Sistema colaborativo para armazenamento e análise de dados de HIV

Luciano Vieira de Araújo

TESE APRESENTADA

AO PROGRAMA INTERUNIDADES EM BIOINFORMÁTICA

DA

UNIVERSIDADE DE SÃO PAULO

PARA OBTENÇÃO DO GRAU DE DOUTOR

EM CIÊNCIAS

ÁREA DE CONCENTRAÇÃO: BIOINFORMÁTICA

ORIENTADOR: Prof. Dr. João Eduardo Ferreira

CO-ORIENTADOR: Profª. Drª. Ester Cerdeira Sabino

São Paulo, fevereiro de 2008

Dedico este trabalho aos meus pais e à minha esposa Lara.

Agradecimentos

O agradecimento é um momento importante dessa caminhada, pois muitas pessoas fizeram parte dessa história e recordá-las é uma oportunidade de consolidar a importância de ser apoiado. Sou grato a todos que das mais variadas formas me ajudaram. Cada conversa, gargalhada, frase, oração e silencio foi especial. Não consigo nominar a todos, mas cada obrigado dito foi com a consciência do apoio recebido. Agradeço a Deus por essa jornada, pelos amigos que encontrei e, em especial, pela saúde e pela vida.

Agradeço a meus orientadores Profª. Dr. Ester C. Sabino e Prof. Dr. João Eduardo Ferreira pela orientação, estímulo e exemplo de dedicação ao trabalho e às pessoas. Vocês foram especiais e marcantes.

Agradeço também ao pessoal do laboratório de Banco de Dados e Bioinformática do IME da USP, Márcio, Kelly e Pedro, pelo companheirismo, revisão de textos, idéias e sugestões. Sou grato também ao Sabri pelo apoio com as seqüências, revisões de texto, dicas e testes das ferramentas de análises.

À Lara, minha admiração e gratidão por sua compreensão e amor, componentes essenciais para essa caminhada.

Por fim, agradeço à minha mãe, Antônia, ao meu pai, Nilson e aos meus irmãos, Lívia e Sérgio, que sempre prestaram seu apoio e me estimularam.

Resumo

Este trabalho apresenta o *DBCollHIV*, um ambiente computacional para estudos sobre o HIV. Ele é composto por um banco de dados e um conjunto de ferramentas de bioinformática para análise dos dados armazenados. O *DBCollHIV* atende desafios biológicos relacionados às pesquisas sobre HIV/AIDS como a criação de software para gerenciamento e análise de dados genéticos, clínicos, epidemiológicos e laboratoriais sobre o HIV em larga escala e com foco na validação da qualidade dos dados. Para tanto, sua arquitetura modular permite adaptar as estruturas de coleta e análise de dados para atender as necessidades apresentadas pelo avanço do conhecimento científico. O conjunto de ferramentas de análise desenvolvidas e disponíveis no *DBCollHIV* é formado pelo software *PCR-Contamination* utilizado para a identificação de possíveis contaminações ocorridas durante a obtenção das seqüências genéticas por PCR; pelo software *HIVSetSubtype* usado para subtipagem automática das seqüências do HIV, sejam de subtipos puros ou recombinantes e pelo software *HIVdag*, um ambiente para criação e execução de algoritmos para identificação de resistência às drogas usadas no tratamento dos pacientes. Com o *HIVdag*, as regras de resistência às drogas são mapeadas para expressões da álgebra de processo, o que permite uso do arcabouço algébrico para geração automática e precisa dos algoritmos propostos pelos pesquisadores. O *DBCollHIV* permite o trabalho cooperativo por meio do compartilhamento de dados gerenciado pelo proprietário do dado e pelo reaproveitamento e evolução de sua estrutura computacional.

Abstract

This work presents DBCollHIV, a computational environment for studies on HIV. It consists of a database and a set of bioinformatics tools for analysis of stored data. The DBCollHIV handles challenges related to biological research on HIV / AIDS, such as the creation of software for managing and analyzing genetic, clinical, epidemiological and laboratory data related with HIV, on a large scale and focused on the validation of data quality. Therefore, its modular architecture allows to adapt the structure of gathering and analyzing data to meet the needs presented by the advance of scientific knowledge. The set of analytical tools developed and available in DBCollHIV is formed by the software PCR-Contamination for identifying carry-over contamination in PCR; the HIVSetSubtype software for automatically subtype sequences of HIV subtypes and the software HIVdag, an environment for creation and execution of algorithms for identification of resistance to drugs used in treatment of patients. With HIVdag rules of drug resistance are mapped to Process Algebra's expressions, which allow the use of the algebraic framework for automatic and accurate generation of algorithms proposed by researchers. Finally, the DBCollHIV enables cooperative work by sharing data managed by the owner of the data and the reuse and evolution of its computational structure.

Índice

Índice de Figuras

Índice de Tabelas

Siglas

AIDS	Acquired Immunodeficiency Syndrome
AP	Álgebra de Processos
ARC	AlgorithmsResultComparison
DBCollHIV	Database System for Collaborative HIV analysis - Sistema de banco de dados cooperativo para análises de dados sobre HIV
DNA	Deoxyribonucleic Acid ou Ácido Desoxirribonucléico
DST	Doença Sexualmente Transmissível
GO	Ação - Go On – indica a continuidade da avaliação da expressão da álgebra de processos.
HAART	Highly Active Antiretroviral Therapy
HIV	Human Imunnedeficiency Virus
LANL	Los Alamos National Laboratory
MS	Ação - MutationSearch
SIDA	Síndrome da Imunodeficiência Adquirida
RENAGENO	Rede Nacional de Genotipagem
RE	Ação - ResultsEquivalence
RNA	Ácido Ribonucléico (do inglês, Ribonucleic Acid)
RS	Ação - ResultsSynchronization
RT	Reverse Transcriptase ou Transcriptase reversa
SC	Ação - ScoreClassification
SRL	Ação - SetResistanceLevel
SS	Ação – SetScore
TARV	Terapia Anti-retroviral
T-CD4	Célula do sistema imunológico destruída pelo HIV.
PCR	Polymerase chain reaction

Capítulo 1

Introdução

Entender a diversidade genética do HIV, vírus da imunodeficiência humana (HIV do inglês Human Imunnedeficiency Virus), e suas conseqüências biológicas é importante para o avanço do combate a AIDS (do inglês, Acquired Immunodeficiency Syndrome). O constante desenvolvimento da tecnologia de seqüenciamento tem aumentado significativamente a capacidade de geração de dados sobre seqüências genéticas. Com o amplo uso de medicamentos anti-retrovirais no tratamento de pacientes infectados pelo HIV, a resistência à droga tem se tornado um importante ponto a ser considerado no tratamento de pacientes. Testes genotípicos de resistência à droga têm mostrado grande benefício ao tratamento de pacientes HIV positivo e seu uso é crescente no acompanhamento de indivíduos com falha terapêutica (Shafer, 2002).

No Brasil, o uso de testes de genotipagem ganhou força com a organização, pelo Ministério da Saúde, de uma rede de laboratórios para realização de testes de genotipagem para pacientes com falha terapêutica (RENAGENO – Rede Nacional de Genotipagem). O projeto RENAGENO prevê a realização anual de 5.000 testes de genotipagem. Além das seqüências genéticas do HIV, também serão obtidos dados clínicos dos pacientes. Um volume crescente de dados como esse demanda ferramentas apropriadas para o gerenciamento, análise e validação da qualidade dos dados produzidos.

Atualmente sistemas gerenciadores de banco de dados que organizam e armazenam seqüências genéticas de HIV juntamente com suas anotações estão disponíveis (http://www.hiv-web.lanl.gov; http://hivdb.stanford.edu) (Kuiken et al., 2003) e são úteis para a obtenção de alinhamentos, seqüências de referência e determinadas análises, porém não possuem dados clínicos e laboratoriais associados às seqüências. Além disso, tais informações, mesmo quando utilizadas em alguns estudos, não ficam disponíveis para uso público, limitando as pesquisas de larga escala às seqüências genéticas e suas anotações.

Assim, um desafio nessa área de pesquisa é a criação de um ambiente de análise de dados sobre HIV que ofereça recursos para:

a) o gerenciamento não somente de seqüências, mas também dos demais dados relacionados ao paciente e à doença;

b) análises de dados em larga escala, que permita a análise dos dados sempre que necessário e que os resultados obtidos sejam armazenados no banco de dados;

c) oferecer facilidades para o acompanhamento do ciclo de análises essenciais para os estudos sobre o HIV.

