Tyr	Tirosina	Y
Val	Valina	V

e quatro linhas, onde cada linha contém os códons iniciados pelo nucleotídeo correspondente. Cada coluna especifica o segundo nucleotídeo da seqüência do códon. Existe uma coluna extra ao final da tabela que especifica a ordem do terceiro nucleotídeo pertencente à seqüência dos códons. A codificação mostrada na tabela contém os códigos RNA dos códons, utilizando as bases descritas na Seção 2.1.1. Existem 64 possibilidades possíveis de combinações utilizando três nucleotídeos, mas somente 20 amino-ácidos são codificados. Isto ocorre porque alguns códons são redundantes, enquanto outros desempenham funções especiais (*stop codons*). A descrição dos aminoácidos é apresentada na Tabela 2.2.

Durante o processo de tradução, moléculas de amino-ácidos presentes na célula serão transportadas até o local adequado e unidas de forma a construir uma cadeia completa. Dentro do citoplasma celular, existem diversas moléculas de tRNA contendo exatamente um códon. Estas moléculas são ligadas ao amino-ácido equivalente ao códon representado, transportando o amino-ácido até o ribossomo, onde o códon contido na molécula de tRNA é associado a um códon complementar equivalente na molécula de mRNA (Figura 2.5). Deste modo, a proteína desejada é construída, amino-ácido a amino-ácido, até a molécula correspondente estar completa e pronta a ser utilizada pela célula.

2.2 Armazenamento e Obtenção de Informações Genéticas

Atualmente, o mapeamento da estrutura e funcionalidade de genomas tem ocupado cada vez mais espaço no meio científico. A obtenção de informações precisas, a nível genético, sobre muitos organismos pode levar a tratamentos mais eficazes para pragas

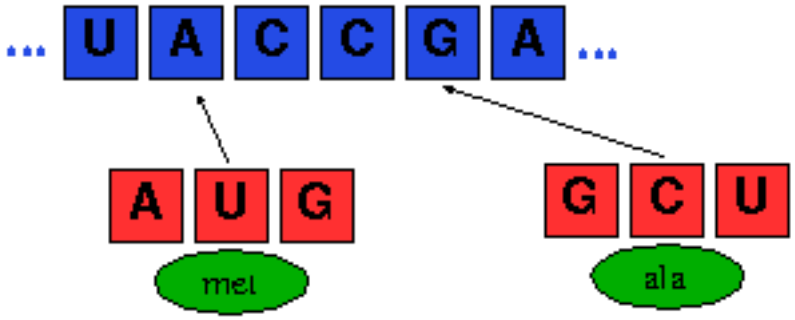

Figura 2.5: O processo de tradução.

e doenças, bem como a um melhor esclarecimento do funcionamento de nosso próprio organismo. Nesta seção serão apresentadas as principais questões relacionadas à análise da estrutura e funcionalidade de genes e genomas.

2.2.1 Representação de Seqüências Genéticas

Muitas informações extraídas a partir de pesquisas sobre a estrutura e função de muitas moléculas presentes em diversos genomas de diferentes organismos encontra-se armazenada em meio digital, disponível através de bancos de dados acessíveis a partir da Internet. Para que este armazenamento digital seja possível, é preciso representar a informação genética de uma forma passível de processamento por um computador.

Esta tarefa é realizada durante o processo de *seqüenciamento*, onde a seqüência de DNA é codificada em uma seqüência de caracteres representando sua estrutura química. Cada caractere representa um nucleotídeo diferente. Assim, uma seqüência de DNA pode ser representada como “ATGCCTAAGC”, onde a letra “A” representa um nucleotídeo de adenina, “T” representa um nucleotídeo de timina, e assim por diante. Codificação semelhante é adotada para moléculas de RNA e proteínas, estas últimas utilizando letras para representar amino-ácidos. *Strings* como estes podem ser processados em um computador e armazenados em bancos de dados específicos para dados genéticos.

2.2.2 Identificando Seqüências Genéticas

No momento em que a seqüência de nucleotídeos de um genoma está disponível, outras informações podem ser obtidas, principalmente a partir de ferramentas computacionais. Algumas das principais características relacionadas a genes e outras entidades presentes nos genomas serão descritas a seguir.

2.2.2.1 Homologia e Ortologia

Os conceitos de homologia e ortologia definidos para proteínas (seção 2.1.2) podem ser estendidos para genes. A determinação de genes ortólogos pode auxiliar na análise da evolução de determinados genes e na construção de árvores filogenéticas.

A principal fonte de informação com relação a genes ortólogos é o banco de dados COG (*Cluster of Orthologous Groups*) (seção 2.2.3.2). No COG existem diversos genes identificados em diferentes organismos que são classificados em grupos juntamente com outros genes ortólogos. Além disso, ferramentas de comparação de seqüências, como o BLAST (seção 2.2.4.1) podem ser utilizadas na determinação de genes ortólogos.

2.2.2.2 Motivos

Um ítem bastante considerado durante a anotação genética é a associação de uma seqüência de aminoácidos desconhecida a alguma família ou domínio conhecido de proteínas. Estes grupos são geralmentes representados por padrões especiais encontrados em seqüências, que estão relacionados a funcionalidades específicas. Famílias e domínios de proteínas conhecidos encontram-se armazenados em bancos de dados especialmente construídos para tal.

As famílias de proteínas e os domínios funcionais são representados por *motivos*, geralmente associados à função biológica desempenhada pelos primeiros. Os motivos são por sua vez representados por algum mecanismo de especificação de padrões, como expressões regulares ou HMMs (*Hidden Markov Models*). Um exemplo de representação de motivos utilizado no banco de dados Prosite (FALQUET et al., 2002) é mostrado abaixo:

$$K - [RG] - X(2) - C - DE - X(3,5) - [LIVM]$$

O padrão deve ser lido da seguinte maneira: o amino-ácido *lisina* (K), seguido tanto por *arginina* (R) ou *glicina* (G); dois resíduos quaisquer; uma *cisteína* (C); qualquer resíduo, menos *ácido aspártico* (D) ou *ácido glutâmico* (E); de três a cinco resíduos não especificados, terminando com *leucina* (L), *isoleucina* (I), *valina* (V) ou *metionina* (M). As proteínas que fazem parte desta família apresentam esse padrão em algum lugar na sua seqüência.

2.2.2.3 Vias Metabólicas

Processos como os ocorridos nas chamadas *vias metabólicas* podem conter diversos genes ativados simultaneamente ou em uma seqüência definida, que possibilitam que determinada ação seja realizada pela célula. Alterações nos genes podem levar a falta de proteínas importantes para a reação, impossibilitando a realização de um ou mais processos.

Logo, a determinação de quais vias metabólicas estão relacionadas a um gene é uma informação bastante interessante na especificação completa de sua função. A análise neste nível de processos celulares é bastante complexa, muitas vezes exigindo análises químicas sobre as moléculas estudadas.

2.2.3 Bancos de Dados Biológicos

A grande quantidade de informações provenientes da análise de genomas em todo o mundo está distribuída em diversos bancos de dados públicos, disponíveis na Internet. Estes bancos de dados são mantidos por diversas organizações diferentes, possuindo estruturas e mesmo terminologias diversificadas. O acesso a estes dados pode ser feito de diversas maneiras, incluindo transferência de dados via *e-mail*, cópia de arquivos através de FTP e, mais recentemente, consultas efetuadas via um formulário HTML na Web.

Os bancos de dados envolvendo seqüências de nucleotídeos, amino-ácidos ou estruturas de proteínas podem ser classificados em bancos de seqüências primários ou secundários. Os primeiros armazenam diretamente seqüências de amino-ácidos, nucleotídeos e estruturas de proteínas, sem qualquer tipo de processamento ou análise. Dentre estes podemos citar o GenBank (BENSON et al., 2004), o DDBJ (MIYAZAKI et al., 2004) (*DNA Data Bank of Japan*) e o PDB (BERNSTEIN et al., 1977) (*Protein Data Bank*).

Os bancos de seqüências secundários, como o PIR (BARKER et al., 1999) (*Protein*

Information Resource) e o SWISS-PROT (GASTEIGER; JUNG; BAIROCH, 2001) são aqueles que derivam dos primeiros. O SWISS-PROT, por exemplo, mantém um banco de dados de proteínas com um alto nível de anotação funcional. Associado a esta base, existe o TrEMBL, um banco de dados contendo as seqüências de proteínas não completamente anotadas.

2.2.3.1 GenBank

O GenBank (BENSON et al., 2004) (Genetic Sequence Data Bank) é um repositório internacional de seqüências genéticas conhecidas a partir de uma variedade de organismos. Neste banco de dados é mantida uma coleção de nucleotídeos e proteínas, juntamente às referências bibliográficas que detalham seu papel e à anotação biológica equivalente.

O Genbank é mantido pelo NCBI (*National Center for Biotechnology Information*). O NCBI, criado em 1988 pelo governo dos Estados Unidos, é responsável por criar e manter bancos de dados públicos, conduzir pesquisas na área de biologia computacional e desenvolver ferramentas de software próprias para análise de dados genéticos, além de permitir a disseminação de informações biomédicas.

Atualmente os dados enviados ao GenBank são provenientes, principalmente, de submissões diretas realizadas pelos próprios autores de descobertas de seqüências, seguidas pelas submissões de ESTs (*Expressed Sequence Tags*)[3] e dados de *high-throughput* [4] gerados por centros de seqüenciamento, além de dados patenteados e seqüências provenientes de outros dois bancos colaboradores - EMBL (KULIKOVA et al., 2004) e DDBJ. Os dados são disponibilizados na Internet, sem custo, via FTP ou HTTP.

O GenBank, juntamente com o EMBL e o DDBJ, faz parte da *International Nucleotide Sequence Database Collaboration* (INSDC). Estes três bancos compartilham as seqüências armazenadas, de modo que quando uma seqüência está presente em um deles, também pode ser encontrada nos outros dois. Esta troca de informações é viabilizada graças à utilização de um formato padrão de armazenagem. Ferramentas de busca por similaridade e de consulta a seqüências estão também disponíveis.

Cada entrada no GenBank é uma descrição concisa sobre a seqüência. A informação armazenada inclui o nome científico (campo *Source*) e a taxonomia[5] (campo *Organism*) do organismo ao qual a seqüência teve origem, referências bibliográficas (seção *Reference*, contendo os campos *Authors*, *Title*, *Journal*, *Pubmed*) e uma tabela de características (seção *Features*) que identifica as regiões codificantes, regiões com significado biológico e as traduções das regiões codificantes. Além disso, a própria seqüência de DNA pode ser visualizada no campo *Origin*. Um exemplo de registro pode ser visto na Figura 2.6.

Cada registro no GenBank possui um número de acesso, que é estável e único. Com o propósito de identificar seqüências específicas provenientes de diferentes fontes, o GenBank adicionalmente inclui um identificador único chamado *gi*. O *gi* é um número arbitrário, que se refere especificamente à seqüência exata. Se uma base que compõe a seqüência for alterada, um novo registro é adicionado contendo um novo número *gi*, sem que com isso o registro antigo seja perdido, pois o novo registro aponta para o antigo e o antigo passa a apontar para o novo.

[3] Partes da seqüência do gene que efetivamente codificam uma proteína. Estas seqüências contêm o código do gene, sem nenhuma outra seqüência reguladora.

[4] Seqüências provenientes de processos de decodificação de genomas em andamento.

[5] Categorização do organismo onde a protéina foi identificada.

```
LOCUS       NM_198314               1371 bp    mRNA    linear   PRI 11-MAR-20(
DEFINITION  Homo sapiens 5-hydroxytryptamine serotonin receptor 3E (HTR3E),
            transcript variant 2, mRNA.
ACCESSION   NM_198314
VERSION     NM_198314.1  GI:45356152
KEYWORDS    .
SOURCE      Homo sapiens (human)
  ORGANISM  Homo sapiens
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
            Mammalia; Eutheria; Primates; Catarrhini; Hominidae; Homo.
REFERENCE   1  (bases 1 to 1371)
  AUTHORS   Karnovsky,A.M., Gotow,L.F., McKinley,D.D., Piechan,J.L.,
            Ruble,C.L., Mills,C.J., Schellin,K.A.B., Slightom,J.L.,
            Fitzgerald,L.R., Benjamin,C.W. and Roberds,S.L.
  TITLE     A cluster of novel serotonin receptor 3-like genes on human
            chromosome 3
  JOURNAL   Gene 319, 137-148 (2003)
   PUBMED   14597179
REFERENCE   2  (bases 1 to 1371)
  AUTHORS   Niesler,B., Frank,B., Kapeller,J. and Rappold,G.A.
  TITLE     Cloning, physical mapping and expression analysis of the human
            5-HT3 serotonin receptor-like genes HTR3C, HTR3D and HTR3E
  JOURNAL   Gene 310, 101-111 (2003)
   PUBMED   12801637
COMMENT     PROVISIONAL REFSEQ: This record has not yet been subject to final
            NCBI review. The reference sequence was derived from AY349353.1.
FEATURES             Location/Qualifiers
     source          1..1371
                     /organism="Homo sapiens"
                     /mol_type="mRNA"
                     /db_xref="taxon:9606"
                     /chromosome="3"
                     /map="3q27.3"
     gene            1..1371
                     /gene="HTR3E"
                     /note="synonym: 5-HT3c1"
                     /db_xref="GeneID:285242"
                     /db_xref="LocusID:285242"
     CDS             1..1371
                     /gene="HTR3E"
                     /note="isoform b is encoded by transcript variant 2"
                     /codon_start=1
                     /product="5-hydroxytryptamine serotonin receptor 3E
                     isoform b"
                     /protein_id="NP_938056.1"
                     /db_xref="GI:45356153"
                     /db_xref="GeneID:285242"
                     /db_xref="LocusID:285242"
                     /translation="MLAFILSRATPRPALGPLSYREHRVALLHLTHSMSTTGRGVTI
                     INCSGFGQHGADPTALNSVFNRKPFRPVTNISVPTQVNISFAMSAILDVVWDNPFIS
                     NPEECEGITKMSMAAKNLWLPDIFIIELMDVDKTPKGLTAYVSNEGRIRYKKPMKVI
                     ICNLDIFYFPFDQQNCTLTFSSFLYTVDSMLLDMEKEVWEITDASRNILQTHGEWEI
                     GLSKATAKLSRGGNLYDQIVFYVAIRRRPSLYVINLLVPSGFLVAIDALSFYLPVKS
                     NRVPFKITLLLGYNVFLLMMSDLLPTSGTPLIGVYFALCLSLMVGSLLETIFITHLI
                     VATTQPPPLPRWLHSLLLHCNSPGRCCPTAPQKENKGPGLTPTHLPGVKEPEVSAG(
                     PGPAEAELTGGSEWTRAQREHEAQKQHSVELWLQFSHAMDAMLFRLYLLFMASSII1
                     ICLWNT"
     misc_feature    196..1323
                     /gene="HTR3E"
                     /note="KOG3645; Region: Acetylcholine receptor [Signal
                     transduction mechanisms]"
                     /db_xref="CDD:21426"
ORIGIN
        1 atgttagctt tcattttatc acgggcgacc ccacgccctg ccttggggcc cctctcatat
       61 agggagcaca gggttgctct ccttcatctc acacattcga tgtccactac aggaaggggc
      121 gttactttca ccatcaattg ctcagggttt ggccagcacg gggcggatcc cactgctctg
      181 aattcagtgt ttaatagaaa gcccttccgt ccggtcacca acatcagcgt ccccacccaa
      241 gtcaacatct ccttcgcgat gtctgccatc ctagatgtgg tttgggataa cccatttatc
      301 agctggaacc cagaggaatg tgagggcatc acgaagatga gtatggcagc caagaacctg
      361 tggctcccag acattttcat cattgaactc atggatgtgg ataagacccc aaaaggcctc
      421 acagcatatg taagtaatga aggtcgcatc aggtataaga aacccatgaa ggtggacagt
      481 atctgtaacc tggacatctt ctacttcccc ttcgaccagc agaactgcac actcaccttc
      541 agctcattcc tctacacagt ggacagcatg ttgctggaca tggagaaaga agtgtgggaa
      601 ataacagacg catcccggaa catccttcag acccatggag aatgggagct cctgggcctc
      661 agcaaggcca ccgcaaagtt gtccagggga ggcaacctgt atgatcagat cgtgttctat
      721 gtggccatca ggcgcaggcc cagcctctat gtcataaacc ttctcgtgcc cagtggcttt
      781 ctggttgcca tcgatgccct cagcttctac ctgccagtga aaagtgggaa tcgtgtccca
      841 ttcaagataa cgctcctgct gggctacaac gtcttcctgc tcatgatgag tgacttgctc
      901 cccaccagtg gcaccccct catcggtgtc tacttcgccc tgtgcctgtc cctgatggtg
      961 ggcagcctgc tggagaccat cttcatcacc cacctgctgc acgtggccac cacccagccc
     1021 ccacccctgc ctcggtggct ccactccctg ctgctccact gcaacagccc ggggagatgc
     1081 tgtcccactg cgccccagaa ggaaaataag ggcccgggtc tcacccccac ccacctgccc
     1141 ggtgtgaagg agccagaggt atcagcaggg cagatgccgg gccctgcgga ggcagagctg
     1201 acaggggct cagaatggac aagggcccag cgggaacacg aggcccagaa gcagcactca
     1261 gtggagctgt ggttgcagtt cagccacgcg atggacgcca tgctcttccg cctctacctg
     1321 ctcttcatgg cctcctctat catcaccgtc atatgcctct ggaacaccta g
//
```

Figura 2.6: Um exemplo de registro no GenBank.