Este trabalho foi proposto para atender à demanda desse tipo de ambiente integrado e de ferramentas de análise em de HIV/AIDS no Brasil, cuja característica principal é a geração de grande volume de dados clínicos e genéticos. Para tanto, foi desenvolvido o DBCollHIV - Database System for Collaborative HIV analysis (Araújo, et Al, 2006), que é um ambiente para análise de dados de HIV, formado por um banco de dados integrado a ferramentas para análise de seqüências genéticas. O DBCollHIV gerencia dados clínicos, laboratoriais, seqüências genéticas e dados sobre o tratamento do paciente e oferece uma arquitetura que permite a sua expansão para gerenciamento de novos dados e inclusão de novas ferramentas de análise. Dessa forma, o projeto não visa somente a atender às

necessidades atuais, como também permitir a expansão do conjunto de dados gerenciados e das ferramentas de análise e, assim, acompanhar a evolução das pesquisas. Suas funcionalidades estão baseadas em necessidades de grupos de pesquisadores brasileiros vinculados ao programa de DST/AIDS do Ministério da Saúde do Brasil e que podem também ser aplicadas a pesquisas internacionais. Além disso, o DBCollHIV agrega requisitos desejados em sistemas de bioinformática, como: segurança, facilidade de integração de ferramentas de análise e compartilhamento de dados.

O DBCollHIV contempla soluções na área computacional e biológica. Na área computacional, são usados recursos de arquitetura de banco de dados para criar um ambiente modular e flexível (Barrera et al., 2004; Ferreira e Busichia, 1999) para permitir a expansão do ambiente de coleta e análise de dados. Para controle de execução, são utilizados recursos de encadeamento de processos que permitem a utilização de ferramentas de análise como módulos a serem encadeados em uma seqüência de análise (Oikawa et al., 2004). Ainda na área computacional, foi desenvolvido um ambiente para criação e comparação de testes de genotipagem baseado em álgebra de processos (Folkkink, 2000). Quanto à parte biológica, ferramentas para a análise do HIV foram criadas para automatizar tarefas essenciais para avaliação das seqüências do vírus. Além disso, a criação automática de testes de genotipagem baseados em álgebra de processos potencializa os estudos sobre resistência a drogas por meio do mapeamento das regras de resistência em expressões da álgebra de processos, o que permite a criação automática do programa para aplicação das regras e comparação dos resultados obtidos por diferentes conjuntos de regras.

1.2 Objetivos

Este trabalho tem como objetivo produzir um ambiente para análise e armazenamento de dados de HIV como forma a apoiar projetos de pesquisa nessa área.

Nessa perspectiva, o banco de dados deve armazenar e gerenciar dados relacionados a diferentes tópicos que envolvem o tratamento de pacientes HIV positivo, como: informações clínicas, laboratoriais, epidemiologias, histórico de uso de drogas e informações genéticas sobre a cepa viral. Além disso, seu projeto deve prever facilidades para a realização de alterações necessárias a sua adequação aos avanços científicos.

As ferramentas de análise de dados devem oferecer análises essenciais à realização de projetos de pesquisa sobre o HIV e permitir que as análises sejam refeitas sempre que desejado.

De maneira geral, o ambiente tem como meta favorecer a cooperação entre grupos de pesquisas por meio do compartilhamento de dados e de uma estrutura comum para seu gerenciamento e análise.

1.3 Contribuições do trabalho

Como contribuições este trabalho apresenta um ambiente integrado para análise de dados de HIV, que permite ao pesquisador organizar e analisar seus dados. Os dados armazenados no DBCollHIV podem ser compartilhados com outros usuários sempre que desejado pelo proprietário do dado. O conjunto de ferramentas criadas por este trabalho é formado pelo PCR Contamination usado para identificar possíveis contaminações durante a obtenção de seqüências do HIV por PCR, O HIVSetSubtype que classifica as seqüências de acordo com o seu subtipo e o HIVdag com a qual é possível criar e avaliar regras para algoritmos de resistência à droga e mesmo comparar os resultados de algoritmos. Para o desenvolvimento do HIVdag, foi definido um mapeamento das regras dos algoritmos de genotipagem para expressões da álgebra de processos.

1.4 Organização do texto

Este trabalho está organizado em cinco capítulos, como descrito a seguir: Capítulo 1 – Introdução com a motivação para criação do ambiente de análise e seus objetivos; Capítulo 2 – Fundamentos de biologia e computação necessários para a realização e compreensão do trabalho; Capítulo 3 – Trabalhos relacionados; Capítulo 4 – Descrição do Ambiente DBCollHIV e suas ferramentas, e Capítulo 5 – Conclusões e futuras pesquisas. Referências bibliográficas.

Capítulo 2

Fundamentos

Este capítulo aborda conceitos biológicos e computacionais utilizados ao longo deste trabalho. Os conceitos biológicos abrangem informações sobre o HIV, sua forma de infecção, exames, tratamentos com uso de drogas, seqüências genéticas e mutações do vírus. Os conceitos computacionais incluem conceitos de bancos de dados modulares e álgebra de processos.

2.1 Fundamentos Biológicos

O HIV é o responsável pela infecção de milhares de pessoas no Brasil e no mundo. Devido à sua capacidade de infectar e de causar a morte de seres humanos, tem sido foco de inúmeras pesquisas científicas e ações governamentais que visam ao seu combate. Entre as características mais marcantes do HIV destacam-se: o ataque às células do sistema imunológico e sua capacidade de mutação. Tais características amplificam os efeitos da infecção pelo vírus, dificultam o seu combate e, também, o desenvolvimento de vacinas.

Os estudos sobre o HIV podem apresentar diferentes perspectivas como: desenvolvimento de vacinas, prevenção da doença, tratamento do paciente, impacto social, etc. Entre as possíveis abordagens, o DBCollHIV oferece recursos para trabalhar com dados referentes ao tratamento do paciente e sobre informações genéticas contidas na seqüência de

DNA do vírus. Com esse foco, a seguir são apresentados fundamentos biológicos relacionados ao conjunto de dados coletados.

2.1.1 Ácidos nucléicos.

Os ácidos nucléicos ADN (Ácido Desoxirribonucléico) ou DNA (do inglês, Deoxyribonucleic Acid) e ARN (Ácido ribonucléico) ou RNA (do inglês, Ribonucleic Acid) são moléculas centrais no processo de reprodução e transmissão de características hereditárias (Watson et al., 2003).

O DNA é uma molécula formada por duas cadeias em forma helicoidal, constituídas de um açúcar, chamado desoxirribose, por um grupo fosfato e por uma base nitrogenada. Essa combinação de açúcar, grupo fosfato e base nitrogenada forma um nucleotídeo. Na estrutura do DNA são encontrados quatro tipos de nucleotídeos, são eles: Adenina (A), Citosina (C), Timina (T) e Guanina (G). A Figura 2.1 mostra as duas fitas do DNA unidas pelas ligações entre os nucleotídeos. No DNA, a Adenina se liga à Timina e a Citosina se liga à Guanina. Essa especificidade de ligação entre os nucleotídeos é uma importante característica para a formação correta da molécula.

O RNA é uma molécula de cadeia simples composta pelo açúcar ribose, um grupo fosfato e pelos nucleotídeos Uracila (U), Adenina (A), Citosina (C) e Guanina (G). Diferente do DNA, a Uracila se liga à Adenina na composição da molécula de RNA.

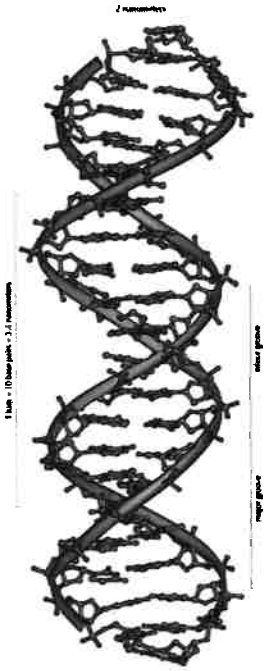

Figura 2.1: DNA e sua cadeia dupla em forma helicoidal (Fonte: Wikipédia)

A ordem em que os nucleotídeos aparecem na cadeia da molécula de DNA é conhecida como seqüência do DNA e está armazenada a informação genética do indivíduo. Uma unidade importante dentro da seqüência de nucleotídeos é o códon, formado por um conjunto de três nucleotídeos que definem os aminoácidos, como mostra a Figura 2.2. Por sua vez, os aminoácidos são as unidades formadoras das proteínas (Watson et al., 2003).

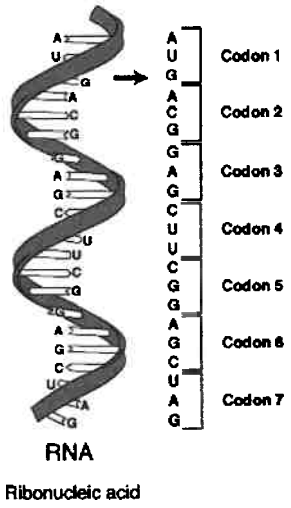

Figura 2.2: Códons de um trecho de uma molécula de RNA (Fonte: Wikipédia).