A quantidade de dados armazenados no GenBank cresce exponencialmente - historicamente os dados vinham dobrando a cada 18 meses, mas a média diminuiu para 15 meses. Essa diferença se deve, principalmente, ao enorme crescimento de dados provenientes do seqüenciamento de ESTs.

As seqüências armazenadas no Genbank encontram-se tradicionalmente divididas em "divisões" que correspondem aproximadamente a grupos taxonômicos tais como bactérias (BCT), vírus (VRL), primatas (PRI) e roedores (ROD). Nos últimos anos, novas divisões foram adicionadas a fim de dar suporte a seqüências provenientes de estratégias específicas de seqüenciamento. Dentre estas podemos destacar as divisões para EST (dbEST), STS (dbSTS)[6], seqüências *high-throughput* (dbHTG) e seqüências *high-throughput* provenientes de moléculas de cDNA (dbHTC).

O GenPept (APWEILER; BAIROCH; WU, 2004) (Genbank Gene Products Data Bank) contém traduções efetuadas a partir das seqüências armazenadas no Genbank. As entradas armazenadas contêm um mínimo de informação anotada, proveniente principalmente dos correspondentes registros de nucleotídeos.

2.2.3.2 COG

O banco de dados COG (*Clusters of Orthologous Groups*) (TATUSOV et al., 2003) é uma tentativa de classificar filogeneticamente genes presentes em genomas completos. Cada registro no COG contém três ou mais genes considerados ortólogos. A versão atual do COG consiste de 4873 grupos, incluindo 136.711 proteínas relacionadas a genes de 66 genomas. As informações presentes no banco de dados são atualizadas periodicamente, sempre que novos genomas são disponibilizados.

2.2.3.3 Interpro

O InterPro (MULDER et al., 2003) é uma fonte integrada de documentação sobre domínios, famílias e características funcionais de proteínas. Criado em 1999, seu objetivo é reunir em uma única fonte todos os principais bancos de *assinaturas*[7] de proteínas, dentre eles o PROSITE e o Pfam (BATEMAN et al., 2002). O InterPro foi desenvolvido com o intuito de racionalizar o processo de identificação de famílias de proteínas através da criação de um recurso coerente de diagnóstico e documentação.

Os dados provenientes dos bancos de dados membros são integrados manualmente a intervalos regulares por uma equipe de biólogos, cujo papel inclui também anotar entradas novas e já existentes. Cada uma destas entradas é descrita por uma ou mais assinaturas, correspondendo a algum domínio, família, repetição ou algum outro padrão biologicamente significativo.

2.2.4 Busca em Bancos de Dados Genéticos

A busca por similaridade em bancos de dados permite determinar quais das centenas de milhares de seqüências presentes nestes bancos estão potencialmente relacionadas a uma seqüência particular. Neste tipo de busca, a operação básica consiste em alinhar iterativamente uma seqüência de consulta a cada seqüência presente no banco de dados analisado, determinando o grau de similaridade entre ambas.

Os atuais bancos de seqüências são imensos e continuam a crescer em uma taxa exponencial. Isto torna a aplicação de métodos ótimos de alinhamento, baseados em progra-

[6] Sequence Tagged Sites

[7] Padrões encontrados na seqüência de amino-ácidos que identificam alguma característica especial.

Tabela 2.3: Variantes do BLAST

Programa	**Consulta**	**Banco de Dados**
BLASTP	proteína	proteína
BLASTN	nucleotídeo	nucleotídeo
BLASTX	nucleotídeo(traduzido)	proteína
TBLASTN	proteína	nucleotídeo(traduzido)
TBLASTX	nucleotídeo(traduzido)	nucleotídeo(traduzido)

mação dinâmica (SMITH; WATERMAN, 1981) impraticáveis para buscas nestes bancos de dados. Com isso, torna-se necessária a utilização de métodos heurísticos, que fazem uso de aproximações para aumentar significativamente a velocidade das comparações.

Um dos métodos heurísticos mais utilizados é baseado na estratégia de quebrar uma seqüência em pequenos pedaços de nucleotídeos ou amino-ácidos consecutivos, chamados *palavras*. A idéia básica é que um alinhamento representando um relacionamento verdadeiro entre seqüências contém ao menos uma palavra comum a ambas as seqüências. Estes casamentos entre palavras em duas seqüências são chamados *word hits*. *Word hits* podem ser identificados rapidamente pela pré-indexação de todas as palavras de uma seqüência de consulta. Este índice é acessado toda vez que a seqüência de consulta é comparada a uma entrada diferente no banco de dados.

2.2.4.1 BLAST

A família de programas BLAST (ALTSCHUL et al., 1990) é sem dúvida o conjunto de programas de análise mais utilizados em projetos de anotação e reanotação automáticas. O BLAST em si é um algoritmo de comparação de seqüências otimizado utilizado na busca em bancos de seqüências biológicas por alinhamentos locais ótimos relativos a uma dada seqüência desejada.

A busca inicial é feita por uma palavra de comprimento W, retirada da seqüência de consulta, que alcance um *score* de no mínimo T, quando comparada a seqüências no banco de dados, de acordo com uma *matriz de substituição*[8]. *Hits* de palavras em seqüências armazenadas são então estendidos em ambas as direções, na tentativa de gerar um alinhamento que exceda um dado score S. O parâmetro T dita a velocidade e a sensibilidade da busca. Um dos principais parâmetros especificados no BLAST é o valor E, responsável por filtrar o resultado da busca. O parâmetro E mede a probabilidade de que a similaridade identificada entre duas seqüências seja aleatória. Portanto, quanto mais próximo de zero, maior o grau de similaridade entre as seqüências. No caso de uma busca utilizando o BLAST, apenas os resultados que contiveram um valor de E menor que o especificado serão reportados.

Existem diversas variantes do BLAST, cada uma diferenciada das demais pelo tipo da seqüência (nucleotídeos ou proteína) da consulta e das seqüências armazenadas no banco de dados alvo. A Tabela 2.3 mostra um breve resumo destas variantes.

O BLASTP compara uma seqüência de amino-ácidos de uma proteína a seqüências pertencentes a um banco de dados contendo proteínas. O programa correspondente para seqüências de nucleotídeos é o BLASTN. Se os tipos das seqüências diferem, a seqüên-

[8]Uma matriz de subsituição contém valores proporcionais à probabilidade de que o amino-ácido i sofra uma mutação e se transforme no amino-ácido j para todos os pares de amino-ácidos possíveis. Em termos biológicos, a matriz de substituição indica se a troca de um amino-ácido por outro na cadeia de uma proteína pode levar a uma alteração significativa em sua função.

cia de DNA é traduzida pelo programa e é comparada a cada seqüência de amino-ácidos. O BLASTX compara uma seqüência de DNA a um banco de proteínas, o que é útil na análise de novos dados sobre seqüências e ESTs (*Expression Sequence Tags*). Para uma busca envolvendo uma seqüência de consulta de amino-ácidos e um banco de seqüências de nucleotídeos utiliza-se o programa TBLASTN. Este programa é útil na busca de regiões codificantes não anotadas, presentes em um banco de dados. Uma variante final, o TBLASTX, toma uma seqüência de DNA e realiza uma comparação com um banco de seqüências de DNA, traduzindo ambas e comparando-as como proteínas. Este programa é útil na comparação entre ESTs, onde existe uma suspeita de que uma das seqüências possua um potencial de codificação, mesmo que a exata região codificadora não tenha sido determinada.

3 ANOTAÇÃO E RE-ANOTAÇÃO DE GENOMAS

As moléculas de DNA presentes na célula dos organismos vivos contêm uma quantidade considerável de informação. A análise da organização estrutural e funcional desta informação é chamada *Anotação*.

A partir dos conceitos apresentados no capítulo anterior, podemos definir o processo de anotação como a tarefa de identificar os genes e suas respectivas funções dentro da célula. A identificação de tais informações exige a análise de quantidades imensas de dados, o que torna a sua execução manual proibitiva.

Logo, os computadores são ferramentas importantíssimas no processo de anotação. Diversas ferramentas computacionais foram desenvolvidas com a finalidade de executar tarefas bastante específicas na análise de um genoma (identificação de genes, comparação entre seqüências, pesquisa em bases de dados, ...).

Informações provenientes de anotações realizadas em centenas de organismos encontram-se armazenadas em bancos de dados publicamente disponíveis. Tais bancos, como o *Genbank*, contém dezenas de milhares de registros e são uma importante fonte de comparação entre seqüências genéticas, tarefa importantíssima na anotação genética.

Basicamente, um processo de anotação completo pode envolver três passos:

- *Análise em nível de nucleotídeos*, no qual o objetivo é identificar a localização de possíveis genes dentro do genoma;
- *Análise em nível de proteínas*, no qual procura-se determinar o produto dos genes identificados na etapa anterior;
- *Análise em nível de processos*, no qual procura-se determinar as vias metabólicas e os processos celulares aos quais os genes estão relacionados.

Os processos envolvidos em cada um destes níveis serão detalhados nas próximas sessões.

3.1 Identificando Genes e Regiões Codificantes

Uma das etapas mais importantes realizadas durante o processo de análise do genoma de um organismo é a detecção de regiões codificantes e não-codificantes. A grande maioria dos métodos utilizados atualmente tenta detectar a presença de um gene a partir do conjunto de *Open Reading Frames* (ORFs) encontradas no genoma. Uma ORF é uma subseqüência do DNA localizada entre dois *stop codons* que contém uma possível região codificadora, representada nas seis possíveis leituras efetuadas a partir da seqüência

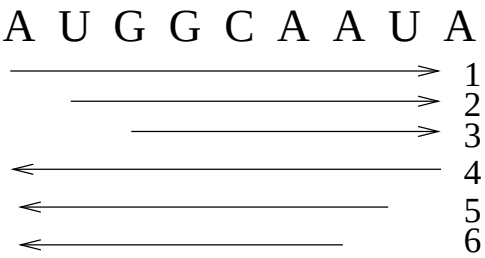

Figura 3.1: As 6 possíveis interpretações presentes em uma ORF.

identificada, como pode ser visto na Figura 3.1. As linhas 1-3 identificam as leituras efetuadas a partir da primeira, segunda ou terceira base respectivamente. As outras três linhas mostradas na figura identificam as leituras efetuadas no sentido inverso considerando a seqüência complementar, pois não sabemos qual fita de DNA poderá conter realmente uma região codificadora. Cada uma destas seqüências leva a seqüências de amino-ácidos diferentes, e as seis opções são consideradas simultaneamente durante o processo de análise.

Basicamente, os métodos de previsão de genes podem ser agrupados em três categorias principais:

1. *Métodos baseados em conteúdo*: Estes métodos consideram características como a freqüência de utilização de determinados códons, a periodicidade de repetições e a complexidade do conteúdo da seqüência.

2. *Métodos baseados em padrões*: Buscam pela presença ou ausência de padrões ou seqüências reguladoras. Estes métodos são utilizados para a detecção de características como *start* e *stop codons*, cadeias *polyA*, etc.

3. *Métodos comparativos*: Comparam a seqüência analisada com outras seqüências presentes em bancos de dados, com o objetivo de determinar se uma seqüência já conhecida está presente em uma dada região analisada.

Cada método de previsão é melhor utilizado em problemas práticos distintos. Não é raro que dois ou mais métodos sejam utilizados em conjunto, a fim de que características diferentes de uma seqüência sejam determinadas.

Diversos softwares sofisticados têm sido desenvolvidos para lidar com a detecção de genes em genomas eucarióticos e procarióticos. Dentre as técnicas computacionais mais utilizadas, podemos citar as Redes Neurais Artificiais, os Sistemas Baseados em Regras e os Modelos Ocultos de Markov (HMM). A abordagem HMM têm a vantagem de modelar explicitamente como as probabilidades individuais de uma seqüência de características são combinadas em uma estimativa de probabilidade para o gene inteiro. Estas técnicas são utilizadas basicamente por métodos baseados em conteúdo e por métodos baseados em padrões.

Métodos comparativos utilizam uma abordagem baseada na busca por similaridade, na qual o BLAST se destaca como principal ferramenta. Nessa abordagem, compara-se a seqüência analisada com outras seqüências já anotadas, em busca de similaridades. Semelhanças com genes já anotados são uma boa evidência de que uma região pertence a um gene. Abordagens que consideram similaridade se mostraram bastante eficientes, como pode ser observado em (REESE et al., 2000).

3.1.1 Glimmer

O Glimmer (DELCHER et al., 1999) é um programa de identificação de genes em organismos procariontes desenvolvido com base em uma técnica denominada *Interpolated Markov Models*. A grande diferença entre IMMs e os modelos HMM tradicionais consiste no fato de que os primeiros combinam probabilidades relacionadas a subseqüências de tamanho variável, enquanto os últimos utilizam um modelo de probabilidades associado a todas as subseqüências de tamanho fixo.

Estas subseqüências estão relacionadas a determinados contextos na seqüência de DNA e são utilizadas para calcular a probalidade de que o próximo nucleotídeo faça parte de um gene. Logo é necessária uma etapa de treinamento, anterior ao processo de identificação de genes propriamente dito, para que os modelos probabilísticos sejam construídos.

Depois de criar os IMMs, o GLIMMER utiliza-os então para avaliar o conjunto de ORFs presentes no genoma do organismo analisado. Ao final do processo, um conjunto de prováveis genes é retornado pelo sistema.

O GLIMMER tem se mostrado bastante eficiente na análise de genomas procariontes, alcançando taxas de acerto acima dos 95%. Outra alternativa ao Glimmer para genomas procariontes é o Genemark (LUKASHIN; BORODOVSKY, 1998), baseado em modelos HMM tradicionais.

3.1.2 GRAIL

O GRAIL (XU et al., 1994) (*Gene Recognition and Analysis Internet Link*) foi uma das primeiras ferramentas desenvolvidas para detecção de genes em organismos eucariontes. O núcleo do sistema utiliza uma rede neural para reconhecer potenciais regiões codificantes, dentro de uma janela deslocada sobre a seqüência analisada.

A primeira versão disponibilizada deste programa continha janelas de tamanho fixo (100 bases) e ignorava informações de contexto relacionadas a janela analisada (como *start* e *stop* codons e outros sinais). Já a versão 2 do sistema passou a utilizar informações de contexto e janelas de tamanho variável.

A versão mais recente, GRAIL-EXP, passou a incluir também análises comparativas com seqüências já identificadas. A inclusão desta etapa melhorou consideravelmente a performance do programa original.

3.1.3 Genscan

O Genscan (BURGE; KARLIN, 1997) foi projetado para prever estruturas completas de genes em genomas eucariontes. O sistema utiliza um modelo probabilístico da composição genética e da estrutura dos genes. Quando busca por descrições estruturais de genes que sejam equivalentes ou consistentes com a seqüência consultada, o algoritmo pode associar uma medida de probabilidade, como a chance que um dado trecho da seqüência represente um exon, por exemplo. Os exons que possuírem a probabilidade mais alta serão selecionados. O método também seleciona exons subótimos, que possuem uma probabilidade aceitável mas não tão boa quanto as anteriores, representando alternativas adicionais na análise de determinada região do genoma por parte de especialistas humanos.

3.2 Analisando Produtos e Funções de Seqüências

O próximo passo, após identificar os genes, é compilar um catálogo de proteínas produzidas por um organismo e a influência de cada uma nas diversas atividades desempenhadas pelas células. Este é o objetivo principal deste nível da anotação.

Utiliza-se novamente a busca por similaridade como ferramenta básica para a identificação da proteína. Mais precisamente, busca-se classificar proteínas desconhecidas em grupos ou famílias e identificar similaridades com proteínas anotadas de outras espécies.

O problema envolvido neste processo está na natureza do processo evolucionário. Durante a evolução de uma família de proteínas, uma proteína ancestral é duplicada uma ou mais vezes e as cópias divergem, formando um grupo de proteínas relacionadas. Entretanto, similaridades funcionais não garantem o compartilhamento de um ancestral comum, e existem muitos casos de proteínas classificadas em uma mesma família que têm funções divergentes.

Uma típica seqüência de anotação de proteínas irá buscar primeiramente similaridades com outras proteínas já anotadas. Uma abordagem complementar é a busca contra bancos de dados de *domínios funcionais*, que armazenam estruturas de motivos.