O DNA participa de dois processos importantes na transmissão de informação hereditária. No primeiro, chamado de replicação, a molécula de DNA é copiada. Nele, as ligações entre as fitas do DNA são rompidas e cada fita simples passa a funcionar como molde para inserção dos nucleotídeos que formarão a fita complementar, dando origem a duas novas moléculas de DNA, como mostra a Figura 2.3. As fitas de DNA possuem direção de leitura, conhecidas como 5'e 3'. A direção 5' é identificada por possuir um fosfato ligado ao carbono número 5 da pentose e a direção 3' possui um grupo hidroxil livre ligado ao carbono número 3 da pentose. Como as fitas de DNA são antiparalelas, a direção de leitura de uma fita é 5' – 3', enquanto a outra é 3' – 5'.

O segundo processo é conhecido como transcrição, nele são sintetizadas moléculas de RNA. A partir do RNA proteínas são sintetizadas no processo chamado de tradução. Nele, o mRNA indica para o tRNA a ordem de inserção dos aminoácidos, levando à formação das proteínas.

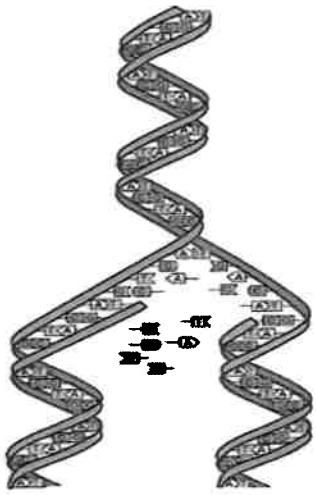

Figura 2.3: Replicação do DNA (Fonte: Wikipédia).

2.1.2 Estrutura do HIV

HIV é um retrovírus e seu material genético é formado por RNA, que é sintetizado em DNA através da enzima viral transcriptase reversa (TR ou RT) e, posteriormente, integrado à célula do hospedeiro (Coffin 1992 a, b, 1996). Além dos genes principais *env*, *gag* e *pol*, comuns aos retrovírus, o HIV possui genes regulatórios e auxiliares como *tat*, *rev*, *nef* que o torna diferente de outros retrovírus (Greene, 1991).

2.1.3 Infecção

O HIV ataca o sistema imunológico por meio da infecção e destruição de células do sistema imunológico conhecidas como células T-CD4. A redução na quantidade dessas células impede uma resposta imunológica e permite a manifestação de doenças oportunistas, tais como: Candidíase, Tuberculoses, Pancreatite, Sarcoma de Kaposi que caracterizam a Síndrome da Imunodeficiência Humana (AIDS) (WHO, 2005; CDC, 1993).

Uma vez dentro da célula hospedeira, o HIV usa enzimas RT para sintetizar DNA a partir do RNA (Baltimore, 1970; Temin e Mizutani, 1970). A RT é uma das responsáveis pela taxa de mutação ou variabilidade genética do HIV, que é aproximadamente 10^{-4} bases em cada ciclo replicativo (Preston, et al., 1988), ou seja, um erro a cada ciclo replicativo. Com essa freqüência de erro/mutação a população de retrovírus, como o HIV, praticamente não apresenta genomas idênticos e são conhecidas como "quasispecie" (Holland, et al., 1992).

A mutação é uma mudança permanente que ocorre no DNA. Ela é chamada de permanente, pois não foi corrigida pelos mecanismos de correção que atuam durante o

processo de cópia do DNA. Logo, ela passa a fazer parte do DNA do organismo. A mutação é caracterizada pela inserção, remoção ou alteração de um ou mais nucleotídeos.

Uma mutação na seqüência do DNA pode resultar na codificação de aminoácidos diferentes dos codificados pelo DNA original. Porém, nem sempre uma mutação implica na codificação de um novo aminoácido, pois a codificação de um aminoácido é definida por um códon, porém, como o código genético é degenerado, um aminoácido pode ser codificado por diferentes códons (Watson et al., 2003). Por exemplo, a Serina – Ser, que pode ser codificada pelos códons UCU, UCC, UCA e UCG; ou mesmo, os códons UAA, UAG, UGA, que não codificam aminoácidos e, portanto, são conhecidos como códons de parada (do inglês, stop codon). Assim, uma mutação na terceira posição do códon UCU que resulte na troca do nucleotídeo U pelo C não alteraria o aminoácido codificado.

2.1.4 Variabilidade genética

A variabilidade genética do HIV permite que esse se adapte às mudanças do ambiente e escape do sistema imunológico de seu hospedeiro. De acordo com a variação genética, o HIV está classificado em tipos, grupos e subtipos. Quanto ao tipo, o HIV é classificado em HIV-1 ou HIV-2 (Robertson, et al., 2000). O HIV-1 é responsável pela epidemia mundial, enquanto o HIV-2 é menos freqüente e aparenta ser menos patogênico, sendo raro e predominantemente encontrado no oeste da África. A definição de subtipos do HIV é baseada em análise filogenética e agrupa cepas virais de acordo com um possível ancestral comum (Wainberg, 2004). Vários trabalhos, como Baeten, et al., 2007; Laeyendecker, et al., 2006; Kanki, et al., 1999; buscam entender se esta classificação tem impacto na função biológica, ou seja, se os subtipos apresentam diferenças em relação à capacidade de transmissão, patogenicidade e resposta a determinados tratamentos. Para que

estes estudos sejam realizados, no entanto, é necessário que a classificação da cepa, presente em cada indivíduo, seja feita de forma precisa. Os subtipos do HIV podem ser puros ou recombinantes, de acordo com o número de ancestrais ou histórias evolutivas representadas pela cepa do vírus, ou seja, os subtipos puros representam seqüências com material genético de um ancestral de subtipo único e os subtipos recombinantes indicam cepas formadas por um mosaico de material genético originário de mais de um subtipo, resultado de uma recombinação de DNA (Posada D e Crandall KA, 2002).

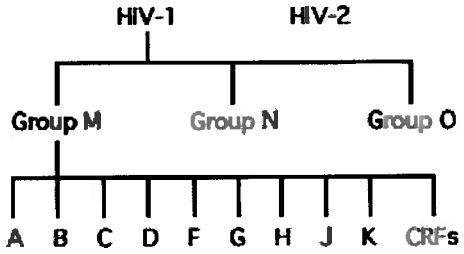

Figura 2.4: Classificação subtipos HIV (Fonte: http://www.avert.org/hivtypes.htm).

Conforme mostra a Figura 2.4, as cepas do HIV-1 podem ser classificadas em 3 grupos, são eles: M (do inglês Major), N (do inglês NEW) e O (do inglês Outlier). Por fim, as cepas de cada subgrupo são classificadas em subtipos. Como exemplo pode-se citar as cepas do subtipo M, responsáveis pela maioria das infecções causadas pelo HIV-1, que são classificadas em 11 subtipos puros (A1, A2, B, C, D, F1, F2, G, H, J e K) e mais de 30 formas recombinantes ou CRFs (do inglês, Circulating Recombinant Forms) (Robertson, et al., 1999). O National Laboratory of Los Alamos mantém em seu site (http://hiv-web.lanl.gov/) uma relação atualizada com as formas recombinantes já identificadas e um mapa sobre a distribuição de subtipos no mundo (Figura 2.5).

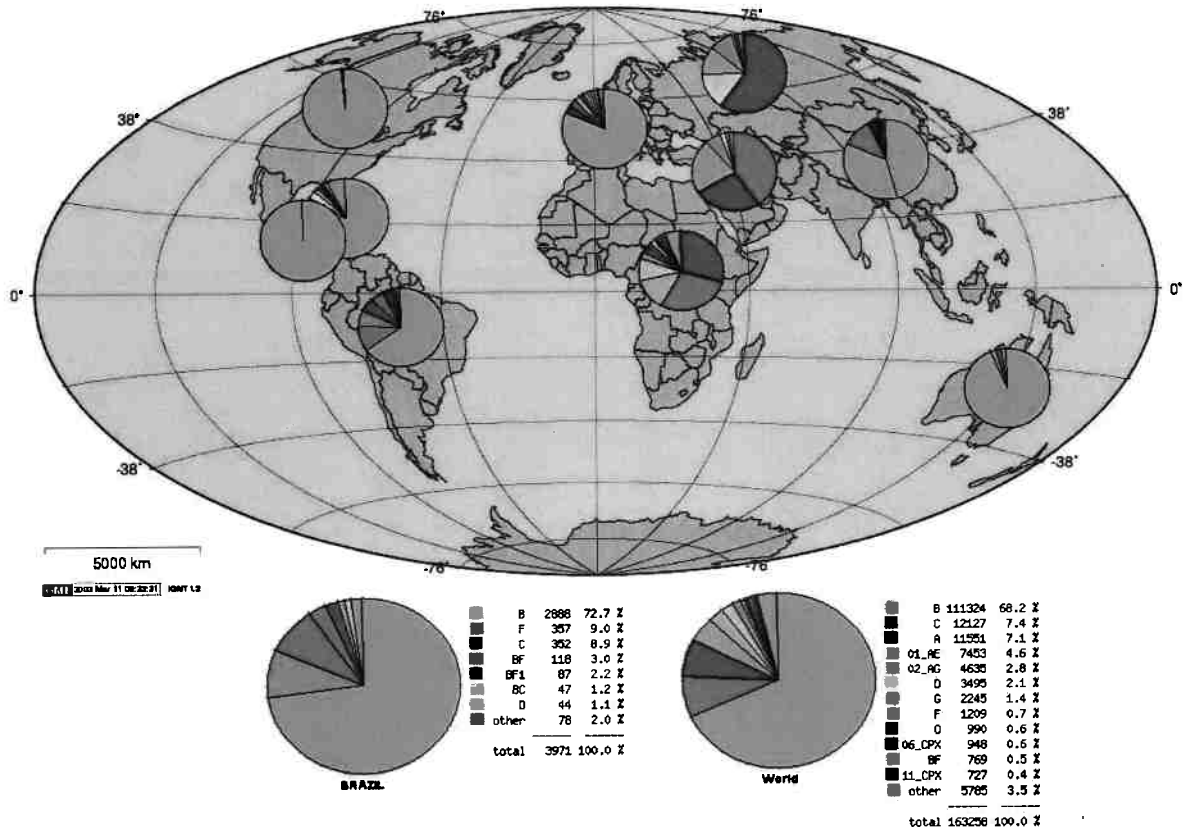

Figura 2.5: Distribuição de subtipos pelo mundo. Em destaque distribuição brasileira e mundial. (Fonte: http://hiv-web.lanl.gov/).

2.1.5 Detecção da infecção pelo HIV

A detecção da presença do HIV no organismo é feita de forma indireta através da detecção de anticorpos específicos por meio de exames como: ELISA e Western-Blot.

O ELISA (Proffitt e Yen-Lieberman, 1993; George e Schochetman, 1994) é um teste de alta sensibilidade, porém sua especificidade não garante a ausência de resultados falso positivos. Em outras palavras, se o HIV estiver presente, existe uma alta probabilidade de o exame ELISA detectá-lo e gerar um resultado positivo. Porém, nem todo resultado positivo implica na presença do vírus. Por esse motivo, é necessário um teste para confirmar a presença do

HIV, entre os vários testes confirmatórios, o Western-Blot é o mais conhecido, que avalia a presença da proteína do vírus (Gürtler, 1996).

2.1.6 Tratamento do paciente HIV+

O tratamento do paciente HIV+ consiste no uso de medicamentos anti-retrovirias que inibem a reprodução do HIV no organismo. Esse tratamento é conhecido como Terapia Anti-retroviral (TARV) e atualmente conta com mais de 20 Drogas. Tais drogas buscam inibir alguns dos mecanismos usados pelo vírus para sua replicação. Baseado nessa característica, elas podem ser classificadas em quatro classes, de acordo com o seu foco de atuação/inibição:

- Inibidores de transcriptase reversa RT:
    - Análogos de nucleosídeos ou nucleotídeos;
    - Não análogos dos nucleotídeos;
- Inibidores de protease;
- Inibidores de fusão (impedem a entrada do vírus na célula);
- Inibidores da integrase.

A Tabela 2.1 mostra exemplo de drogas classificadas de acordo com o seu mecanismo de atuação.

Inibidores de TR Análogos aos nucleotídeo (NRTI)	Inibidores de TR Não Análogos aos nucleotídeos (NNRTI)	Inibidores de protease	Inibidores de fusão
ABC – Abacavir (Ziagen)	DLV - Delavirdina (Rescriptor®)	APV - Amprenavir (Agenerase®)	T-20 (Enfurvitide, Fuzeon®)
AZT – Zidovudina (Retrovir®)	EFV - Efavirenze (Sustiva® ou Stocrin™)	ATV - Atazanavir (Reyataz ®)	
ddC - Zalcitabina (Hivid®)	NVP - Nevirapina (Viramune®)	FPV - Fosamprenavir (Telzir®, Lexiva®)	
ddI - Didanosina (Videx®)		IDV - Indinavir (Crixivan®)	
d4T - Stavudina (Zerit®)		LPV - Lopinavir/r (Kaletra®).	
FTC- Emtricitabina (Emtriva®)		NFV - Nelfinavir (Viracept®)	
3TC - Lamivudina (Epivir®)		RTV - Ritonavir (Norvir®)	
TDF - Tenofovir (Viread®)		SQV- Saquinavir (Invirase500™)	
CBV – Combinavir (AZT+3TC)		TPV - Tipranavir (Aptivus®)	

Tabela 2.1: Drogas anti-retrovirais. (Fonte: Amadeo - HIVmedicine2007; adaptada)

O tratamento necessariamente precisa ser realizado com múltiplas drogas para evitar surgimento de resistência (Hammer, et al., 1996; Saravolatz, et al., 1996). O uso de combinação de droga, conhecido como Terapia Anti-retoviral de Alta Atividade ou HAART (do inglês, Highly Active Antiretroviral Therapy), mudou a história da doença e diminuiu drasticamente a mortalidade e a mortandade entre os pacientes (Alter, 2003; Ortiz, et al., 2002, Oxenius,et al., 2000). O tratamento, porém, é complexo e falhas são comuns e levam ao surgimento de cepas resistentes (Deeks, 2003). Para melhor seguimento desses pacientes foram desenvolvidos testes de genotipagem, que associa dados de seqüência viral à

suscetibilidade às drogas, o que permite ao médico decidir qual o melhor tratamento a ser ministrado ao paciente.

2.1.7 PCR

A técnica de PCR (Polymerase Chain Reaction) (Mullis,1986) sintetiza de maneira exponencial cópias de um segmento especifico de DNA (*seqüência-alvo*). Para tanto, são necessários iniciadores ("primers") que são seqüências curtas de DNA complementares a seqüência que flanqueiam o fragmento-alvo. Além dos iniciadores e do DNA a ser copiado, a reação de PCR necessita de:

- Desoxinucleotídeos trifosfatados de adenina(+dATP), de citosina(+dCTP), de timina(+dTTP) e de guanina (+dGTP), usados para compor a seqüência copiada;
- Enzima DNA polimerase de *T. aquaticus*, denominada Taq polimerase, é a responsável pela introdução dos desoxinucleotídeos trifosfatados na seqüência a partir dos iniciadores após hibridação com fita simples do DNA que lhe servirá de molde.

A reação de PCR é feita em um equipamento conhecido como termociclador, cuja função é executar ciclos de aquecimento e resfriamento da mistura de reagentes de forma a executar as três diferentes etapas do PCR (Figura 2.6), descritas a seguir:

1. Desnaturação - Fase em que os componentes da reação são aquecidos a temperaturas de 94 a 96 °C. Nessa temperatura, as pontes de hidrogênio que ligam as duas fitas do DNA se rompem dando origem a duas fitas simples, que servirão de molde para a cópia do DNA.

2. Hibridação dos iniciadores (primers) – Com as fitas separadas, a reação é resfriada a temperatura própria para a hibridação dos iniciadores, em geral próxima a 55 °C. A hibridação ocorre após alguns segundos nessa temperatura.
3. Extensão dos iniciadores ou alongamento – Após a hibridação dos iniciadores, a reação é aquecida à temperatura de ação da Taq polimerase, aproximadamente de 72 °C. A essa temperatura, a Taq polimerase inicia a inserção de nucleotídeos trifosfatados a partir dos iniciadores e usando como molde a fita simples do DNA. Dessa maneira, uma nova fita de DNA é formada e as duas fitas simples de DNA, obtidas na desnaturação, formam duas moléculas de DNA com fita dupla. Esse processo demora aproximadamente 2 minutos e as cópias produzidas são conhecidas como *amplicons*.

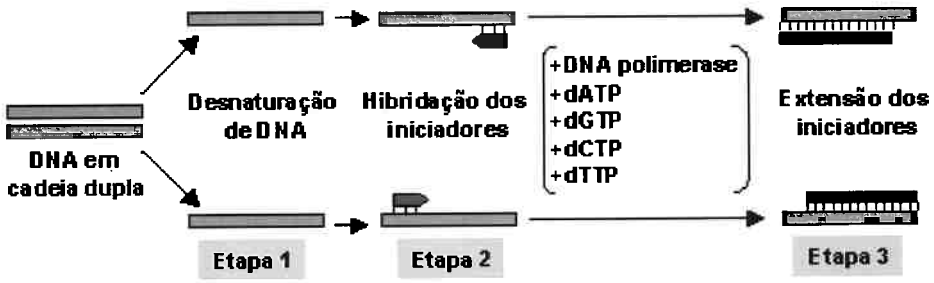

Figura 2.6: Etapas do processo de PCR (Fonte: http://www.e-escola.pt/site/topico.asp?topico=339).