O maior desafio na anotação de um genoma é associá-lo aos processos biológicos ocorridos dentro de uma célula. Esta última etapa visa identificar como os genes e as proteínas se associam ao ciclo celular, metabolismo e à manutenção da vida.

Processos como os ocorridos nas chamadas *vias metabólicas* podem conter diversos genes ativados simultaneamente ou em uma seqüência definida, que possibilitam que determinada ação seja realizada pela célula. Alterações nos genes podem levar a falta de proteínas importantes para a reação, impossibilitando a realização de um ou mais processos.

Uma das principais técnicas utilizadas visa analisar mutações ocorridas em genes, e verificar se estas mutações influenciam a operação do gene. Em caso positivo, é possível determinar se um gene participa ou não de determinado processo, analisando quais ações deixaram de ocorrer em detrimento da mutação. O produto final desta análise é uma possível rede de relações funcionais entre genes, denominada *rede genética*, representando diversas vias metabólicas existentes.

A anotação a nível de processos transcende o trabalho puramente computacional. Outras técnicas não computacionais são utilizadas, como análise da expressão de *microarrays*, interferência de RNA, etc.

3.3 Ferramentas Integradas de Anotação

Reunir todas as ferramentas computacionais necessárias para efetuar uma determinada análise sobre um genoma e integrá-las de forma correta é um processo muitas vezes realizado manualmente. Geralmente, o tempo gasto para tal demanda uma boa parte do tabalho efetuado.

A necessidade de sistemas que automatizem a utilização integrada de todas estas ferramentas é uma necessidade evidente, a fim de que o tempo gasto com este processo seja minimizado. Nesta seção são citados alguns sistemas que integram diversas ferramentas de bioinformática em um único sistema de anotação, fornecendo uma interface unificada e integrada ao usuário.

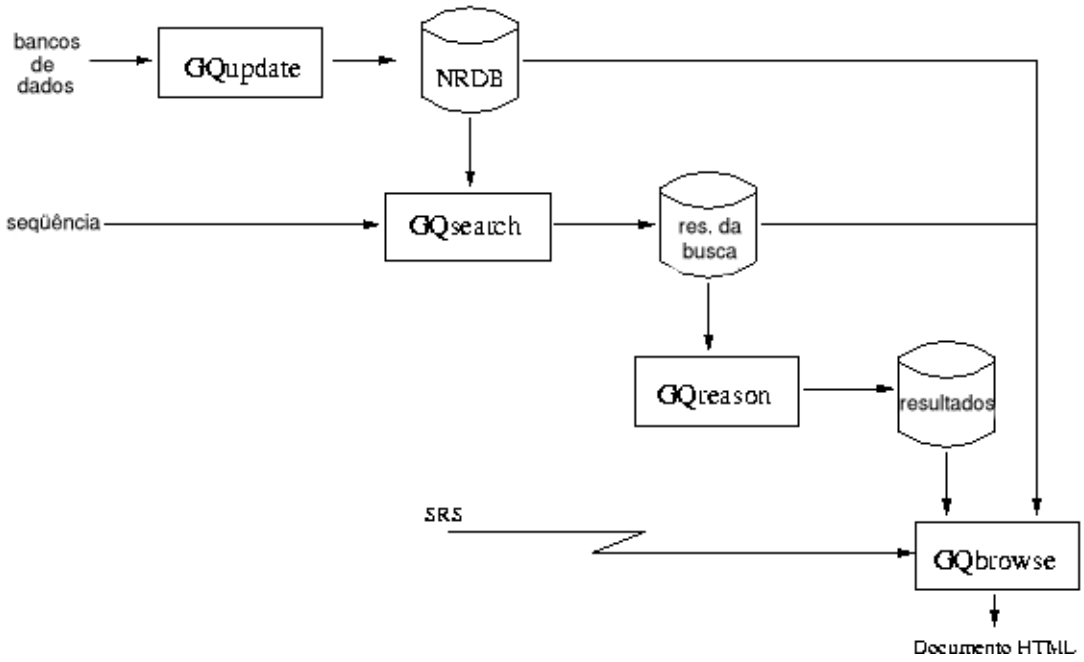

Figura 3.2: Esquema básico do GeneQuiz, mostrando os quatro módulos utilizados.

3.3.1 O Sistema GeneQuiz

O GeneQuiz(ANDRADE et al., 1999) é um sistema semi-autônomo de análise de seqüências de proteínas. O principal propósito do sistema é identificar uma proteína, através principalmente da análise de similaridades com outras seqüências.

O sistema é composto de quatro módulos: GQupdate, GQsearch, GQreason e GQbrowse, como mostrado na Figura 3.2. Os módulos foram desenvolvidos em Perl, e a interface para visualização dos resultados pode ser acessada via navegadores web. Estes módulos desempenham diversas tarefas repetitivas de forma automática, como atualização de bases de dados, busca por seqüências similares, integração de resultados de uma maneira uniforme e a autônoma avaliação e interpretação dos resultados utilizando o conhecimento de especialistas, codificado em regras.

O módulo GQupdate é responsável por manter integrados e atualizados diversos bancos de dados locais não redundantes, formados por seqüências de proteínas e nucleotídeos, derivadas de um conjunto de bancos de dados públicos(dentre eles, SWISS-PROT, PIR, TrEMBL, GenBank, PROSITE e PDB). Este módulo opera como um processo autônomo, executando atualizações diariamente. Quando uma nova seqüência é adicionada, um banco de dados não redundante(NRDB) de proteínas ou nucleotídeos é atualizado automaticamente.

Uma sessão do GeneQuiz é iniciada pela entrada, por parte do usuário, de uma seqüência de proteínas a ser analisada pelo sistema. O módulo GQsearch recebe esta seqüência, e aplica uma série de ferramentas de análise sobre ela, disponibilizando os resultados para os estágios subseqüentes de processamento. O conjunto de métodos é executado em uma ordem pré-determinada, baseada em um arquivo de configuração especificando (i) as dependências entre métodos, (ii) argumentos e os dados de entrada exigidos por cada método, e (iii) o armazenamento da saída de cada método pelo sistema.

Os métodos podem ser divididos em três categorias, de acordo com seu papel no GeneQuiz:

1. *Filtros de seqüências*: Utilizados para mascarar partes da seqüência que afetam de forma negativa a performance dos métodos de busca de seqüências(tais como regiões de baixa complexidade).

2. *Métodos de comparação*: Aplicados para a obtenção automática de uma anotação funcional, através da busca por proteínas homólogas, utilizando programas de busca por similaridade, como o BLAST.

3. *Métodos de suporte*: Executados sobre a seqüência para fornecer ao usuário informações adicionais para confirmar ou não a anotação automática. Tais métodos incluem detecção de padrões (repetições e motivos), alinhamento múltiplo de seqüências e a inferência da estrutura tridimensional da proteína.

O módulo GQreason possui principalmente dois propósitos: (i) determinar a função celular geral referente à família da seqüência de entrada analisando o conjunto de homólogos da proteína; e (ii) associar uma função específica à própria seqüência de entrada, se possível, pela análise das seqüências homólogas encontradas. Ambas as tarefas dependem da escolha cuidadosa dos homólogos e da análise sistemática das anotações presentes nos bancos de seqüências.

A lista de homólogos é computada a partir da união dos resultados reportados pelos programas de busca do estágio GQsearch anterior. Já a extração sistemática de informação funcional a partir das anotações expressas em vários campos e formatos específicos é um problema nada trivial. Critérios como o grau de similaridade entre as seqüências, a qualidade dos bancos de dados pesquisados e a qualidade das anotações verificadas afetam diretamente a qualidade da anotação final gerada.

A computação da classe funcional geral associada à seqüência e sua família de homólogos é determinada através da análise de palavras-chave presentes nas anotações armazenadas no NRDB. O método de classificação utilizado cria um dicionário de palavras-chave, criado a partir de regras que associam estas palavras-chave a classes funcionais. Finalmente, o módulo GQbrowser permite ao usuário consultar os resultados do processo de anotação através de um browser.

3.3.2 O Sistema SABIA

O SABIA(ALMEIDA et al., 2004)(*System for Automated Bacterial Integrated Annotation*) é uma ferramenta desenvolvida pela equipe do Laboratório de Bioinformática do LNCC (Laboratório Nacional de Computação Científica), utilizada para montagem e anotação de genomas de bactérias. O sistema permite automatizar diversos processos, tais como a análise e identificação de ORFs, análise de regiões extragenéticas e identificação de vias metabólicas. Diversas ferramentas computacionais estão integradas ao sistema, incluindo Glimmer, Genemark e BLAST. A arquitetura geral do sistema pode ser vista na Figura 3.3.

O sistema é composto basicamente de dois módulos: um módulo de montagem e um módulo de anotação. O módulo de montagem é responsável por receber diversas seqüências provenientes do seqüenciamento de um genoma a ser analisado. Estas seqüências são reunidas com o objetivo de se obter a seqüência completa do genoma desejado. Estatísticas sobre a qualidade das seqüências informadas e sobre os resultados obtidos após o processo de montagem podem ser consultadas, permitindo que possíveis problemas sejam detectados e possivelmente corrigidos.

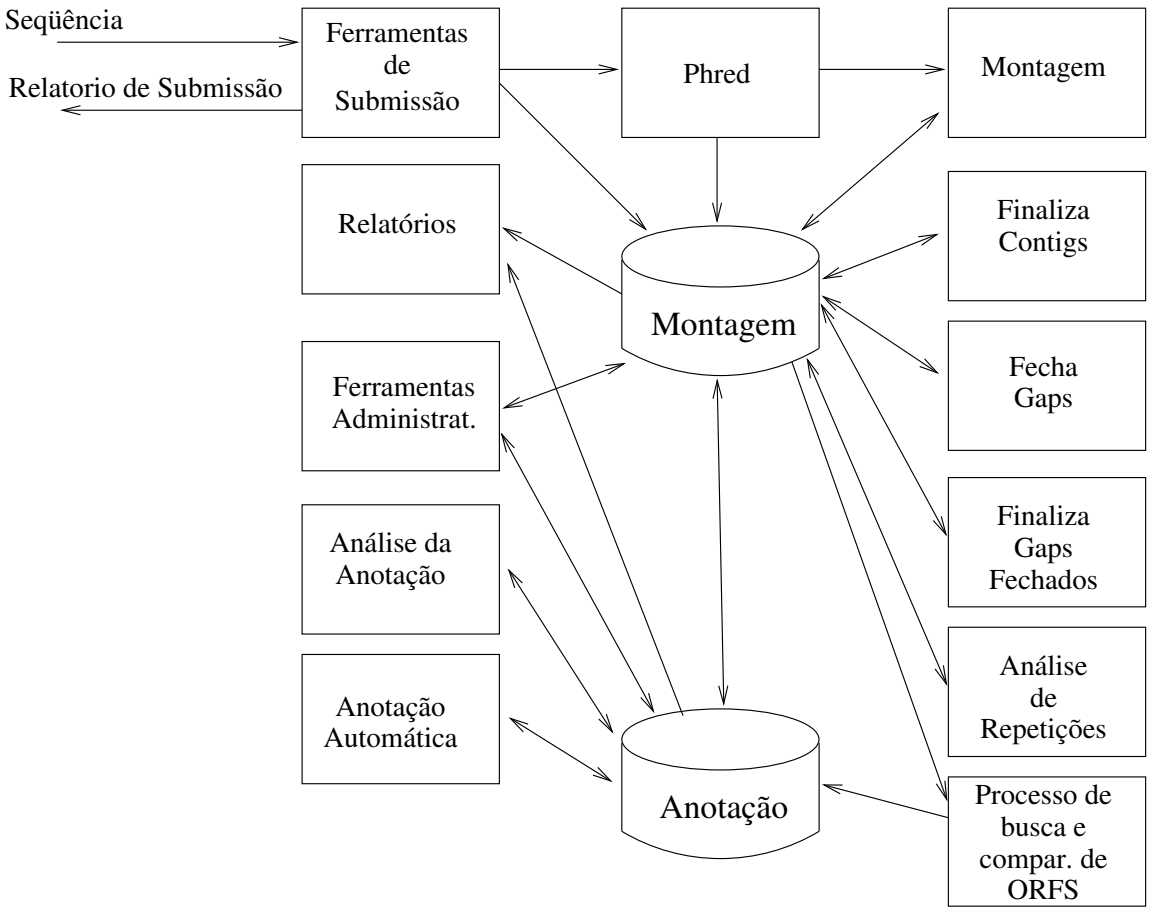

Figura 3.3: Arquitetura básica do SABIA.

O módulo de anotação é responsável pela identificação e caracterização funcional das ORFs encontradas no genoma. A primeira fase da anotação toma as seqüências identificadas na etapa anterior e procura determinar automaticamente as ORFs e outras estruturas do genoma utilizando programas como o Glimmer e o Genemark. Para cada ORF encontrada, o SABIA identifica diversas informações, dentre elas a posição no genoma, as respectivas seqüências de nucleotídeos e amino-ácidos, pontos isoelétricos(PI), peso molecular e informação sobre regiões extragenéticas, tais como promotores e *start codons* opcionais.

Além disso, a classificação funcional de cada ORF relativa aos bancos de dados KEGG (KANEHISA et al., 2004) e COG é obtida e pode ser visualizada. Tais informações incluem o organismo, nome do gene, grupos funcionais e vias metabólicas, obtidas através de uma consulta BLAST. Outros ferramentas também podem ser utilizadas, como o InterPro. A seqüência de amino-ácidos pode ser utilizada como entrada para um processo que tenta prever a localização celular de uma proteína.

Como recursos adicionais, o sistema ainda possibilita a comparação entre genomas analisados(verificando genes presentes em diversos organismos em análise) e a busca em regiões não codificantes, utilizando BLAST, para identificação de possíveis ORFs não reconhecidas durante o processo inicial de análise.

3.4 Anotação Automática e Inteligência Artificial

Alguns sistemas de anotação automática vêm utilizando técnicas de Inteligência Artificial para aprimorar a utilização do conhecimento obtido através de bancos de dados biológicos. Essas técnicas envolvem algoritmos de *data mining* e aprendizado de máquina, bem como a tecnologia de sistemas multiagentes, para a organização dos módulos do sistema.

3.4.1 O Sistema MASKS

O sistema MASKS (SCHROEDER; BAZZAN, 2002) é um ambiente de anotação automática cujo objetivo é preencher o campo *Keywords*, conforme definido no banco de dados Swiss-Prot. O sistema utiliza diferentes algoritmos de aprendizado de máquina encapsulados em agentes para classificar os dados. Seu objetivo é melhorar o aprendizado simbólico através da troca de conhecimento entre estes agentes.

A arquitetura dos agentes utilizados consiste de cinco componentes, como mostrado na Figura 3.4. A unidade de *Performance* tem a tarefa de controlar, monitorar e guiar o progresso da unidade de aprendizado. Esta unidade é responsável pela entrada e saída de dados interna e também externa, com outros agentes e o ambiente.

A unidade de *Aprendizado* contém o indutor de regras, responsável pelo aprendizado. Este aprendizado acontece em dois estágios. O primeiro estágio é dedicado ao aprendizado individual, onde o indutor de regras é aplicado ao arquivo de treinamento.

As regras geradas são avaliadas pela unidade de *Avaliação*. Esta unidade aplica as regras geradas a um arquivo de teste informado, medindo a qualidade de cada uma através da função de avaliação de regras do algoritmo CN2(CLARK; NIBLETT, 1989). As regras que atingirem uma pontuação igual ou superior a um *threshold* informado são armazenadas na *Base de Conhecimento*.

O segundo estágio do processo de aprendizado consiste do aprendizado cooperativo. A entrada deste estágio consiste das bases de conhecimento contendo o conhecimento obtido durante o aprendizado individual. A primeira tarefa executada pelos agentes nesta

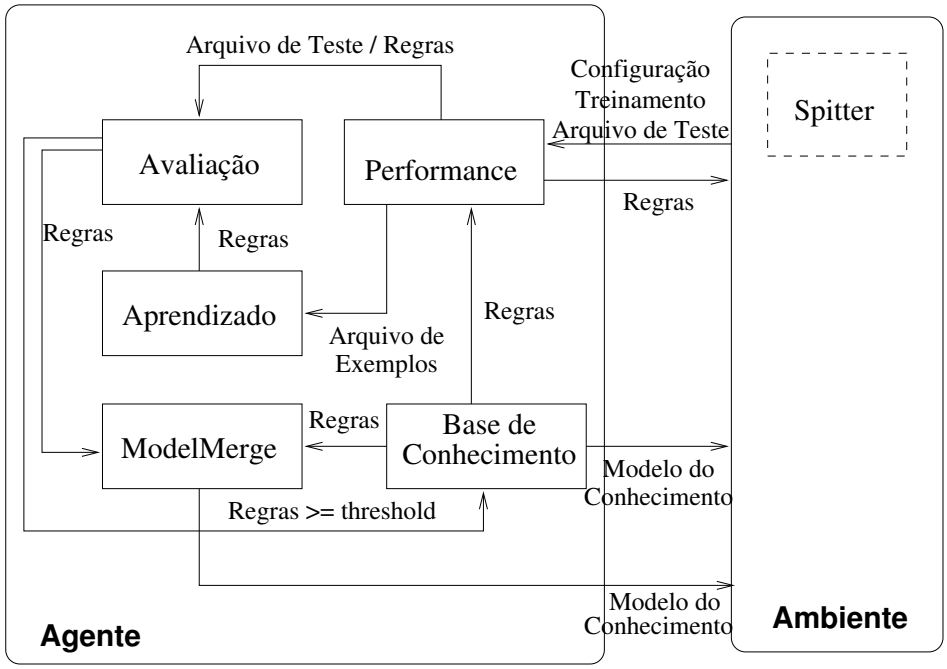

Figura 3.4: Arquitetura dos agentes no MASKS.

nova etapa é consultar as bases de conhecimento de outros agentes. O agente então procura por regras equivalentes às presentes em sua própria base de conhecimento, selecionando aquelas que possuem um nível de qualidade superior. As regras selecionadas são armazenadas no componente *ModelMerge*. No final do processo, o agente copia as regras presentes na base de conhecimento local não modificadas para o ModelMerge.

Neste ponto o agente possui dois modelos distintos sobre o domínio do problema (a base de conhecimento local e o ModelMerge). O modelo que cobrir o maior número de instâncias presentes no arquivo de teste é selecionado como o modelo final para o agente. A saída gerada pelo ambiente conterá o melhor modelo dentre todos os agentes.

3.4.2 O Sistema GeneWeaver

O GeneWeaver é um sistema multiagente desenvolvido para lidar com muitos dos problemas no domínio da análise de genomas e previsão de estruturas de proteínas. Os agentes no sistema podem estar encarregados do gerenciamento de bancos de dados primários, análise de seqüências utilizando ferramentas existentes, ou de armazenar e apresentar os resultados gerados. Um ponto importante a notar é que o sistema não oferece novos métodos para executar estas tarefas, mas organiza as ferramentas existentes de forma a atingir uma operação mais flexível e efetiva(BRYSON et al., 2000).

A arquitetura geral do sistema e os diferentes tipos de agentes podem ser vistos na Figura 3.5. No lado esquerdo, os agentes PDB, Swiss e PIR gerenciam os bancos de seqüências indicados por seus nomes, e interagem com o agente NRDB, que combina os dados enviados em um banco de dados não-redundante de proteínas. No lado direito da figura, os agentes de cálculo (incluindo os agentes BLAST e CLUSTAL que executam tarefas de análise específicas) tentam anotar as seqüências presentes no banco de dados utilizando diversos programas de bioinformática. Um agente de cálculo especialista pode combinar as habilidades dos outros agentes de cálculo utilizando o conhecimento de especialistas codificado em planos. Neste caso, ele pode utilizar o agente BLAST para

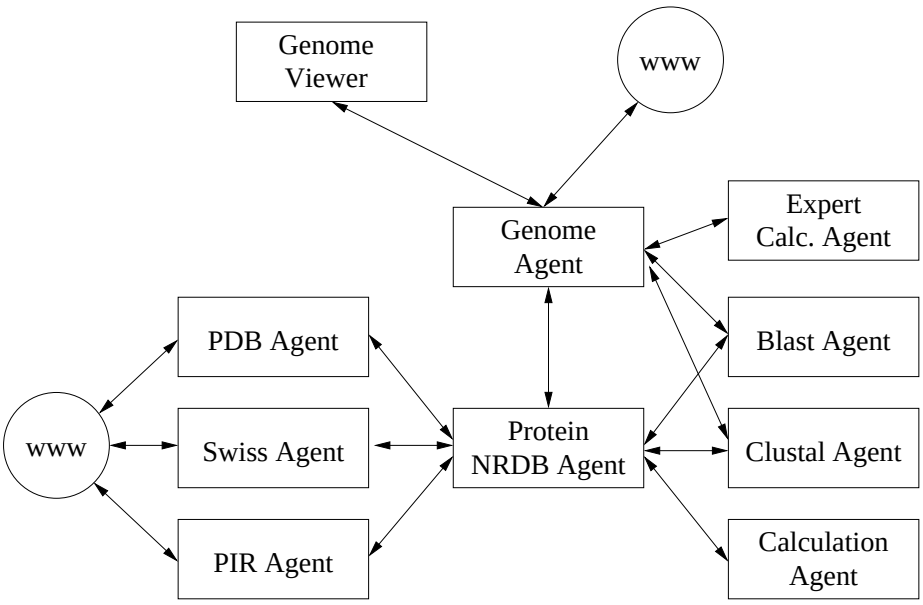

Figura 3.5: Comunidade de Agentes do GeneWeaver.

encontrar proteínas similares e então utilizar o agente CLUSTAL para comparar as proteínas obtidas mais precisamente. Finalmente, os resultados gerados pelo sistema podem ser acessados externamente através do agente de Genoma. Esta interface externa com o usuário é realizada via Internet.

Existem cinco tipos de agentes presentes no GeneWeaver:

- *Broker agents*, que não são mostrados na figura, funcionam como facilitadores, necessários para registrar informações sobre outros agentes na comunidade.
- *Agentes de bancos de dados primários*, que gerenciam bancos de dados primários remotos, e mantém os dados contidos neles atualizados e em um formato que permita a outros agentes consultá-los.
- *Agentes de bancos de dados não-redundantes*, que possuem a tarefa de construir e gerenciar bancos de dados não-redundantes a partir dos dados informados por outros agentes na comunidade, que lidam com bancos de dados primários.
- *Agentes de cálculo*, que encapsulam algum método ou ferramenta pré-existente para análise dos dados de uma seqüência. Estes agentes tentam determinar a estrutura ou função desta seqüência. Alguns destes agentes possuem o conhecimento de especialistas codificado em planos, o que possibilita a eles executarem tarefas avançadas utilizando outros agentes de cálculo.
- *Agentes genoma*, responsáveis por gerenciar a informação genética sobre um organismo particular.

Cada agente no GeneWeaver compartilha uma arquitetura comum, composta por diversos módulos internos e repositórios externos de dados que são utilizados para armazenagem de dados manipulados, ou os programas de análise utilizados para obtenção de informações. A Figura 3.6 mostra a arquitetura básica dos agentes. Essencialmente, tudo gira em volta do módulo de *controle*, que é direcionado pelo módulo de *motivação* através

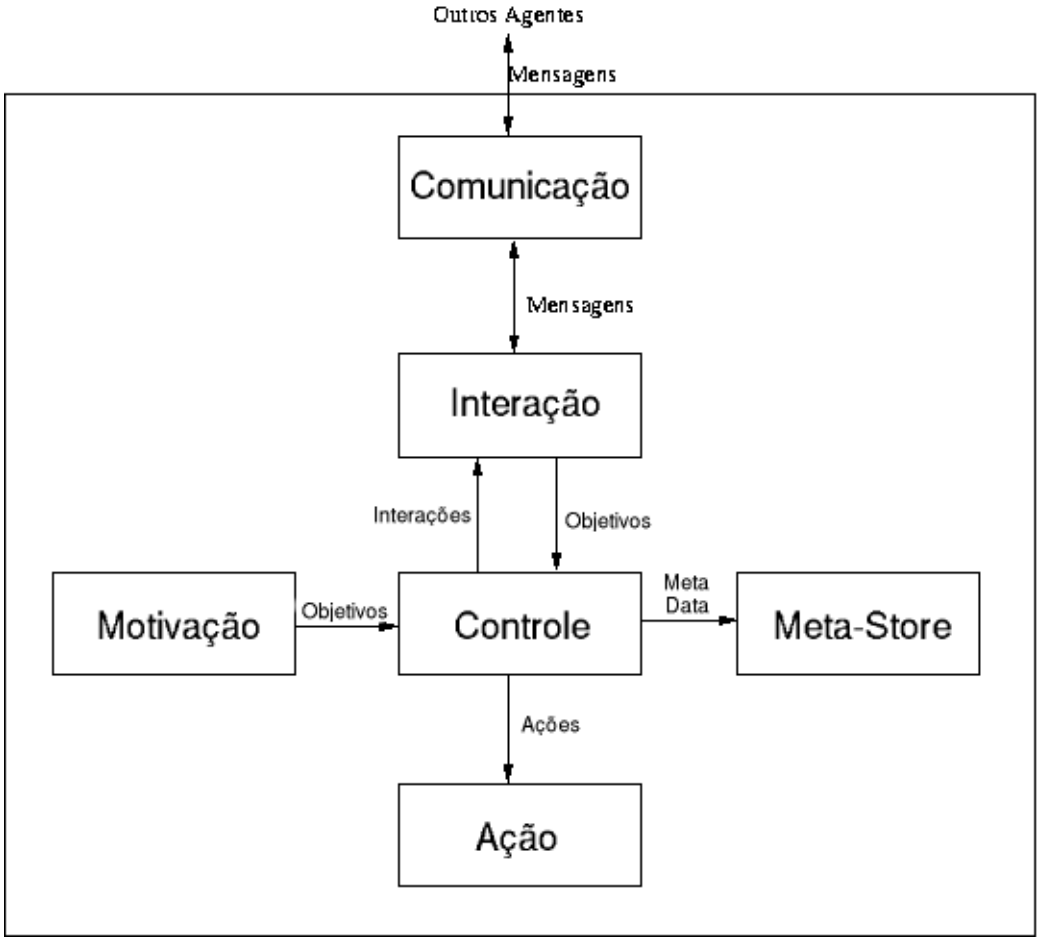

Figura 3.6: Arquitetura básica dos agentes no GeneWeaver.

de objetivos particulares, a partir dos quais então decide a melhor forma de alcançá-los. Estes objetivos são realizados através de ações executadas pelo módulo de *ação*, ou através de uma requisição de ajuda a outros agentes através do módulo de *interação*. Este módulo, por sua vez, utiliza o módulo de *comunicação* para os mecanismos de interação. O módulo *meta-store* simplesmente fornece um repositório para informação local tais como as habilidades de outros agentes.

3.5 Re-anotação de Genomas

Mesmo quando um projeto de anotação é completado e a informação sobre as seqüências analisadas é disponibilizada publicamente, é comum que ocorram processos de "revisão" dos resultados originais, devido a diversos motivos. A este processo de anotar seqüências previamente já anotadas dá-se o nome de *re-anotação*. A motivação para os processos de re-anotação incluem a descoberta de novos genes e novas funções relacionadas a estes genes, além do teste e comparação de novos métodos de anotação desenvolvidos. A reanotação de diversos genomas tem fornecido informações atualizadas, utilizando as mais recentes técnicas e recursos disponíveis para tal - como novos algoritmos e bancos de dados mais ricos (OUZOUNIS; KARP, 2002).

3.5.1 A Qualidade de um Processo de Anotação

Os projetos de reanotação realizados até o momento afirmam ter alcançado melhorias em relação aos resultados obtidos com os projetos de anotação originais. Mas nem sempre esta aparente "melhoria" dos resultados pode significar uma identificação mais precisa de um gene ou de sua função. Muitas vezes esses resultados são influenciados por uma tolerância menor na análise ou pela utilização de bancos de dados mais atualizados, mas pouco precisos. Assim, resultados superiores aos originais podem ser questionados, devido à subjetividade associada à análise efetuada.

Métodos de avaliação de anotações mais precisos são necessários. Tomando-se um conjunto de anotações corretamente completadas como padrão, pode-se definir duas medidas de exatidão. A primeira define uma medida de cobertura, que leva em conta a taxa de acertos sobre a soma de acertos e falsos negativos. Assim, se não há falsos negativos, a cobertura é de 100%. Na segunda medida, a precisão é definida como o número de acertos sobre a soma de acertos e falsos positivos. Assim, se não há falsos positivos, a precisão é de 100%.

O problema atualmente é de que estas medidas não possuem um padrão de anotação comprovadamente correto a seguir. Não existe nenhum genoma completo cujas estruturas e funções tenham sido completamente comprovados experimentalmente. Logo, todo o conhecimento adquirido até agora sobre a avaliação de processos de anotação é ainda relativo, não absoluto. Para tal fim seriam necessárias comparações entre diferentes abordagens de anotação genética, mas não se tem certeza ainda sobre a exatidão das previsões efetuadas.

3.5.2 Abordagens Utilizadas para Re-anotação

As abordagens utilizadas em re-anotação diferem pouco das utilizadas em projetos de anotação normais. As ferramentas utilizadas incluem basicamente programas de busca por similaridade como o BLAST. Outras técnicas, como análise filogenética também foram utilizadas em alguns projetos recentes de re-anotação (DANDEKAR et al., 2000).

Buscas por homólogos de seqüências já anotadas em versões mais recentes de bancos de dados públicos têm sido uma fonte bastante interessante de novas informações. Esta abordagem se beneficia de novas atualizações efetuadas nessas bases de dados para extrair novas informações sobre funcionalidades relativas a proteínas e genes. Algumas abordagens combinam diferentes programas de busca por similaridades para conseguir resultados ainda mais confiáveis, já que grande parte destes algoritmos são heurísticos.

Outra abordagem utilizada é re-analisar seqüências tidas como não-codificantes em processos anteriores de anotação, em busca de possíveis novos genes. Para esta tarefa utilizam-se novamente a busca por similaridade, a fim de encontrar genes ou proteínas identificadas que possam ser próximas a subseqüências pertencentes à região analisada. Esta alternativa também mostrou-se interessante na busca de novas regiões codificadores nos recentes projetos de re-anotação.

Para o futuro, novos métodos de análise de seqüências e busca por genes podem ser úteis para projetos de anotação e re-anotação. Além disso, a crescente atualização dos bancos de dados biológicos públicos com novas informações provenientes de análises realizadas em todo mundo tem aumentado ainda mais o conhecimento sobre os organismos e seus genomas. Este conhecimento é uma fonte valiosa de recursos para projetos de anotação e re-anotação, possibilitando que estes sejam mais eficazes em suas tarefas.

3.5.3 Genomas Re-anotados

Projetos de re-anotação para espécies individuais foram desenvolvidos por alguns grupos nos últimos anos. As espécies re-anotadas incluem: *Haemophilus influenzae* (TATUSOV et al., 1996), *Mycoplasma genitalium* (OUZOUNIS et al., 1996b), *Methanococcus jannaschii* (KYRPIDES et al., 1996), diversas especies do grupo *Archaea* (RAGHAVAN; OUZOUNIS, 1999), *Mycoplasma pneumoniae* (DANDEKAR et al., 2000), *Thermotoga maritima* (KYRPIDES et al., 2000), e casos isolados de genes individuais. Um dos fatos mais interessantes a emergir destes estudos é o nível de melhoria obtido nos resultados fornecidos pela re-anotação, calculado sobre o número de genes expressados para os quais novas funções são previstas como um percentual do número total de genes no genoma (OUZOUNIS; KARP, 2002).

3.5.3.1 Re-anotação do Genoma do Mycoplasma pneumoniae

Quatro anos após a anotação original(HIMMELREICH et al., 1996), o genoma completo do *Mycoplasma pneumoniae* foi re-anotado. Para realizar este processo, comparações com outros genomas foram realizadas (particularmente ao *M. genitalium*), além de buscas em bancos de dados atualizados utilizando ferramentas tais como o PSI-BLAST (ALTSCHUL et al., 1997). Para checar e testar os resultados obtidos, utilizou-se outros programas similares, como o HMM e FASTA, mas também ferramentas e métodos complementares, como análise de domínio, análise filogenética, análise de contexto e grupos de genes ortólogos, além da análise de genes duplicados. Técnicas experimentais não computacionais, como espectometria de massa e expressão de mRNAs também foram utilizadas.

Como resultados obteve-se a identificação de 12 novas proteínas, através da re-análise de regiões intergenéticas (2 proteínas identificadas por espectometria de massa sem função definida, seis proteínas hipotéticas e quatro com características funcionais previstas). Outras cinco ORFs foram descartadas, através de comparação com outras proteínas, por conterem pseudogenes. Como resultado final, obteve-se um acréscimo de 11 novas proteínas identificadas, chegando a um total de 688 proteínas.

3.5.3.2 Re-anotação do Genoma do Haemophilus influenzae

H. influenzae foi o primeiro organismo unicelular cuja conservação de proteínas pôde ser analisada no contexto de um genoma inteiro. As seqüências do genoma deste organismo submetidas ao Genbank incluem 1747 genes codificadores de proteínas previstos. Comparações efetuadas entre regiões intergenéticas e seqüências armazenadas no Genbank utilizando o programa BLASTX revelaram um alto número de similaridades, indicando que estas regiões podem conter genes.