As três fases do PCR são repetidas em ciclos e, após cada um deles, são produzidas 2^n cópias da seqüência alvo ou amplicons, onde **n** representa o número do ciclo atual. Assim, considerando uma eficiência máxima, após 30 ciclos, são produzidos 2^{30} amplicons.

2.1.8 Seqüenciamento de DNA

O seqüenciamento do DNA identifica a seqüência de nucleotídeos que forma o DNA analisado. Um dos métodos de seqüenciamento mais conhecidos é o Sanger (Sanger, et al., 1977). Nesse método, além do DNA-alvo usado como molde dos iniciadores e dos nucleotídeos para extensão da cadeia (dATP, dCTP, dTTP, dGTP), são usados também nucleotídeos terminadores de cadeia, conhecidos como dideoxi-nucleotídeos. Por não possuírem o grupo hidroxil (-OH) no carbono 3 da pentose, ao serem inseridos, eles impossibilitam a ligação dos próximos nucleotídeos e interrompem a síntese da cadeia. Além da falta do grupo hidroxil, os terminadores são marcados por fluorescência que emitem luz de cor específica ao serem excitados por um laser. Como são usados terminadores correspondentes aos quatro nucleotídeos (A, C, T, G), quando um deles é inserido na cadeia, o processo de síntese é terminado e o nucleotídeo correspondente pode ser identificado ao passar pelo laser e ter sua cor ativada.

Durante a reação, os iniciadores anelam no início da DNA-alvo que é estendido pela Taq polimerase até que um terminador seja inserido na cadeia. O produto da reação é colocado em um canal de seqüenciamento para a realização da eletroforese. A eletroforese é uma técnica de separação de moléculas baseada no tamanho. As moléculas contidas no gel são submetidas a uma diferença de potencial elétrico para a sua separação. Como a carga global da fita do DNA é negativa, as moléculas migram para o pólo positivo no final da placa. As moléculas de menor tamanho e peso molecular migram mais rapidamente que as moléculas maiores e mais pesadas, criando uma ordenação dos fragmentos. A passagem dos terminadores pelo laser de leitura é automaticamente identificada pelo computador que reconhece o nucleotídeo do terminador pela cor emitida e registra a ordem de passagem de

cada um deles. Ao final do processo é gerada uma imagem, chamada de cromatograma, com os picos, representando a intensidade das cores lidas, e letras, representando os nucleotídeos correspondentes aos picos, como mostra a Figura 2.7.

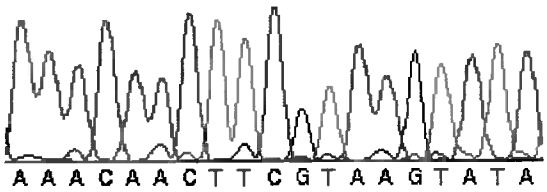

Figura 2.7: Imagem do cromatograma com a seqüência de nucleotídeos e os picos correspondentes. (Fonte: Wikepedia)

A leitura e interpretação do cromatograma dá origem a um arquivo contendo a seqüência dos nucleotídeos referentes ao DNA estudado. Normalmente, esses arquivos são gravados em formato FASTA. O arquivo no formato FASTA, conforme descrito no site do NCBI (http://www.ncbi.nlm.nih.gov/blast/fasta.shtml), começa com uma linha de comentário iniciada pelo sinal de maior (>). Nas demais linhas do arquivo são representados os nucleotídeos ou aminoácidos que formam a seqüência genética. Tais linhas devem ter tamanho menor que 80 caracteres. Esse formato de arquivo é amplamente utilizado por programas de análise de seqüências.

2.1.9 Testes de genotipagem

O uso de medicamentos anti-retrovirais trouxe muitos benefícios ao tratamento do paciente HIV+, porém sua eficiência é limitada pela alta replicação e capacidade de mutação do HIV que geram cepas resistentes as quais acabam sendo selecionadas durante o tratamento (Hammer, et al., 1996; Saravolatz, et al., 1996). Os testes de genotipagem identificam as mutações presentes no genoma viral e, com isso, é possível inferir quais medicamentos possuem maior chance de serem mais eficazes. Os testes de fenotipagem

detectam o efeito biológico das mutações através do cultivo do vírus na presença de diferentes concentrações das drogas. Os resultados obtidos com a cepa do paciente são comparados com aqueles obtidos com uma cepa padrão e assim é inferido se a cepa é sensível ou resistente ao medicamento (Petropoulos, et al., 2000).

Tendo como base informações de testes de fenotipagem, informações clínicas e laboratoriais grupos de pesquisas como Stanford, no EUA, ANRS, na França, e RENAGENO, no Brasil, são criadas regras para avaliar as mutações encontradas no vírus e definir o nível de resistência às drogas existentes. O resultado do teste de genotipagem indica o nível de resistência do vírus a cada uma das drogas, cabe ao médico a decisão sobre quais medicamentos o paciente deve usar. A Figura 2.5 mostra um exemplo de relatório de teste de genotipagem, baseado nas regras desenvolvidas pelo comitê brasileiro do projeto RENAGENO.

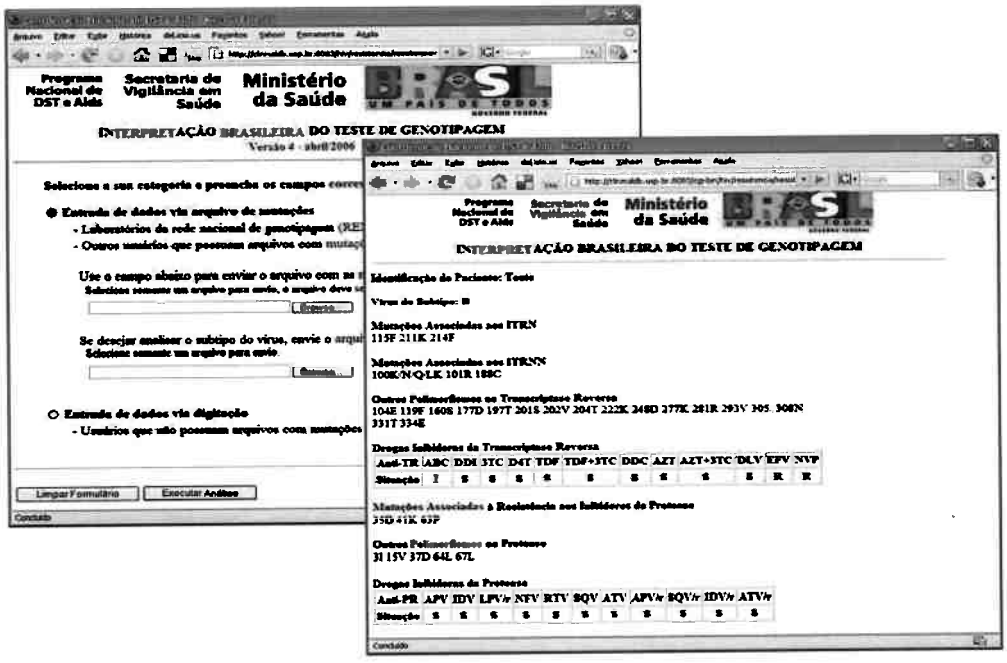

Figura 2.8: Teste de genotipagem brasileiro – Tela de entrada de dados e relatório de resistência.

Os algoritmos para interpretação de resistência ou suscetibilidade a drogas estão baseados em dois conjuntos principais de informação. As mutações existentes no vírus e as regras que as analisam.

A identificação das mutações no DNA do vírus do paciente é feita através do alinhamento e comparação de sua seqüência de aminoácidos com a da cepa HXB2, considerada padrão (GenBank Accession Number K03455) (Korber, et al., 1998). Com esta comparação pode-se gerar uma lista com todas as mutações encontradas e que são avaliadas de forma diferente em cada regra do algoritmo. Como resultado, o algoritmo apresenta um laudo com as mutações encontradas no vírus e o provável impacto que elas causam a cada uma das drogas.

2.2 Fundamentos Computacionais

Os sistemas de computação desenvolvidos para atender às necessidades de pesquisas científicas não podem ser limitados por conceitos estáticos, uma vez que o contexto para o qual foram desenvolvidos se encontra em constante mudança. Assim sendo, tais sistemas devem ter capacidade de agregar ou eliminar características, funcionalidades e, até mesmo, dados de acordo com a evolução do conhecimento científico e/ou da necessidade da pesquisa. Com essa perspectiva, nessa seção são apresentados conceitos sobre banco de dados e sistemas modulares e álgebra de processos.

2.2.1 Sistemas de banco de dados modulares

Uma maneira de permitir o desenvolvimento de sistemas flexíveis para integração de aplicações heterogêneas é por meio da construção de sistemas de banco de dados modulares (Ferreira e Busichia, 1999). Nesses sistemas, cada unidade computacional é formada pelos módulos mais adequados à realização de determinada tarefa, inclusive, diferentes versões de um módulo também podem ser consideradas na escolha dos módulos mais adequados.

A identificação de módulos em um sistema é baseada no relacionamento entre suas funcionalidades e sua respectiva necessidade de dados (Sommerville, 2006), ou seja, o módulo deve ser capaz de executar uma determinada tarefa e, para tanto, deve conter todas as funcionalidades e dados necessários.

Para alcançar autonomia de administração de dados dos subsistemas de uma aplicação, é necessário projetar o banco de dados de forma modular, para, assim, garantir que cada módulo possua seu próprio repositório de dados contendo os dados necessários para suas transações. Portanto, a idéia básica da modularização de banco de dados é dividir o esquema global de dados da aplicação em subesquemas e reduzir a interseção dos subesquemas que caracterizam dados compartilhados por diferentes módulos do sistema (Özsu e Valduriez,1999; DAFTG,1986).

A redução das interseções diminui a interdependência de dados entre os módulos conhecida como acoplamento. Quanto menor o acoplamento entre os módulos, mais independentes eles são e mais fácil é a inclusão ou remoção dos mesmos em um sistema.

A Figura 2.9 mostra um diagrama simplificado de um banco de dados modular para dados de pacientes HIV+. Nela, o módulo de paciente ocupa a região central da figura e apresenta relacionamento com todos os módulos. Nesse caso, o módulo paciente possui alto

acoplamento e sua remoção envolve alteração em todos os módulos. Porém, o alto acoplamento do módulo paciente não é um problema, uma vez que ele é o principal módulo do sistema e, sem ele, o sistema praticamente não faz sentido. Já o módulo de doenças apresenta ligação somente com o módulo paciente, o que representa o seu baixo acoplamento e conseqüente facilidade para sua alteração/remoção. Os módulos Tratamento e Vírus apresentam várias ligações com o módulo paciente, além de ligação entre eles. Tais módulos possuem alto acoplamento, o que pode resultar em grande impacto caso ocorra alguma alteração no módulo paciente.

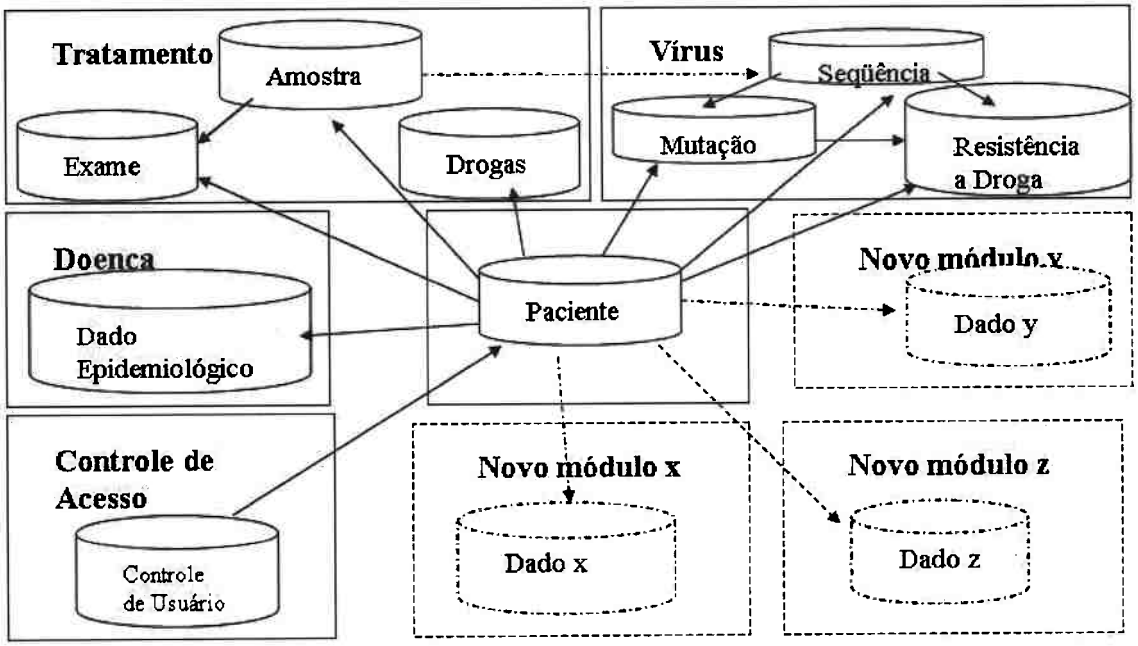

Figura 2.9: Diagrama de um banco de dados modular para HIV.

O acoplamento entre os módulos pode ser reduzido diminuindo ao máximo a necessidade de dados de outros módulos (Sommerville, 2006), porém, deve ser preservada capacidade de representação do banco de dados, sua eficiência e integridade. Como exemplo, de redução de acoplamento, na Figura 2.9, a ligação entre o módulo paciente e os

dados sobre as mutações e resistência a drogas poderiam ser removidas sem prejuízo para representação do modelo, desde que os dados sobre as seqüências estejam presentes e permitam a recuperação das informações sobre as mutações e sobre a resistência a drogas.

Caso novos módulos sejam criados, como exemplificado por módulos hipotéticos x, y e z, o nível de acoplamento desses módulos indicará a complexidade para a introdução dos mesmos no sistema. Caso possuam apenas ligações com o módulo paciente, como na Figura 2.9, esse processo será o mais simples possível do ponto de vista de relacionamento de dados.

Ainda em busca de flexibilidade e facilidade de integração, os módulos devem oferecer recursos para exportação de dados em diferentes formatos, tais como: o XML (Achard et al., 2001) e o FASTA para seqüências genéticas e para os demais dados. O uso de formatos de arquivos amplamente conhecidos facilita o acesso aos dados. Além disso, a exportação de dados em diferentes formatos colabora para a substituição de módulos e para o uso dos mesmos em outros sistemas.

2.2.2 Álgebra de processos.

Entender e representar os processos presentes nas diferentes áreas de conhecimento tem sido o ponto de partida para a concepção de diversos sistemas computacionais (Aalst et al., 2003). Quanto mais estruturado e definido for um processo melhor será a sua representação em um sistema computacional. A representação de processo pode ser feita utilizando diferentes linguagens e/ou representações gráficas, como é o caso de redes de Petri (Murata, 1989), bem conhecida devido à facilidade de representação gráfica dos processos. Apesar da popularidade das redes de Petri, este trabalho está baseado na Álgebra de processos – AP (Folkkink, 2000), que, assim como as redes de Petri, oferece representação de processos e propriedades matemáticas para sua avaliação. Entre as vantagens da álgebra de processo em relação à rede de Petri apresentada em Best et al. (2001), destacam-se :

- Oferecer um conjunto de leis algébricas que permitem refinar a especificação de um sistema, provar sua corretude e mesmo permitir seu gerenciamento;
- Permitir a composição estruturada de sistemas mais amplos usando outros sistemas;
- Permitir o uso de álgebras derivadas ou concomitantes para facilitar a compreensão, definição e manipulação dos sistemas.

A álgebra de processos é um arcabouço formal para representação de processos por meio de uma coleção operadores. A avaliação das expressões da AP permite detectar propriedades indesejáveis e formalmente produzir propriedades desejáveis (Folkkink, 2000). Uma álgebra de processos é formada por um conjunto de operadores e seus respectivos

axiomas que definem as regras e comportamentos de seus operadores. Normalmente, uma álgebra de processos oferece operadores básicos que definem processos finitos, operadores de comunicação usados para definir processos concorrentes e operadores que permitem a execução recursiva dos processos. Além disso, novos operadores podem ser criados para facilitar a representação e gerenciamento de processos (Bergstra et al., 2001). Essa possibilidade de criação de novos operadores permite a criação de diferentes álgebras de processos, como: CCS (*Calculus of Communicating Systems*) (Milner, 1982), CSP (*Communicating Sequential Processes*) (Hoare, 1978) e a ACP (*Algebra of Communicating Processes*) (Achard et al., 2001) (Bergstra, et al., 2001), usada com referência neste trabalho.