Dadas estas incertezas, uma revisão do conjunto de proteínas codificadas pelo genoma do *H. influenzae* foi realizada. Este processo de re-anotação combinou buscas por seqüências similares com análises estatísticas da seqüência de DNA, analisada utilizando-se o programa Genemark. Como resultado ampliou-se o conjunto de genes codificadores para 1703, contendo 23 novas ORFs e 107 ORFs modificadas a partir da anotação original. Foram eliminados 47 genes, já que sua existência não foi corroborada pelos métodos aplicados.

3.5.3.3 Reanotação do Genoma do Thermotoga maritima

A reanotação efetuada a partir do genoma completo da bactéria *Thermotoga maritima* comparou 1877 ORFs encontradas originalmente com as correspondentes predições efetuadas mais recentemente. Após descartar todos os casos onde as duas análises independentes concordavam(985 casos), foram analisados em detalhe os casos onde ocorreram aparentes divergências (29 casos), bem como as proteínas hipotéticas encontradas(863 casos).

Para realizar tal análise foram utilizadas diversas ferramentas de previsão de função. A análise completa incluiu:

- Mais de cinco iterações do algoritmo PSI-BLAST.
- Presença de padrões do PROSITE.
- Verificação de famílias de proteínas relacionadas a seqüências através de buscas efetuadas nos bancos de dados Pfam e COG.
- Envolvimento em caminhos metabólicos.
- Vizinhança dos genes correspondentes no cromossomo.
- Organização em domínios funcionais através do banco de dados PRODOM.

A análise final demonstrou que 90% das associações funcionais efetuadas estavam iguais àquelas realizadas originalmente. No total, foram identificados 193 casos onde ocorreu conflito(10,3% do genoma inteiro), dos quais 164 continham novas funções identificadas e os 29 casos restantes eram correções de funções previstas anteriormente. O número total de associações funcionais subiu de 1014(54%) para 1178(63%), um acréscimo de 16% sobre os resultados originais.

4 O MODELO DE RE-ANOTAÇÃO

Neste capítulo será apresentado o modelo referente ao sistema de re-anotação automática desenvolvido neste trabalho. O modelo desenvolvido é baseado no conceito de agentes, de modo que a informação possa ser coletada e processada de forma automática pelos agentes.

A motivação deste trabalho reside no fato de que não existem, até o momento, sistemas integrados que auxiliem a re-anotação automática de genomas. Os poucos processos de re-anotação realizados até o momento realizam basicamente análise manual de dados obtidos a partir da execução isolada de ferramentas computacionais como o BLAST.

O principal objetivo deste modelo é, portanto, possibilitar que seqüências genéticas anotadas possam ser re-analisadas periodicamente de forma automática. A intervenção do usuário é solicitada no momento de selecionar as alterações válidas a partir de um conjunto de sugestões do modelo. Desse modo, os diversos agentes que compõem o sistema coletam a informação, realizam um processamento preliminar sobre ela e no final elaboram uma lista de alterações sugeridas que será enviada a um ou mais especialistas cadastrados.

4.1 Re-anotação Automática

Processos de re-anotação genômica ainda são pouco efetuados. Muitas partes de seqüências genéticas disponíveis ainda não se encontram anotadas ou suas anotações ainda podem ser revistas. A análise manual de dados provenientes de bases de dados disponíveis e de novas descobertas efetuadas é um processo lento, já que depende de processos demorados de análise e comprovação dos resultados, para que estes possam ser publicados e se tornem disponíveis.

Ferramentas construídas para agilizar e integrar as diversas etapas do processo de re-anotação ainda não existem. Os projetos de re-anotação desenvolvidos até hoje basicamente utilizaram ferramentas computacionais de forma isolada (por exemplo o BLAST) cujos resultados são analisados manualmente por especialistas.

A dificuldade maior na obtenção de ferramentas automáticas de re-anotação esbarra no mesmo problema encontrado na anotação automática: a pouca precisão obtida a partir de modelos genéticos computacionais. Apesar disso os resultados obtidos a partir de ferramentas automáticas pode ser um bom ponto de partida para a análise de especialistas humanos.

Técnicas computacionais provenientes da área de Inteligência Artificial podem auxiliar a obtenção de arquiteturas automáticas de re-anotação. Abordagens tais como os sistemas baseados em agentes podem facilitar a modelagem de sistemas distribuídos, automáticos e inteligentes de re-anotação. Sem dúvida o desenvolvimento de novas tec-

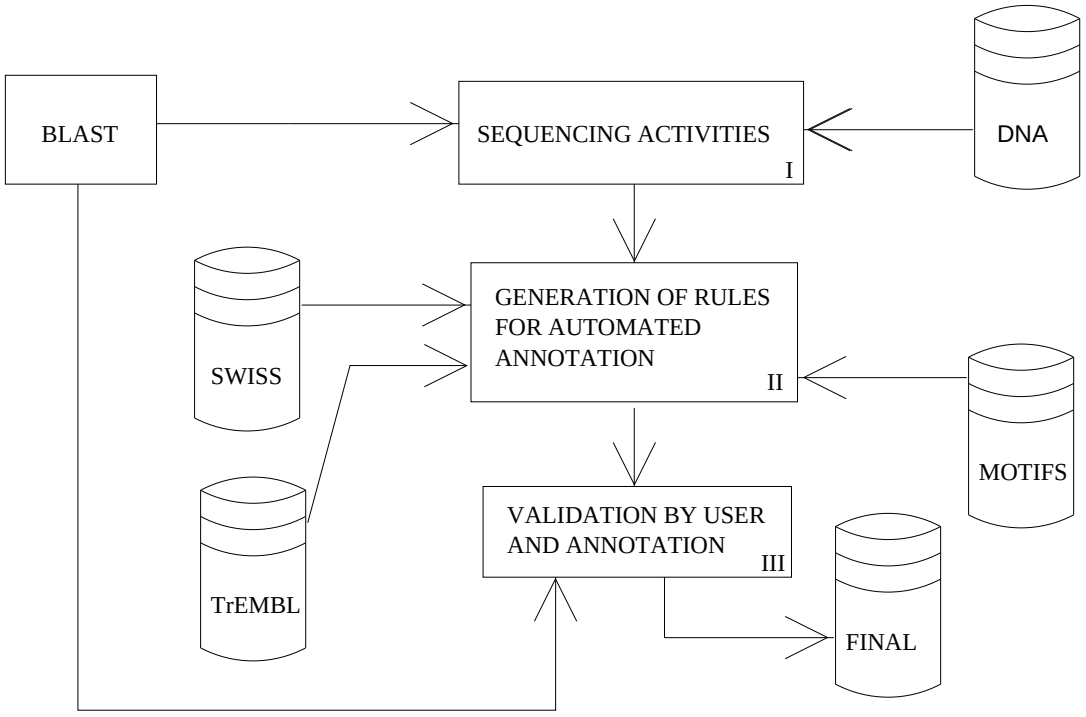

Figura 4.1: Arquitetura básica do sistema ATUCG.

nologias e métodos computacionais mais eficientes continuamente estão melhorando a capacidade de detecção de novas funcionalidades codificadas em seqüências genéticas.

4.2 Re-anotação e Busca por Similaridade

A principal técnica utilizada para re-anotação é a busca por similaridade. Como discutido no capítulo anterior, diversos projetos de re-anotação foram capazes de identificar alterações no resultado da anotação a partir dos novos resultados reportados por programas de busca, em especial o BLAST.

Logo, a re-anotação baseia-se principalmente na identificação de alterações em bancos de dados utilizados durante a anotação e se estas alterações podem causar um impacto nos resultados já obtidos. Se sim, a anotação deve ser alterada a fim de refletir as alterações encontradas.

Desse modo, um sistema de re-anotação automática deveria ter como base a capacidade de realizar comparações periódicas entre as informações provenientes da anotação e os dados armazenados nos bancos de dados atualizados. Essa é a tarefa realizada pelo modelo de re-anotação aqui proposto, onde as seqüências presentes no banco de dados do sistema de anotação são comparadas ao Genbank e COG a cada atualização destes últimos, em busca de novas informações que possam ser úteis ao usuário especialista.

4.3 Descrição Geral do Modelo

O módulo de re-anotação desenvolvido neste trabalho está acoplado ao sistema ATUCG (BAZZAN et al., 2003) (*Agent-based environmenT for aUtomatiC annotation of Genomes*). O ambiente ATUCG é um sistema integrado de anotação baseado na tecnologia de agentes. Um diagrama básico da arquitetura do sistema é mostrado na Figura 4.1.

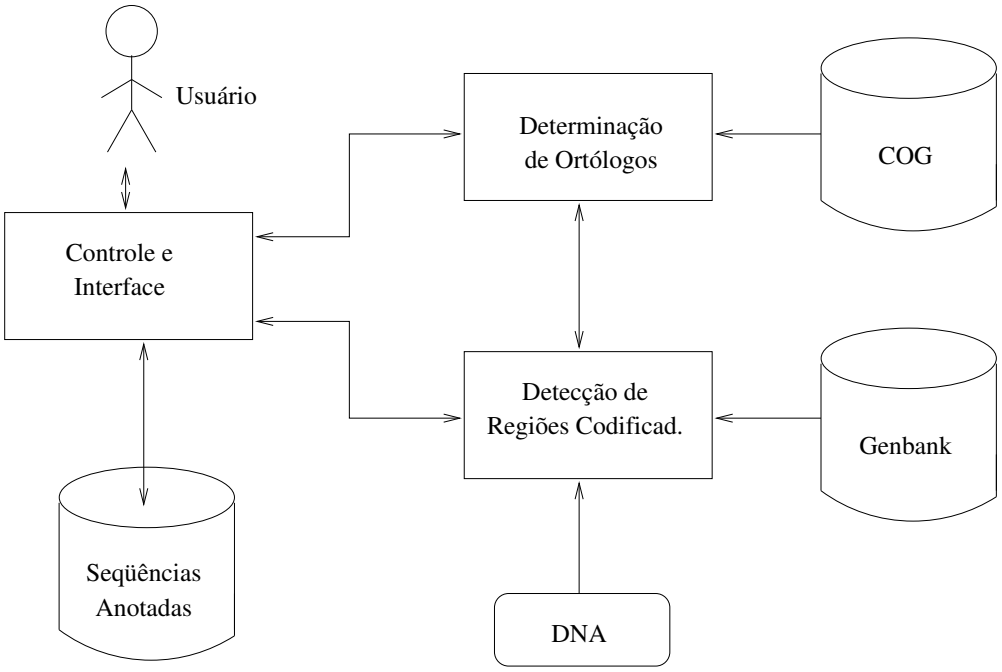

Figura 4.2: Oraganização geral do modelo de re-anotação.

A camada I é responsável pela determinação de ORFs e possíveis regiões codificantes. Agentes executando ferramentas diferentes de detecção destas regiões analisam a seqüência de DNA do organismo desejado e reportam possíveis seqüências encontradas. O resultado devolvido por esta camada é uma lista de ORFs não redundantes contendo possíveis regiões codificantes, a partir da verificação efetuada pelo usuário especialista sobre o resultado reportado pelos agentes.

Já a camada II é responsável pela anotação do campo *Keywords* do banco de dados Swiss-Prot para cada possível região codificadora encontrada na camada I. Esta anotação é realizada com base em regras de classificação geradas automaticamente a partir de dados extraídos do banco de dados Swiss-Prot. Finalmente, a camada III é responsável pela apresentação e validação da anotação gerada.

O módulo de re-anotação encontra-se inserido no contexto da camada I do ATUCG. Sua função é analisar periodicamente as regiões anotadas do genoma e realizar a re-análise destas informações , na tentativa de identificar novas anotações ou estender as já existentes.

O modelo implementado neste trabalho está organizado de forma distribuída, onde agentes desempenhando funções distintas interagem a fim de obter informações relacionadas ao genoma analisado. Basicamente, os agentes desempenham duas tarefas:

- *Análise de regiões anotadas*, onde as regiões já anotadas do genoma são re-analisadas, com o objetivo de procurar por novos dados relacionadas àquelas regiões;

- *Análise de regiões não anotadas*, onde ORFs não anotadas no genoma são analisadas a fim de que possíveis novas regiões codificantes possam ser identificadas.

A organização destes agentes pode ser visualizada na Figura 4.2. Cada um destes agentes será detalhado nas próximas seções.

Cada agente segue a estrutura mostrada na Figura 4.3. O módulo de *Conhecimento* contém as informações conhecidas pelo agente acerca das seqüências anotadas, que po-

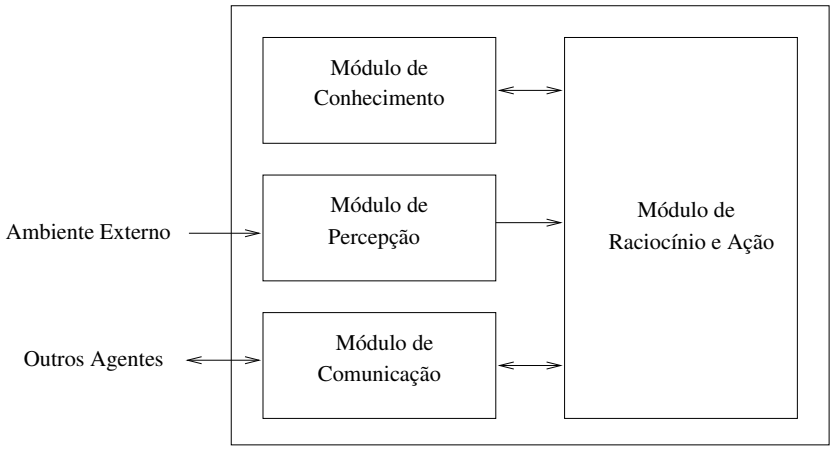

Figura 4.3: Estrutura dos agentes.

dem ser modificadas pela execução de ações. Estas ações são executadas pelo módulo de *Raciocínio e Ação*. Este módulo analisa as percepções do agente, assim como as mensagens recebidas de outros agentes, e executa a seqüência de ações apropriada. As ações executadas pelos agentes incluem buscas utilizando BLAST, análises através do Glimmer, etc. O módulo de *Percepção* contém sensores capazes de identificar determinadas informações do ambiente. Sensores utilizados pelo sistema incluem a verificação de modificões no Genbank e COG. Finalmente, o módulo de *Comunicação* processa as mensagens recebidas e enviadas pelo agente.

4.4 Detecção de Regiões Codificantes

Este agente é responsável pela identificação de possíveis novas regiões codificantes no genoma. O agente constantemente checa por novas atualizações na base de proteínas do Genbank. Caso novas seqüências tenham sido cadastradas e o banco de dados modificado, o agente inicia uma nova análise sobre as seqüências anotadas. Esta análise ocorre em duas etapas:

1. Primeiramente uma lista de possíveis genes é construída utilizando o programa Glimmer. Esta lista é comparada às seqüências codificantes já identificadas e, caso alguma possível nova seqüência codificadora tenha sido encontrada, esta seqüência é adicionada à lista reportada pelo agente.

2. A segunda etapa consiste em analisar regiões do genoma não anotadas e não identificadas pelo Glimmer utilizando uma busca BLAST no banco de dados de proteínas do Genbank. Similaridades com outras seqüências já anotadas presentes neste banco de dados podem indicar a presença de uma região codificante. ORFs identificadas no genoma, presentes entre *start codons* e *stop codons*, são utilizadas nesta análise e comparadas às seqüências no Genbank utilizando a variante BLASTP do BLAST. Convencionou-se como start codons as seqüências ATG, TTG e GTG e como stop codons TAG, TAA e TGA[1], como ilustrado na Figura 4.4.

[1]Este stop codon não foi utilizado nos testes com o genoma do *Mycoplasma pneumoniae*, pois é traduzido para o amino-ácido Triptofano(W) nestes organismos.

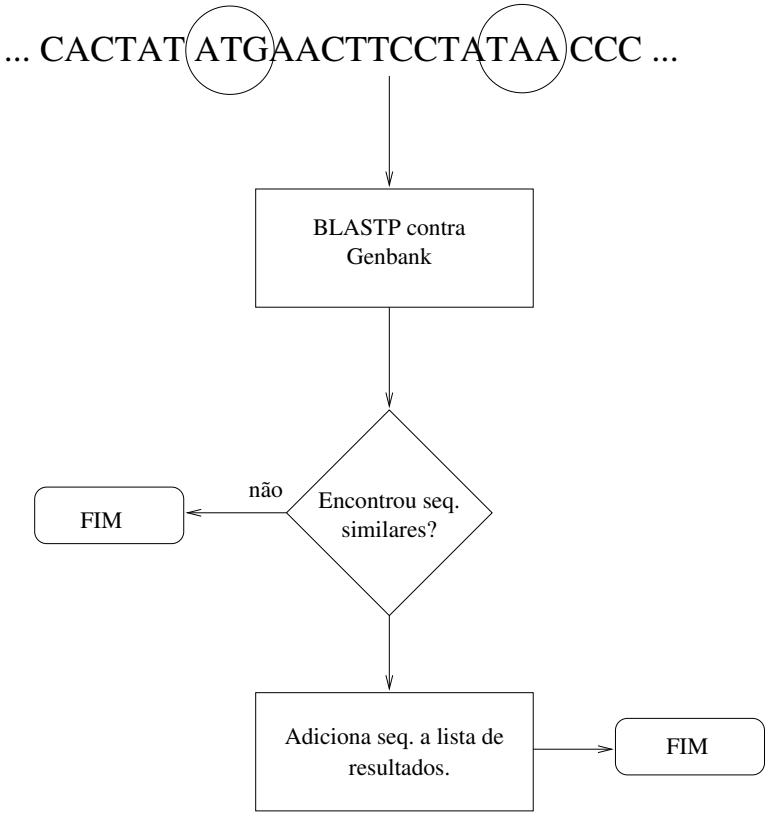

Figura 4.4: Detecção de prováveis regiões codificantes através de busca por similaridade.