A NPDL utiliza a álgebra de processos ACP como arcabouço formal para oferecer subsídios para a criação, execução e gerenciamento de processos descritos nas expressões da álgebra de processos. Além dos operadores da álgebra de processos, a NPDL também conta com operadores extras, usados para o gerenciamento das execuções das expressões.

A seguir são apresentados os operadores da AP e NPDL utilizados na representação de regras de mutações. A descrição mais detalhada de todos os operadores da álgebra de processos pode ser encontrada em Folkkink, 2000 e Bragetto, et al., 2007. Para apresentar os operadores, são usados os termos t_1 e t_2 que representam termos de um processo.

Operadores da álgebra de processos:

1. Composição alternativa "+" – Na expressão $t_1 + t_2$, o operador de composição alternativa + define um processo, onde ambos os termos t_1 e t_2 são executados, porém um de cada vez, ou seja, o processo pode executar t_1 e t_2 ou t_2 e t_1;
2. Composição seqüencial "." – Na expressão $t_1 . t_2$, o operador . define o processo que executa t_2 após a conclusão da execução de t_1;

3. Composição paralela "||" - Na expressão t_1 || t_2, o operador || define o processo que executa t_1 e t_2 simultaneamente;

Operador da NPDL

1. Execução condicional "% *r*" - Na expressão % *r* t_1, o operador % r define o processo que executa t_1 somente se a regra booleana *r* gerar um valor verdadeiro. O complementar da execução condicional é representado por "% ! *r*", nesse caso, t_1 somente será executado se a regra *r* gerar um valor falso. O operador % pode ser associado a uma *ação silenciosa* que não executa nenhuma tarefa, ela apenas indica que a avaliação da expressão deve continuar.

O operador execução condicional da "% r" da NPDL desempenha um papel importante para o mapeamento de processos, pois permite decidir, em tempo de execução, qual parte da expressão deve ser executada. A expressão *P = %r (A .B) + %!r (C||D)* mostra um exemplo de expressão da NPDL. Nela, P indica um processo que avalia o resultado da regra *r* para decidir qual ação deve ser executada. Se a regra *r* gerar valor verdadeiro, a ação A será executada e somente após seu término, a ação B será executada. Caso o valor gerado por *r* seja falso, as ações C e D serão executadas simultaneamente.

Em resumo, o uso da álgebra de processos permite uma validação e verificação de propriedades de um sistema de computação. Dessa maneira, os sistemas de computação podem ser avaliados e evoluídos por meio de processos algébricos.

A linguagem Navigation Plan Definition Language (NPDL) (Bragetto et al., 2007) encapsula as vantagens da álgebra de processos e permite criar, executar e gerenciar sistemas de computação definidos por meio de expressões da álgebra de processos. Assim, o uso da

NPDL evita o desenvolvimento de compiladores para tratamento das expressões da álgebra de processos. A NPDL está baseada no conceito de plano de navegação (Ferreira et al., 2006) e foi implementada como uma extensão da linguagem (Structured Query Language) SQL. Essa linguagem proporciona facilidades para o uso da álgebra de processos integrados em diferentes linguagens de programação.

Capítulo 3

Trabalhos Relacionados

3.1 Bancos de dados

Bancos de dados públicos como o GenBank (http://www.ncbi.nlm.nih.gov/), Swiss-Prot (http://www.ebi.ac.uk/swissprot/) e EMBL sequence nucleotide database (http://www.ebi.ac.uk/embl/), entre tantos outros, são fontes preciosas para obtenção de dados biológicos e são amplamente usados pela comunidade científica. Os bons resultados dos bancos de dado públicos têm motivado esforços para ampliar a quantidade de dados disponível seja pela integração de várias desses bancos de dados seja pela criação de bancos de dados especializados.

Esforços para integração de bancos de dados podem ser encontrados em trabalhos como Lee, et al., 2006; Simon, et al., 2006; Sohrab, 2005. De maneira geral, tais pesquisas apresentam ferramentas capazes de acessar os bancos de dados escolhidos, recuperar os dados desejados e integrá-los em um esquema de banco de dados mais amplo. Mesmo o aumento de dados proporcionado por esse tipo de integração não é capaz de suprir a necessidade de dados específicos de um determinado organismo ou doença. Para atender a tal demanda, surgem bancos de dados especializados como: plasmaDB (http://www-lecb.ncifcrf.gov/plasmaDB), microarray (Maurer, et al., 2005), entre outros.

Para o estudo do HIV, existem excelentes bancos de dados sobre seqüências genéticas como: Los Alamos (http://hiv-web.lanl.gov/content/index) e Stanford (http://hivdb.stanford.edu) (Shafer, et al., 1999; Shafer, 2006), que, além de dados sobre seqüências e suas anotações, também oferecem ferramentas para análise dos mesmos.

Porém, tais bancos são especializados em seqüências genéticas e, portanto, não possuem dados referentes ao paciente e seu tratamento. Além disso, não oferecem recursos para que o pesquisador gerencie seus dados durante o desenvolvimento de sua pesquisa. Mesmo as ferramentas para análise de seqüência exigem um tratamento dos resultados gerados para que seja feita a associação entre os dados enviados e os resultados obtidos.

Um esforço, no sentido de organizar dados clínicos e seqüências de DNA, é apresentado pelo HIVBase (http://www.hivbase.com), software comercial que funciona no sistema operacional Windows® da Microsoft, que oferece ao usuário um conjunto de ferramentas de análise e um banco de dados. Porém, por se tratar de um software comercial, apresenta as limitações de usar um formato proprietário em seu desenvolvimento e de ter as atualizações e evoluções dependentes da estratégia comercial da empresa.

O desafio atual se encontra não somente na integração de fontes públicas de dados, mas também na oferta de bancos de dados integrados a ferramentas de análise que possibilitem ao pesquisador o uso de tais recursos em diferentes momentos de seus projetos de pesquisa, seja no planejamento e início da coleta de seus dados, passando pela validação cotidiana dos mesmos até a publicação de seus resultados. Além do apoio ao desenvolvimento dos projetos de pesquisa, ferramentas para análise de dados em larga escala integradas aos bancos de dados podem permitir a validação da qualidade dos dados públicos de forma a aumentar sua confiabilidade e o seu uso. Pois se é possível analisar os dados de maneira simples, o pesquisador pode averiguar sua qualidade e mesmo analisá-los usando uma nova abordagem e novos parâmetros.

3.2 Ferramentas para análise de dados sobre o HIV

3.2.1 Avaliação de contaminação

A identificação de contaminações que podem ocorrer durante a obtenção da seqüência por PCR é importante para garantir a qualidade do dado usado (Kwok e Higuchi 1989). A contaminação acontece quando diferentes amostras de DNA entram em contato, seja contágio de algum material usado no laboratório seja por erro de manipulação das amostras. Ela pode ocorrer em qualquer fase da realização do PCR e em qualquer laboratório, mesmo naqueles que adotam rigorosos processos de prevenção de contaminação (Learn, et al., 1996).

Como conseqüência da contaminação, o laboratório produz seqüências com grande similaridade ou até mesmo idênticas (Schuurman et al., 1999). Em geral, o controle negativo usado no procedimento de PCR não é suficiente para detectar contaminação. Um exemplo famoso é estudo de Frenkel et al. (1998), publicado na revista Science, que mostra como problema de contaminação os resultados apresentados nos trabalhos Bakshi et al. (1995) e Bryson, et al. (1995), indicando reversão de infecção pelo HIV em recém-nascidos.

Alguns artigos (Aslanzadeh, 2004; Hartley e Rashtchian, 1993; Kwok e Higuchi, 1989) apresentam procedimentos de como evitar a contaminação. Porém, até a conclusão deste trabalho, não foi encontrado um programa para identificação automática de contaminação.

3.2.2 Subtipagem de seqüências do HIV

A subtipagem de seqüências do HIV usando programas de análise filogenética como PHYLIP - PHYlogeny Inference Package (http://cmgm.stanford.edu/phylip/), PAUP (http://cmgm.stanford.edu/phylip/) demanda conhecimento sobre filogenia e envolvimento do usuário para avaliação dos resultados gerados. Para tornar esse processo mais simples surgem ferramentas para subtipagem automática de seqüências do HIV, como as apresentadas a seguir:

- NCBI Genotyping Program (http://www.ncbi.nih.gov/projects/genotyping). Nele a seqüência é subtipada com o uso de uma janela deslizante. Cada região limitada pela janela é comparada, usando o programa BLAST, às seqüências de referência do vírus. Assim, a seqüência mais similar à seqüência submetida, define o subtipo da janela. No final, a análise dos subtipos recebidos em todas as janelas define o subtipo da seqüência, seja ele puro ou recombinante.