Os parâmetros utilizados pelo agente incluem o tamanho mínimo (em número de bases) considerado para genes e o parâmetro *E* do BLAST, indicando um valor de significância mínimo para as seqüências a serem selecionadas como similares na busca. Ambos os parâmetros podem ser configurados para cada processo de re-anotação, de forma a tornar o modelo mais flexível.

Para a identificação de modificações nos dados do Genbank, o agente conta com um sensor especializado. Este sensor verifica periodicamente se existem atualizações no Genbank, a partir dos arquivos disponibilizados pelo NCBI[2]. Se o banco de dados foi alterado, o sensor dispara e o agente inicia um novo processo de re-anotação.

Ao final do processo, os resultados obtidos são unificados em uma listagem enviada ao agente de controle e interface. Novas ORFs identificadas são apresentadas também ao agente responsável pela determinação de ortólogos, para que a seqüência identificada possa ser caracterizada.

4.4.1 Utilizando o BLAST na Detecção de Regiões Codificantes

O BLAST é uma ferramenta de busca por similaridade bastante utilizada em projetos de bioinformática e análise de seqüências genéticas. Sua utilização como ferramenta de auxílio na detecção de regiões codificantes em genomas tem sido bastante difundida.

Os genes podem sofrer mutações e se diferenciar a partir de um mesmo ancestral comum. Este fato pode levar à ocorrência de genes que apresentam seqüências diferentes mas uma mesma função em organismos diferentes (ortólogos), quando ambos os organismo tenham evoluído a partir de um mesmo ancestral. Outro fato que ocorre principal-

[2] ftp://ftp.ncbi.nih.gov/blast/db/FASTA/nr.gz

mente em organismos procariontes é a transferência de material genético entre indivíduos, onde genes podem ser transferidos de um genoma para outro. Logo a presença de genes semelhantes em organismos próximos na cadeia evolucionária é bastante comum e um fator a ser explorado durante o processo de detecção de genes.

Muitos genes foram identificados a partir de similaridade, como podemos observar na anotação de diversos organismos, dentre eles o *Mycoplasma pneumoniae* (HIMMELREICH et al., 1996) e de muitas seqüências presentes no Genbank. Apesar disso, a existência de similaridade de uma região do genoma não deve ser considerada como um indício exato da existência de um gene, já que a existência de falsos positivos pode ocorrer. O BLAST é basicamente uma ferramenta heurística e seus resultados podem ser utilizados como um bom ponto de partida para a verificação se uma dada região do genoma analisado realmente é um gene.

No modelo proposto, utiliza-se o BLAST basicamente como uma ferramenta para indicar possíveis novos genes presentes em regiões ignoradas durante o processo de anotação. Logo, o sistema desenvolvido a partir do modelo aqui descrito não visa indicar a presença de genes com absoluta certeza, mas apenas servir como uma ferramenta de alerta para possíveis caminhos de investigação por parte de especialistas humanos.

4.5 Detecção de Ortólogos

O agente de detecção de ortólogos utiliza uma busca por similaridade nas seqüências armazenadas no banco de dados COG para classificar uma data região do genoma em um grupo de genes ortólogos. A busca por similaridade é realizada utilizando a variante BLASTP do BLAST.

A partir das seqüências identificadas na busca pelo BLAST, pode-se determinar quais os possíveis grupos de ortólogos aos quais a região analisada está associada. Os grupos encontrados são então enviados ao agente de Controle e Interface. O processo é ilustrado na Figura 4.5.

O COG basicamente contém duas seções. Uma que lista todos os genes catalogados no banco de dados. Esta listagem é bastante útil quando se deseja encontrar genes similares presentes no COG. Pode-se utilizar o BLAST como ferramenta de busca nesta tarefa.

A outra seção contém a categorização dos genes em grupos de ortólogos. Isso significa que genes de diferentes organismos que apresentam funções e seqüências similares são agrupados em um mesmo grupo. Logo, os resultados obtidos pelo BLAST sobre a listagem de genes catalogados podem ser utilizados como ponto de partida na pesquisa na associação de um determinado grupo de ortólogos a um gene desconhecido. A existência de similaridades com diversos genes de um mesmo grupo pode indicar uma grande probabilidade de que um gene pertença a ele.

Para a identificação de modificações nos dados do COG, o agente conta com um sensor especializado. Este sensor verifica periodicamente se existem atualizações no COG, a partir da Internet. Se o banco de dados foi alterado, o sensor dispara, o agente faz o download do banco de dados para uma cópia local e inicia um novo processo de re-anotação.

No modelo aqui proposto, similaridades com genes catalogados são identificadas mas não analisadas profundamente de forma a eliminar falsos positivos. Logo, o usuário especialista deve realizar uma análise sobre os resultados obtidos a fim de confirmar ou não a re-anotação proposta pelo sistema.

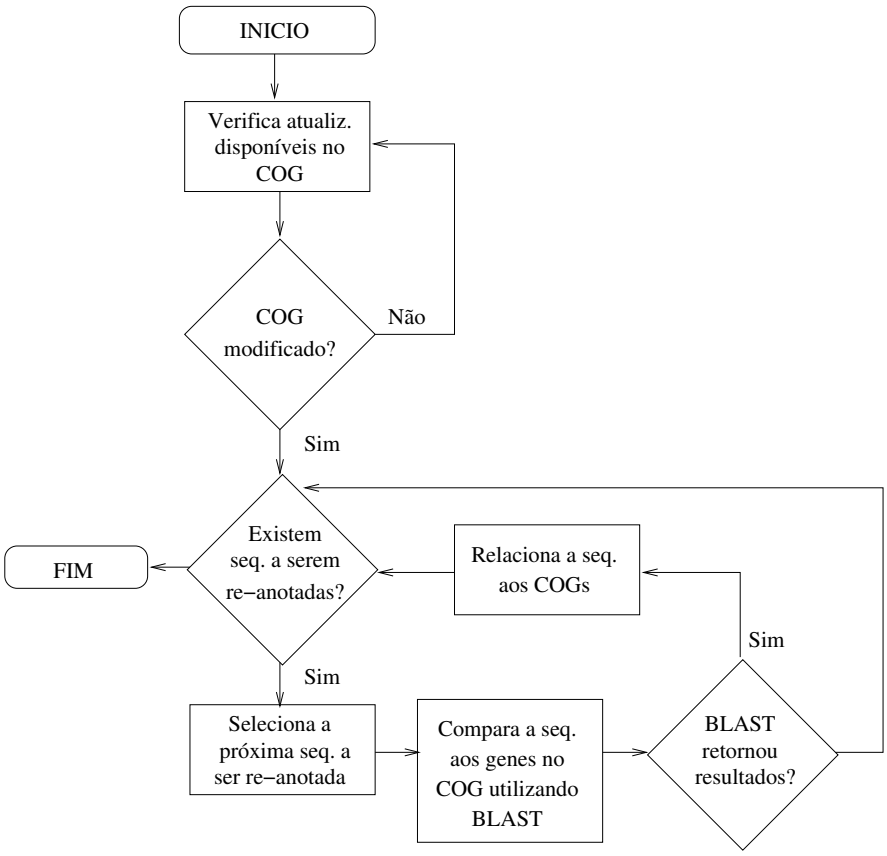

Figura 4.5: Associação de COGs a seqüências no modelo de re-anotação.

4.6 Controle e Visualização do Processo de Re-anotação

O agente de *Controle e Interface* é responsável por receber as requisições provenientes da interface com o usuário e integrar as informações provenientes dos outros agentes do sistema. A interface com o usuário consiste em um relatório sobre os resultados de processos de re-anotação executados pelo sistema sobre os genomas cadastrados. Este relatório pode ser exibido tanto por uma página HTML acessível através da Internet, bem como um e-mail enviado ao usuário.

A integração dos dados é uma tarefa simples. Quando o agente de controle e interface recebe mensagens dos outros agentes do sistema indicando as novas anotações encontradas, estas são integradas em um único registro de re-anotação. Este registro é criado a fim de que o usuário possa acompanhar claramente que informações foram re-anotadas e quando isto ocorreu.

Os registros de re-anotações contêm basicamente três tipos de informações:

- A identificação do gene ou região codificadora re-anotada;
- Uma descrição textual sobre o que foi re-anotado;
- As causas identificadas pelo sistema que levaram à re-anotação sugerida.

A partir dos resultados reportados pelo sistema, o usuário especialista pode aceitar ou não as re-anotações propostas. Falsos positivos podem estar presentes e a intervenção do usuário é imprescindível na etapa de validação destas informações.

4.7 Limitações do Modelo

Como observado em seções anteriores, o modelo aqui proposto contém diversas limitações. Primeiramente, os resultados reportados podem conter falsos positivos, e não devem ser considerados como o resultado final do processo de análise. As ferramentas utilizadas como fonte de informações, o BLAST e o Glimmer, contém limitações e podem fornecer falsos positivos que levam a conclusões errôneas por parte do sistema. A inclusão do conhecimento de especialistas na fase de integração dos resultados é uma proposta interessante, mas mesmo nestes casos o sistema estaria longe de substituir um especialista humano na exatidão dos resultados.

Outra limitação que deve ser mencionada é a ausência de outros tipos de análise, como busca por motivos por exemplo. Este tipo de análise também sofre influência da atualização periódica de bancos de dados e logo seria bastante interessante sua inclusão em versões futuras do sistema.

O modelo foi projetado e validado utilizando genomas de organismos procariontes. Sua utilização na análise de genomas eucariontes exigiria a inclusão de outros tipos de ferramentas e alguns procedimentos devem ser modificados, principalmente na detecção de regiões cofidicadoras. Logo, a utilização do sistema proposto neste trabalho na análise de genomas eucariontes não é recomendada.

5 IMPLEMENTAÇÃO DO MODELO E VALIDAÇÃO

Neste capítulo a implementação do modelo e sua validação utilizando genomas já anotados será detalhada. Primeiramente a Seção 5.1 apresenta a descrição de como o modelo descrito no capítulo anterior foi implementado. A seguir, na Seção 5.2 serão apresentados alguns resultados obtidos com esta implementação na análise de genomas procariontes já anotados durante a validação do sistema.

5.1 Implementação do Modelo

A implementação do modelo de re-anotação aqui proposto foi realizada em ambiente Linux, utilizando as linguagens Java e Perl. A opção pelo sistema Linux se deve a sua ampla utilização em bioinformática e crescimento nos últimos anos, bem como proporcionar um ambiente estável e eficiente para os testes.

A linguagem Java foi utilizada basicamente na implementação dos agentes. A opção por esta linguagem se deveu principalmente à fácil integração do sistema ao *framework* multiagente utilizado no projeto.

A linguagem Perl foi utilizada somente na integração dos agentes com as ferramentas de bioinformática utilizadas. Esta linguagem é largamente utilizada na implementação de programas de análise em bioinformática devido principalmente a sua excelente biblioteca para processamento de *strings*.

5.1.1 FIPA-OS

A implementação dos agentes utilizados no modelo foi realizada em Java, através do framework multiagente FIPA-OS (POSLAD; BUCKLE; HADINGHAM, 2000). A opção pelo FIPA-OS se deveu principalmente por sua facilidade de uso e por ser bastante estável.

O FIPA-OS (*FIPA Open Source*) é uma plataforma multiagente de código fonte aberto desenvolvida originalmente pela Nortel Networks. A plataforma suporta a interação de diversos agentes utilizando um modelo multiagente que segue as especificações da FIPA (*Foundation for Intelligent Physical Agents*) [1]. O FIPA-OS está sendo utilizado por diversas instituições no mundo todo, empregado em diferentes domínios de aplicação, como controle de VPNs (*Virtual Private Networks*), *Distributed Meeting Schedule* entre outros. Além disso, o FIPA-OS pode ser integrado a outras plataformas multiagente que seguem as especificações da FIPA.

[1] http://www.fipa.org

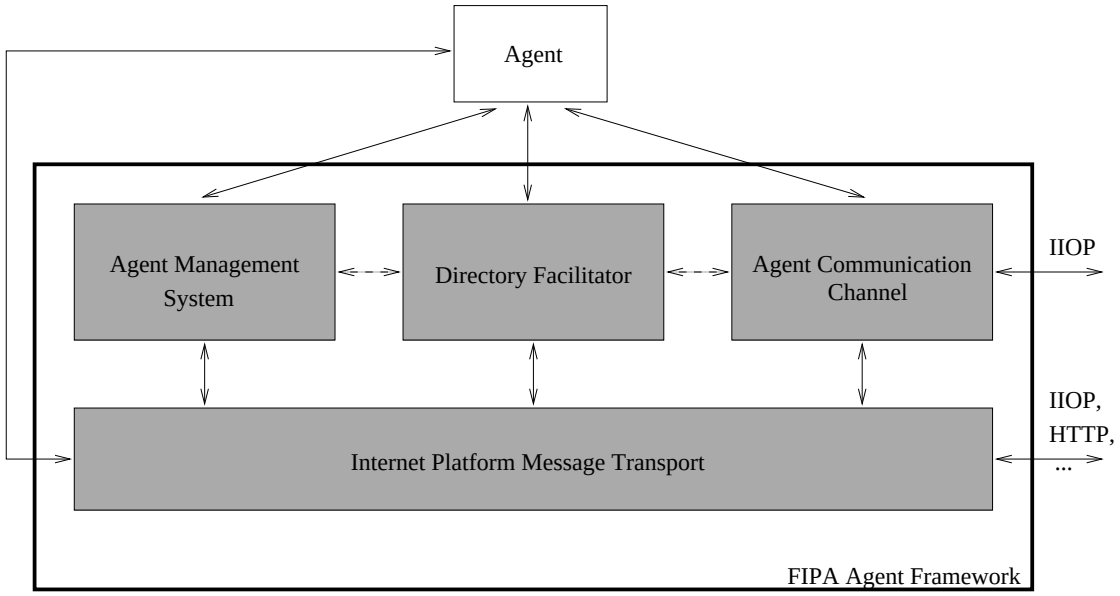

Figura 5.1: Modelo de referência FIPA.

5.1.1.1 O Modelo FIPA

A FIPA (O'BRIEN; NICOL, 1998) é uma organização sem fins lucrativos cujo objetivo é promover o desenvolvimento de especificações relacionadas a sistemas multiagentes que maximizem a interoperabilidade entre aplicações baseadas em agentes. Desenvolvedores são livres para implementar qualquer especificação FIPA, desde que os respectivos sistemas construídos estejam de acordo com o que foi definido.

O modelo de referência FIPA para a implementação de frameworks multiagente é ilustrado na Figura 5.1. Este modelo atualmente fornece a base sobre a qual os agentes FIPA existem e operam. Combinado ao ciclo de vida do agente, estabelece o contexto lógico e temporal para a criação, operação e destruição dos agentes.

Os componentes DF (*Directory Facilitator*), AMS (*Agent Management System*), ACC (*Agent Communication Channel*) e IPMT (*Internal Platform Message Transport*) formam a chamada *Agent Platform* (AP). O componente DF fornece um serviço de "páginas amarelas" aos outros agentes. O componente AMS fornece um serviço de "páginas brancas" e gerenciamento dos agentes. Já o componente ACC suporta a intercomunicação entre agentes dentro e entre plataformas. O IPMT fornece um serviço de transporte de mensagens interno a cada plataforma. Os componentes AMS, DF e ACC podem ser implementados em um único agente ou em três agentes diferentes.

A comunicação entre agentes é realizada através de mensagens codificadas na linguagem FIPA ACL (*Agent Communication Language*). Esta linguagem define um modo padronizado de enpacotar mensagens, de modo a deixar claro para todos os agentes participantes qual o propósito da conversação sendo desenvolvida. A linguagem ACL utiliza um subconjunto de cerca de 20 verbos da língua inglesa como performativas de comunicação. Este método fornece uma maneira flexível para a comunicação entre entidades de software incluindo diversos benefícios como:

- Inclusão e remoção dinâmica de serviços;
- Serviços de qualquer natureza podem ser incluídos no sistema sem a necessidade de re-compilação dos clientes em tempo de execução;

- O processamento decentralizado de informações é facilitado;
- Uma linguagem para troca de mensagens universal fornecendo uma interface baseado em atos de fala consistente através de software;
- Interação assíncrona baseada em troca de mensagens entre entidades.

5.1.1.2 O Framework FIPA-OS

O framework FIPA-OS foi desenvolvido de acordo com a especificação estabelecida pela FIPA. O modelo descrito anteriormente contém o núcleo da plataforma FIPA-OS: o *Directory Facilitator* (DF), o *Agent Management System* (AMS), o *Agent Communication Channel* (ACC) e o *Internal Platform Message Transport* (IPMT).

Além dos componentes definidos pelo modelo de referência FIPA, o FIPA-OS provê suporte para:

- Diferente tipos de *Agent Shells* utilizados no desenvolvimento de agentes que podem interagir através dos mecanismos de comunicação da plataforma FIPA-OS;
- Comunicação multi-camadas entre agentes;
- Gerenciamento de mensagens e conversações;
- Configuração dinâmica de plataformas para suporte de múltiplos IPMTs;
- Ferramentas de diagnóstico e visualização.

Os agentes são definidos através da estensão de classes Java fornecidas pelo framework. A definição de um agente no FIPA-OS é realizada através da criação de uma classe que extenda a classe `FIPAOSAgent`. Esta classe define a estrutura básica de um agente no framework.

5.1.2 Ferramentas de Bioinformática Utilizadas

Para as análises efetuadas sobre as seqüências genéticas submetidas ao sistema de reanotação foram utilizadas duas ferramentas de bioinformática: o programa de busca por similaridade BLAST e o programa de detecção de genes Glimmer. A versão do BLAST utilizada foi a disponibilizada gratuitamente pelo NCBI. O pacote fornecido pelo NCBI contém o programa BLAST propriamente dito e diversas outras ferramentas relacionadas.

Já a versão utilizada do Glimmer foi a versão 2.0, desenvolvida pelo TIGR (*The Institute for Genomic Research*) [2]. Juntamente com o programa Glimmer são fornecidos diversos programas, incluindo softwares para geração de modelos IMM a partir de um conjunto de seqüências genéticas e softwares para visualização dos resultados.

5.1.3 Implementação dos Agentes

A implementação dos agentes seguiu o diagrama de classes mostrado na Figura 5.2. Na figura, uma única classe é utilizada para a implementação dos agentes no sistema. Esta classe, denominada `Agent`, contém a estrutura comum a todos os agentes do sistema. Esta classe basicamente realiza o registro do agente com os agentes AMS e DF do FIPA-OS. O registro com o agente AMS é necessário para que o agente possa ser identificado,

[2] http://www.tigr.org/software/glimmer

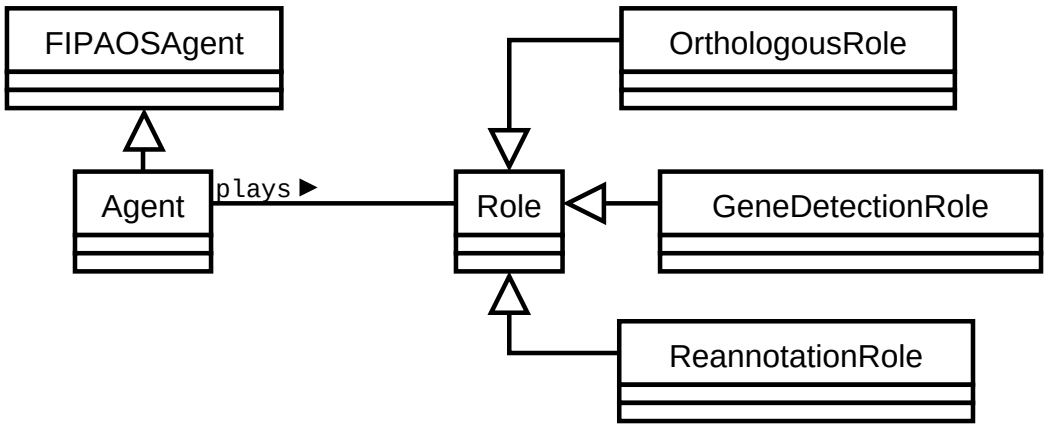

Figura 5.2: Diagrama de classes simplificado do sistema.

dentro do sistema, através do serviço de “páginas brancas”. Já o registro realizado junto ao agente DF é necessário para a identificação dos serviços fornecidos pelo agente. Este registro no DF é bastante útil para que buscas por agentes relacionados a determinados serviços possa ser realizada.

As particularidades de cada agente do modelo foram implementadas como papéis, representados pela classe abstrata `Role`. A classe `Role` é especializada em três classes, uma para cada tipo de agente. Dessa forma para a inclusão de um novo tipo de agente basta que uma nova classe estendendo a classe `Role` seja criada. As classes que representam papéis específicos devem definir as ações que os agentes podem executar e os sensores utilizados pelos agentes.

Os papéis definidos no sistema são os seguintes:

- **OrthologousRole**: Contém a definição do papel de análise de ortologia. Este papel acrescenta às capacidades do agente as ações de busca no COG, através da execução do programa BLAST, e de determinação dos grupos de ortólogos relacionados às seqüências analisadas. Além disso, um sensor para detecção de COGs é também utilizado para a verificação de alterações neste banco de dados. A estrutura completa de classes do agente é mostrada na Figura 5.3.

- **GeneDetectionRole**: Este papel representa a tarefa de detecção de possíveis genes. Como ilustrado na Figura 5.4, são acrescentados ao agente uma ação de busca de seqüências homólogas através do BLAST, uma ação para execução do Glimmer e finalmente uma ação para determinação dos possíveis novos genes detectados. Um sensor para verificação de atualizações no Genbank é utilizado para disparar a re-anotação automaticamente a cada modificação.

- **ReannotationRole**: Este é o papel executado pelo agente de Controle e Interface. Basicamente define rotinas para interface com o usuário e recebe as informações relacionadas aos processos de re-anotação dos outros agentes. A tarefa de integrar as informações de re-anotação é executada pela ação `ReannotationAction`, mostrada na Figura 5.5.

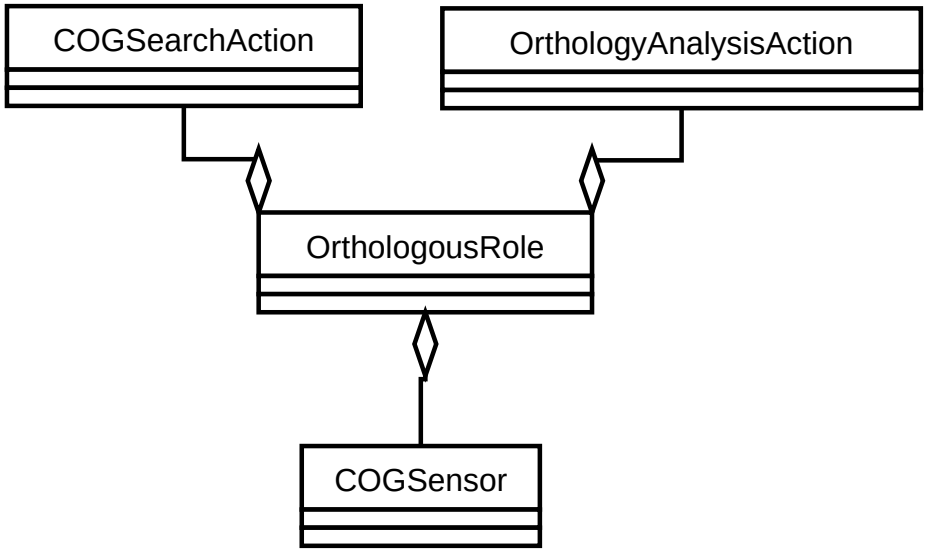

Figura 5.3: Diagrama de classes simplificado do papel de detecção de ortólogos.

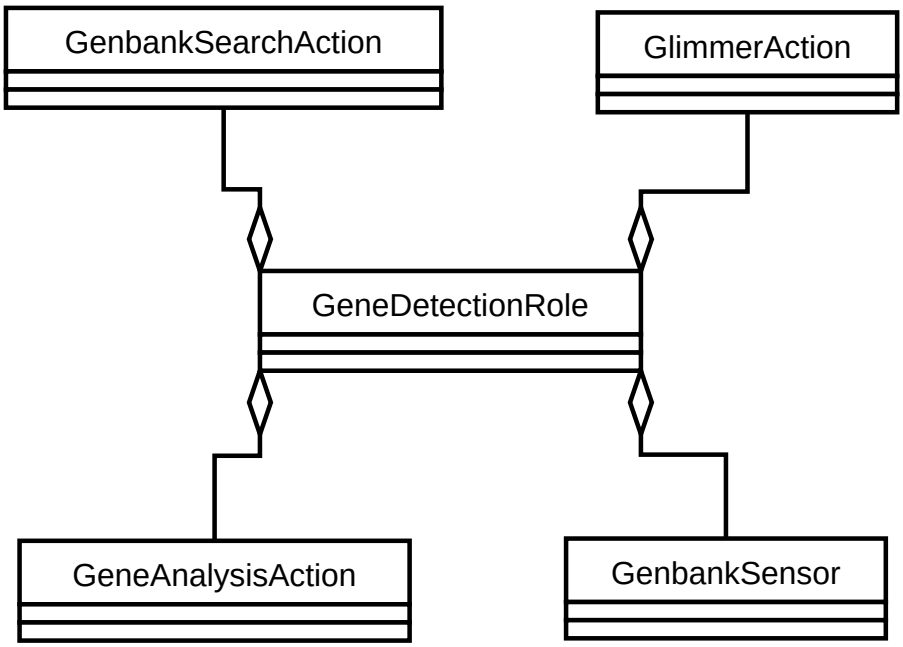

Figura 5.4: Diagrama de classes simplificado do papel de detecção de possíveis genes.

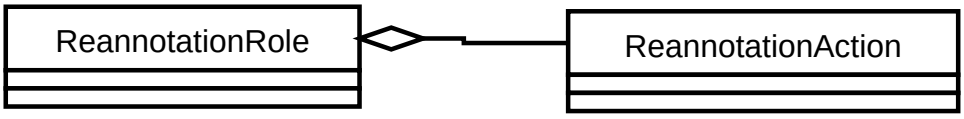

Figura 5.5: Diagrama de classes simplificado do papel de re-anotação.

5.2 Validação

A validação do modelo foi realizada utilizando genomas de dois organismos procariontes: *Mycoplasma pneumoniae* e *Haemophilus influenzae*. O *Mycoplasma pneumoniae* é uma bactéria responsável por alguns tipos de pneumonia em seres humanos. Contém um dos menores genomas já identificados e foi recentemente re-anotado (DANDEKAR et al., 2000). O *Haemophilus influenzae* é uma bactéria associada a alguns tipos de penumoniae que, apesar do nome, não se trata do vírus da influenza responsável pelo resfriado. Como o *M. pneumoniae* também foi recentemente re-anotado, em uma comparação efetuada com o genoma de outra bactéria, o *Escherichia coli* (TATUSOV et al., 1996).

Os testes foram realizados em ambiente Linux, rodando em máquinas PC convencionais. Os genomas utilizados foram obtidos a partir do repositório do NCBI. Os parâmetros utilizados foram os seguintes:

- **BLAST**: Utilizou-se um threshold para o valor E de 10^{-6}.
- **Genes**: O tamanho mínimo considerado para regiões codificantes foi de 100 nucleotídeos.

Os resultados do BLAST foram filtrados a fim de remover hits referentes a seqüências dos organismos analisados. Assim, foram consideradas somente similaridades com seqüências provenientes de outros organismos. Este passo foi necessário, pois muitas das seqüências que deveriam ser re-anotadas nos testes efetuados já estão cadastradas no Genbank e COG.

5.2.1 Análise do Genoma do *Mycoplasma pneumoniae*

O genoma do *M. pneumoniae* foi analisado preliminarmente utilizando somente o BLAST para a detecção de regiões codificantes. Os resultados obtidos foram publicados em (NASCIMENTO; BAZZAN, 2004) e uma parte das regiões detectadas é mostrada na Tabela 5.1. A primeira coluna mostra as regiões identificadas como codificantes a partir da comparação com o Genbank utilizando o programa BLAST. Os intervalos especificados nesta coluna representam o local onde ocorreram hits reportados pelo BLAST na seqüência de DNA do genoma. A segunda coluna mostra as correspondentes regiões codificantes presentes no genoma do *M. pneumoniae* segundo o registro no Genbank (GI: 13507739). Estas seqüências estão identificadas pelo número GI, um índice numérico único utilizado no banco de dados Genbank. Como pode ser observado, as regiões codificadoras foram identificadas, apesar do erro quanto a precisão de seus limites.

O teste mostrado foi realizado sobre o genoma originalmente anotado do *M. pneumoniae* (HIMMELREICH et al., 1996). Este teste basicamente realizou uma busca utilizando o BLAST com as seqüências pertencentes a regiões codificantes e regiões entre estas regiões, originalmente tidas como não-codificantes. Utilizou-se para o parâmetro E o valor 10^{-6} realizando uma busca com a variante blastx, comparando as seqüências de nucleotídeos extraídas do genoma com o banco de proteínas não-redundante do Genbank. O Glimmer não foi utilizado nestes testes.

O sistema neste caso comparou seqüências pertencentes a regiões tidas como não-codificantes com o banco de dados, a fim de identificar novas regiões codificadoras. Regiões anotadas como codificantes também foram comparadas a fim de confirmar sua anotação.

Um novo teste utilizando o Glimmer e o BLAST em conjunto sobre um subconjunto da última versão re-anotada do genoma do *M. pneumoniae* foi realizado. Removeu-se

Tabela 5.1: Resultados da análise do *M. pneumoniae* utilizando BLAST.

Regiões Identificadas	**Regiões Codificantes Presentes no Genbank**
759..1832	ORF: 682..1834 GI: 13507740
1839..2765	ORF: 1838..2767 GI: 13507741
2870..3418 3422..3595 3596..4780	ORF: 2869..4821 GI: 13507742
4822..5979 5980..6261 6262..6663 6664..7146	ORF: 4821..7340 GI: 13507743
7313..7819 7649..8560	ORF: 7312..8574 GI: 13507744
...	

aleatoriamente algumas regiões anotadas como codificantes na versão atual presente no Genbank. Utilizou-se uma subseqüência de DNA do genoma presente entre as bases 1 e 40000. O conjunto das regiões codificantes presente neste intervalo é mostrado na Tabela 5.2. Esta tabela mostra a posição das regiões codificantes (coluna "Posição") e se esta região estava presente ou não nos testes realizados (coluna "Status no Teste").

O resultado esperado nos testes realizados era a identificação das regiões removidas como novas regiões codificantes. A re-anotação do subconjunto apresentado do genoma do *M. pneumoniae* utilizando o sistema é mostrado na Tabela 5.3. A primeira coluna contém a posição das regiões identificadas como codificantes no genoma. A segunda contém a re-anotação obtida a partir da análise utilizando o banco de dados COG (ver Seção 4.5). Como pode ser observado, as regiões identificadas correspondem a subregiões das verdadeiras regiões codificantes encontradas, apesar dos limites exatos não estarem corretos em alguns casos. Isto ocorreu principalmente porque o modelo utilizado pelo Glimmer não prevê o conjunto anormal de stop codons utilizado pelo *M. pneumoniae*. O códon "TGA" é traduzido como o amino-ácido triptofano nas células destes organismos e não é identificado como um stop codon. Logo, o Glimmer pode cometer erros quanto aos limites de regiões codificadoras estabelecidos por stop codons nestes genomas, por não considerar esta particularidade.

5.2.2 Análise do Genoma do *Haemophilus influenzae*

O teste efetuado no genoma do *H. influenzae* utilizou um processo similar ao utilizado nos testes com o *M. pneumoniae*, removendo-se aleatoriamente regiões consideradas codificantes do registro original para que estas fossem re-anotadas pelo sistema. O conjunto das regiões codificantes presentes no intervalo 1..20000 utilizado no teste é mostrado na Tabela 5.4. Esta tabela mostra a posição das regiões codificantes (coluna "Posição") e se esta região estava presente ou não nos testes realizados (coluna "Status no Teste").

Os testes realizados, utilizando o mesmo conjunto de parâmetros do realizado com o genoma do *M. pneumoniae* obteve os resultados mostrados na Tabela 5.5. Esta tabela mostra que foram identificadas regiões codificadoras nas regiões removidas no teste,

Tabela 5.2: Subconjunto do genoma do M. pneumoniae analisado.

Posição	**Status no Teste**
692..1834	Presente
1838..2767	Presente
2869..4821	Presente
4821..7340	Presente
7312..8574	Presente
8579..9211	**Ausente**
9184..9945	**Ausente**
9947..11275	**Ausente**
11275..12060	Presente
12257..12652	Presente
12838..13533 (complemento)	Presente
13558..14265 (complemento)	Presente
14992..15765 (complemento)	Presente
15867..16505 (complemento)	**Ausente**
16482..17339 (complemento)	**Ausente**
17339..18205 (complemento)	**Ausente**
18180..18989 (complemento)	Presente
19325..21196	Presente
21108..23012	Presente
23022..26114	Presente
26160..27332	Presente
27316..28245	Presente
28245..29783	Presente
29804..30244	Presente
30244..31110	Presente
31111..32199	Presente
32202..33026 (complemento)	**Ausente**
33059..33958	Presente
33979..34551	Presente
34469..34975	Presente
34975..35586	Presente
35810..36136 (complemento)	**Ausente**
36140..36760 (complemento)	**Ausente**

Tabela 5.3: Resultado da re-anotação do *M. pneumoniae*.

Posição	**COG**
8549..8875	COG0125 Thymidylate kinase
9184..9945	COG0470 ATPase involved in DNA replication
9923..11275	COG0486 Predicted GTPase
15867..16505 (complemento)	COG0358 DNA primase (bacterial type)
16713..17342 (complemento)	COG0189 Glutathione synthase/Ribosomal protein S6 modification enzyme (glutaminyl transferase)
17342..17830 (complemento)	COG0189 Glutathione synthase/Ribosomal protein S6 modification enzyme (glutaminyl transferase)
17843..18208 (complemento)	COG0189 Glutathione synthase/Ribosomal protein S6 modification enzyme (glutaminyl transferase)
32205..32309 (complemento)	Nenhum
32619..33029 (complemento)	Nenhum
35579..35677 (complemento)	Nenhum
35813..36139 (complemento)	COG0693 Putative intracellular protease/amidase
36143..36763 (complemento)	COG0035 Uracil phosphoribosyltransferase

Tabela 5.4: Subconjunto do genoma do H. influenzae analisado.

Posição	**Status no Teste**
2..1021	Presente
1190..3013	Presente
3050..3838 (complemento)	Presente
3854..4318 (complemento)	Presente
4579..5391 (complemento)	Presente
5662..8748	Ausente
8750..9688	Presente
9681..10397	Presente
10467..11375	Presente
11414..11854 (complemento)	Presente
11857..12261 (complemento)	Presente
12367..13359	Presente
13423..14331 (complemento)	Presente
14328..15011 (complemento)	Presente
15013..16062 (complemento)	**Ausente**
16071..17867 (complemento)	**Ausente**
18035..18418 (complemento)	Presente
18676..19335	Presente

Tabela 5.5: Resultado da re-anotação do *H. influenzae*.

Posição	**COG**
5662..8748	COG0243 Anaerobic dehydrogenases, typically selenocysteine-containing
6322..8748	COG0243 Anaerobic dehydrogenases, typically selenocysteine-containing
15013..16062 (complemento)	COG0681 Signal peptidase I
16071..17867 (complemento)	COG0481 Membrane GTPase LepA
17902..18030 (complemento)	Nenhum

como esperado. A tabela mostra também uma região extra identificada pelo sistema (17902..18030). Casos como este devem ser validados por um especialista humano, pois podem significar falsos positivos. Logo, é necessário que o sistema seja capaz de obter informações adicionais sobre a seqüência de forma a auxiliar o especialista na validação dos dados. Até o momento, o sistema é capaz de realizar a análise de COGs, mas outras funcionalidades podem ser adicionadas no futuro.

6 CONCLUSÃO

O estudo da estrutura e funcionalidade do aparato genético presente nos cromossomos pode fornecer informações preciosas sobre diversos organismos vivos, inclusive sobre os próprios seres humanos. Devido a este fato, muitos esforços têm sido efetuados com o objetivo de mapear diversos genomas através da anotação genética.

Este trabalho apresentou um novo modelo para re-anotação automática de genomas. A re-anotação é basicamente uma tarefa de re-análise de dados, onde as informações obtidas na anotação podem ser reconsideradas em virtude de um novo contexto. Isto ocorre principamente através da existência de novos dados cadastrados em bancos de dados especializados, publicamente disponíveis na Internet. Como um exemplo destes bancos de dados, podemos citar o Genbank.

Os computadores desempenham um papel importante na análise de dados genéticos. Diversas ferramentas computacionais para análise de características específicas em seqüências já foram desenvolvidas e são amplamente utilizadas. Como exemplos, podemos citar o BLAST, utilizado para buscas em bancos de dados através da similaridade entre seqüências, e o Glimmer, utilizado para a detecção de possíveis genes em genomas procariontes.

Um grande problema durante a análise em larga escala de genomas é a integração das informações obtidas a partir das diversas ferramentas de bioinformática utilizadas. Existem diversos sistemas integrados de anotação já desenvolvidos para tal finalidade (por exemplo, o SABIA e o GeneWeaver), utilizando as mais variadas tecnologias.

Nenhum sistema integrado de re-anotação foi desenvolvido até o momento, apesar de diversos projetos de análise envolvendo a revisão de dados de anotações genéticas já terem sido desenvolvidos. Assim, o desenvolvimento de tal sistema é um campo de estudo em aberto. Esta foi a principal motivação para este trabalho.

O sistema aqui apresentado tem o objetivo de ser um complemento a um sistema de anotação já existente. A implementação atual do módulo foi projetada para sua integração ao sistema ATUCG. Este sistema de anotação utiliza uma arquitetura baseada em agentes divida em três camadas. A camada I possui o objetivo de detectar as possíveis regiões codificantes presentes no genoma de um organismo, a partir de sua seqüência de DNA. A camada II recebe a listagem de seqüências identificadas na camada I e procura identificar o conteúdo do campo *Keywords* (definido no banco de dados Swiss-Prot) para cada seqüência. A camada III é uma interface para validação dos resultados por parte do usuário.

O módulo de re-anotação encontra-se inserido na camada I do ATUCG. O sistema contém agentes capazes de analisar características específicas das seqüências presentes no genoma do organismo e interagem a fim de obter o resultado final da re-anotação. O resultado final do processo é uma listagem de possíveis novas características identificadas

na seqüência a partir da análise efetuada.

A validação do sistema utilizando genomas dos organismos *Mycoplasma pneumoniae* e *Haemophilus influenzae* mostrou que o sistema pode ser uma ferramenta útil na detecção de possíveis regiões a serem analisadas. Apesar disso, o modelo pode acrescentar diversas outras funcionalidades a fim de se tornar uma ferramenta mais simples e prática, principalmente quanto a interface de utilização e na integração dos dados.

Como trabalhos futuros pode-se identificar os seguintes pontos:

- É possível adicionar outros agentes ao sistema, de modo que outras características possam ser analisadas. Como exemplo, pode-se estudar a inclusão de um agente que analise novos motivos presentes na seqüência, através de buscas no banco de dados Interpro.

- Pode-se estudar uma forma de incluir uma parte do conhecimento de especialistas na integração das informações provenientes dos agentes. Este conhecimento poderia estar codificado em regras e tais regras seriam aplicadas pelo agente de Controle e Interface no momento da análise das características re-anotadas.

- Um método de identificação de genes através da análise simultânea do genoma por diversas ferramentas de detecção de genes. Tal método utilizaria duas ou mais ferramentas de forma paralela e integraria seus resultados em uma única listagem não redundante de prováveis seqüências codificantes. A implementação poderia utilizar diversos agentes, cada um implementando uma ferramenta de análise, de forma que estes poderiam negociar a fim de obter o resultado final desejado. Este método foi estudado durante o decorrer deste trabalho, mas nenhum resultado conclusivo foi obtido.

Como conclusão final, este trabalho mostra-se apenas o início de um estudo mais profundo sobre como um sistema de re-anotação integrado automático pode ser construído de forma eficiente. Apesar disso, os resultados obtidos mostram que um módulo de re-anotação é uma ferramenta bastante interessante quando utilizada em conjunto a um sistema de anotação já existente. Novas informações sobre seqüências identificadas estão constantemente sendo disponibilizadas na Internet e tal sistema possibilita que o especialista esteja atualizado sobre possíveis modificações de forma automática.

REFERÊNCIAS

ALMEIDA, L. et al. A new set of bioinformatics tools for genome projects. **Genetics and Molecular Research**, [S.l.], v.3, n.1, p.26–52, 2004.

ALTSCHUL, S. F. et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. **Nucleic Acids Research**, [S.l.], v.25, p.3359–3402, 1997.

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

ANDRADE, M. A.; BROWN, N. P.; LEROY, C.; HOERSCH, S.; DARUVAR, A. de; REICH, C.; FRANCHINI, A.; TAMAMES, J.; VALENCIA, A.; OUZOUNIS, C.; SANDER, C. Automated genome sequence analysis and annotation. **Bioinformatics**, [S.l.], v.15, n.5, p.391–412, 1999.

APWEILER, R.; BAIROCH, A.; WU, C. Protein sequence databases. **Current Opinion in Chemical Biology**, [S.l.], v.8, p.76–80, 2004.

BARKER, W.; GARAVELLIAND, J.; MCGARVEY, P.; MARZEC, C.; ORCUTT, B.; SRINIVASARO, G.; YEH, L.; LEDLEY, R.; MEWES, H.; PFEIFFER, F.; TSUGITA, A.; WU, C. The PIR international protein sequence database. **Nucleic Acids Research**, [S.l.], v.27, n.1, p.39–43, 1999.

BATEMAN, A.; BIRNEY, E.; CERRUTI, L.; DURBIN, R.; ETWILLER, L.; EDDY, S.; GRIFFITHS-JONES, S.; HOWE, K.; MARSHALL, M.; SONNHAMMER, E. L. The Pfam Protein Families Database. **Nucleic Acids Research**, [S.l.], v.30, n.1, p.276–280, 2002.

BAZZAN, A. L. C.; DUARTE, R.; PITINGA, A. N.; F., S. L.; SILVA, S. C.; SOUTO, F. A. ATUCG–An Agent-based environment for automatic annotation of genomes. **International Journal of Cooperative Information Systems**, [S.l.], v.12, n.2, p.241–273, June 2003.

BENSON, D.; KARSCH-MIZRACHI, I.; LIPMAN, D.; OSTELL, J.; WHEELER, D. Genbank: update. **Nucleic Acids Research**, [S.l.], v.32, p.D23–D26, 2004.

BERNSTEIN, F.; KOETZLE, T.; WILLIAMS, G.; MEYER JR, J.; BRICE, M.; J., R.; KENNARD, O.; SHIMANOUCHI, T.; TASUMI, M. The protein data bank: a computer-based archival file for macromolecular structures. **Journal of Molecular Biology**, [S.l.], v.2, n.80, p.319–324, 1977.

BRYSON, K.; LUCK, M.; JOY, M.; JONES, D. Applying agents to bioinformatics in GeneWeaver. In: INT. WORKSHOP ON COLLABORATIVE INFORMATION AGENTS, 4., 2000. **Proceedings...** [S.l.: s.n.], 2000. p.60–71. (Lect. Notes in Computer Science, v.1860).

BURGE, C.; KARLIN, S. Prediction of complete gene structure in human genome DNA. **Journal of Molecular Biology**, [S.l.], v.268, p.78–94, 1997.

CLARK, P.; NIBLETT, T. The CN2 Induction Algorithm. **Machine Learning**, [S.l.], v.3, p.261–283, 1989.

DANDEKAR, T. et al. Re-annotating the *Mycoplasma pneumoniae* genome sequence: adding value, function and reading frames. **Nucleic Acids Research**, [S.l.], v.28, p.3278–3288, 2000.

DELCHER, A.; HARMON, D.; KASIF, S.; WHITE, O.; SALZBERG, S. Improved microbial gene identification with Glimmer. **Nucleic Acids Research**, [S.l.], v.27, n.23, p.4636–4641, 1999.

FALQUET, L.; PAGNI, M.; BUCHER, P.; HULO, N.; SIGRIST, C.; HOFMANN, K.; BAIROCH, A. The PROSITE database, its status in 2002. **Nucleic Acids Res.**, [S.l.], v.30, p.235–238, 2002. PubMed: 11752303.

GASTEIGER, E.; JUNG, E.; BAIROCH, A. SWISS-PROT: Connecting biological knowledge via a protein database. **Curr. Issues Mol. Biol.**, [S.l.], v.3, p.47–55, 2001.

HIMMELREICH, R.; HILBERT, H.; PLAGENS, H.; PIRKL, E.; LI, B.; HERRMANN, R. Complete sequence analysis of the genome of the bacterium Mycoplasma pneumoniae. **Nucleic Acids Research**, [S.l.], v.24, p.4420–4449, 1996.

KANEHISA, M.; GOTO, S.; KAWASHIMA, S.; OKUNO, Y.; HATTORI, M. The KEGG resource for deciphering the genome. **Nucleic Acids Research**, [S.l.], v.32, p.D277–D280, 2004.

KULIKOVA, T. et al. The EMBL Nucleotide Sequence Database. **Nucleic Acids Research**, [S.l.], v.32, p.D27–D30, 2004.

KYRPIDES, N. et al. Analysis of the *Thermotoga maritima* genome combining a variety of sequence similarity and genome context tools. **Nucleic Acids Research**, [S.l.], v.28, p.4573–4576, 2000.

KYRPIDES, N.; OLSEN, G.; KLENK, H.; WHITE, O.; WOESE, C. *Methanococcus jannaschi* genome: revisited. **Microb. Comp. Genomics**, [S.l.], v.1, p.329–338, 1996.

LANDER, E. et al. Initial sequencing and analysis of the human genome. **Nature**, [S.l.], v.409, p.860–921, 2001.

LUKASHIN, A.; BORODOVSKY, M. GeneMark.hmm: new solutions for gene finding. **Nucleic Acids Research**, [S.l.], v.26, n.4, p.1107–1115, 1998.

MIYAZAKI, S.; SUGAWARA, H.; IKEO, K.; GOJOTORI, T.; TATENO, Y. DDBJ in the stream of various biological data. **Nucleic Acids Research**, [S.l.], v.32, p.D31–D34, 2004.

MULDER, N. et al. The Interpro Database, 2003 brings increased coverage and new features. **Nucleic Acids Research**, [S.l.], v.31, p.315–318, 2003.

NASCIMENTO, L.; BAZZAN, A. L. An Agent-Based System for Re-annotation of Genomes. In: BRAZILIAN WORKSHOP ON BIOINFORMATICS, 3., 2004, Brasilia, DF. **Proceedings...** [S.l.: s.n.], 2004.

O'BRIEN, P.; NICOL, R. FIPA - towards a standard for software agents. **BT Technology Journal**, [S.l.], v.16, n.3, p.51, 1998.

OUZOUNIS, C.; CASARI, G.; VALENCIA, A.; SANDER, C. Novelties from the complete genome of *Mycoplasma genitalium*. **Molecular Microbiology**, [S.l.], v.20, p.897–899, 1996b.

OUZOUNIS, C.; KARP, P. The past, present and future of genome-wide re-annotation. **Genome Biology**, [S.l.], v.3, n.2, p.1–6, 2002.

POSLAD, S.; BUCKLE, P.; HADINGHAM, R. The FIPA-OS agent platform: Open Source for Open Standards. In: PAAM, 2000, Manchester, UK. **Proceedings...** [S.l.: s.n.], 2000. p.355–368.

RAGHAVAN, S.; OUZOUNIS, C. Novel coding regions in four complete archael genomes. **Nucleic Acids Research**, [S.l.], v.27, p.4405–4408, 1999.

REESE, M. et al. Genome annotation assessment in *Drosophila melanogaster*. **Genome Research**, [S.l.], v.10, n.4, p.483–501, 2000.

SCHROEDER, L. F.; BAZZAN, A. L. C. A Multi-agent System to Facilitate Knowledge Discovery: an Application to Bioinformatics. In: WORKSHOP ON BIOINFORMATICS AND MULTI-AGENT SYSTEMS, BIXMAS, 2002, Bologna, Italy. **Proceedings...** [S.l.: s.n.], 2002. p.44–50.

SMITH, T.; WATERMAN, M. Identification of common molecular subsequences. **Journal of Molecular Biology**, [S.l.], v.147, p.195–197, 1981.

STEIN, L. Genome Annotation: from sequence to biology. **Nature Reviews**, [S.l.], v.2, p.493–503, 2001.

TATUSOV, R. et al. Metabolism and evolution of *Haemophilus influenzae* deduced from a whole-genome comparision with *Escherichia coli*. **Current Biology**, [S.l.], v.6, p.279–291, 1996.

TATUSOV, R. et al. The COG database: an updated version includes eukaryotes. **BMC Bioinformatics**, [S.l.], v.4, p.1471–2105, 2003.

XU, Y.; MURAL, R. J.; SHAH, M. B.; UBERBACHER, E. C. Recognizing Exons in Genomic Sequence Using GRAIL II. In: SETLOW, J. (Ed.). **Genetic Engineering**: Principles and Methods. [S.l.]: Plenum Press, 1994.