- RIP - Los Alamos Recombinant Identification Program (http://hivweb.lanl.gov/RIP/RIPsubmit.html). O RIP (Siepel et al., 1995) realiza um alinhamento múltiplo entre as seqüências submetidas e seqüências de referência dos subtipos (A1, A2, B, C, D, F1, F2, G, H, J, K, CRF01). O alinhamento das seqüências é analisado por uma janela deslizante. Em cada janela são comparadas as similaridades das seqüências de subtipo conhecido com a seqüência submetida. Os resultados das duas seqüências mais similares

são comparados quanto à significância estatística, usando o teste Z. Caso a diferença seja significante, o subtipo mais similar define o subtipo do trecho da seqüência limitado pela janela. Caso a diferença não seja significativa, o subtipo da janela em questão é marcado como indefinido. Ao final, os subtipos atribuídos a cada janela são comparados. Caso o subtipo permaneça o mesmo em todas as janelas, ele é atribuído com subtipo da seqüência. Caso contrário, a seqüência é definida como de subtipo recombinante ou não subtipada.

- Stanford HIV Drug Resistance Database (HIVDB) (http://hivdb.Stanford.edu). No HIVDB (Ravela et al., 2003) a seqüência é subtipada pela comparação das regiões Transcriptase Reversa e da Protease da seqüência analisada com genes das mesmas regiões das seqüências de referência dos subtipos puros (A, B, C, D, F, G, H, J, K) e dos subtipos recombinantes CRF01_AE e CRF02_AG. O subtipo é definido baseado na similaridade da seqüência com um dos subtipos de referência.

- European-based Subtype Analyzer Program (STAR). (www.biochem.ucl.ac.uk/bsm/virus_database). O programa STAR (Myers et al., 2005) realiza um alinhamento múltiplo usando as seqüências de subtipos puros (A, B, C, D, F, G, H, J, K) e dos subtipos recombinantes CRF01_AE e CRF02_AG. O STAR utiliza uma matriz de pontuação por posição (PSSMs – Position-Specific Scoring Matrices) para cada subtipo comparado com a seqüência de referência. A freqüência de aminoácidos em cada posição da

seqüência submetida é comparada com freqüência de aminoácidos das seqüências de referência, o subtipo com a maior soma de freqüências idênticas, validados por teste Z, é definido como o subtipo da seqüência submetida à análise. Ele trata recombinantes, avaliando o desvio padrão da freqüência das similaridades das seqüências (Myers et al., 2005).

- REGA HIV-1 subtyping tool (http://www.bioafrica.net/subtypetool/html). O REGA (Oliveira et al., 2005) faz um alinhamento das seqüências a serem subtipadas com um conjunto de seqüências de referência cujos subtipos são conhecidos. Em seguida, esse alinhamento é utilizado para a construção de árvores filogenéticas. Seqüências que se agrupam com as seqüências de subtipos puros são consideradas como de subtipo puro e as seqüências aparecem em ramos separados ou entre subtipos que são consideradas como recombinantes ou não classificadas. A classificação das seqüências é aferida usando métodos de Bootstrap (Bradley, 1979). O Bootstrap é um método estatístico que considera a amostra obtida como sendo a população total e passa a analisá-la retirando amostras da amostra original e comparando os resultados obtidos. Este processo simula o ocorrido na maioria dos exemplos reais, onde a totalidade da população não pode ser avaliada e somente alguma amostra podem ser avaliada, ou seja, as amostras são testadas estatisticamente pela avaliação de várias amostras retiradas do conjunto de seqüências submetidas e de referência. Essa análise busca confirmar nas amostras menores os resultados obtidos na amostra original. O processo de alinhamento e análise é repetido novamente usando um grupo menor de

seqüências de referência. Em um terceiro passo, as seqüências submetidas são analisadas por uma janela deslizante em busca de recombinação usando métodos de Bootstrap. Como resultado é gerado um relatório com os detalhes de cada árvore e análise realizada.

Os programas disponíveis para subtipagem seguem a linha de comparação de similaridade entre as seqüências. O REGA aborda a criação de árvores filogenéticas e a análise de seus resultados. Uma avaliação da consistência dos algoritmos HIVDB, STAR e REGA HIV-1 subtyping tool foi realizada pelo estudo (Gifford et al., 2006) e mostrou algumas inconsistências entre os três algoritmos. A discrepância entre os algoritmos foi atribuída à análise de subtipos relacionados ou com história evolutiva próxima; como o caso dos subtipos D e B ou no caso de grupos que pertencem a mesmo subtipo dentro da região seqüenciada, como no caso do subtipo A e CRF01_AE. Além disso, a presença de subtipos recombinantes também foi um fator que gerou a discordância entre os três métodos utilizados.

Como a subtipagem tem se tornado padrão para análise de seqüência de HIV, surge a necessidade de programas que não exijam conhecimentos de filogenia para seu uso e que sejam capazes de avaliar grande volume de dados com precisão ou gerando pequeno conjunto de seqüências não subtipadas. No Brasil, laboratórios espalhados por todo o país têm gerado diariamente um grande volume de seqüências de HIV que necessitam ser subtipadas.

3.2.3 Análise de resistência à droga

O nível de resistência do vírus aos anti-retrovirais é uma informação importante para o tratamento do paciente HIV+. Se os vírus forem resistentes aos medicamentos usados no tratamento, o paciente não apresentará melhora em seu quadro clínico, tendendo a agravá-lo (Wensing e Boucher, 2003).

Além da séria questão de saúde, ainda existe a questão econômica. Uma vez que os recursos financeiros são gastos com medicamentos que não produzem o efeito esperado e que expõem o paciente a doenças oportunistas que também geram despesas para o seu tratamento. Esse breve quadro mostra a importância do tema e ajuda a entender a busca de diferentes grupos de pesquisa por métodos e algoritmos para interpretação de resistência.

Atualmente, os algoritmos para avaliação de resistência às drogas usadas no tratamento de pacientes HIV+ mais usados e com acesso público pela Internet são: HIVdb (Ravela et al., 2003) criado pela Universidade de Stanford - EUA; ANRS (Meynard, et al., 2002) desenvolvido pela Agence Nationale de Recherches Sur le Sida; Rega (Van Laethem, et al.,2002) desenvolvido pelo Rega Institute e o Algoritmo Brasileiro para Interpretação do Teste de Genotipagem de HIV – (http://www.aids.gov.br/genotipagem), mantido pelo comitê de especialistas do projeto RENAGENO do Ministério da Saúde do Brasil e usado como parte do tratamento de HIV/AIDS oferecido pelo Ministério da Saúde.

Esses algoritmos são compostos de regras que expressam a relação entre mutação, droga e nível de resistência. Em suas regras, as mutações são representadas por letras e números. Por exemplo, a mutação 41L, em que o número 41 representa a posição do genoma onde ocorreu a mutação, e a letra L, que representa o aminoácido mutante. Algumas posições possuem mais de um aminoácido associados com resistência à droga. Nesse caso,

os aminoácidos são representados por letras separadas por uma barra (/), como em: 181C/I/L. Assim, o número 181 indica a posição do genoma onde a ocorrência de mutação que resulte nos aminoácidos C, I ou L está relacionada a um determinado nível de resistência. A seguir, são apresentados exemplos de regras que compõem os algoritmos Brasileiro, ANRS e HIVdb.

Regra do Algoritmo Brasileiro para a droga DLV

Presença de 1 ou mais de (100I , 181C/I/L, 188L, 230L, 236L) e ausência da 190A indica resistência.

Tal regra indica que o vírus será considerado como resistente à droga DLV caso estejam presentes no vírus pelo menos uma das seguintes mutações 100I, 181C/I/L, 188L, 230L, 236L em conjunto com a ausência da mutação 190A.

Como exemplo do algoritmo ANRS segue:

Exclude 70R AND Exclude 184VI AND Select Atleast 2 From (41L, 69D, 74V, 215FY, 219QE)

Essa regra indica que a presença de pelo menos duas das mutações 41L, 69D, 74V, 215FY, 219QE, na ausência das mutações 70R e 184V/I classifica o vírus como resistente.

O exemplo a seguir aborda parte de uma das regras do algoritmo HIVdb para a droga AZT.

... 116Y =>10,

151L => 20,...

A primeira linha dessa regra indica que a presença da mutação 116Y atribui uma penalidade de 10 pontos para a soma dos pontos de resistência ao AZT. De maneira semelhante, a segunda linha mostra que a mutação 151L atribui uma penalidade de 20

pontos. Ao final da avaliação da lista de mutações do vírus, a pontuação obtida por cada droga é classificada conforme mostra a Tabela 3.1.

Pontuação	Classificação
- infinito ate 10	Susceptível
10 a 15	Potencial baixo nível de resistência
15 a 30	Baixo nível de resistência
30 a 60	Resistência intermediária
60 até + infinito	Alto nível de resistência

Tabela 3.1: Tabela para classificação do algoritmo HIVdb dos níveis de resistência de acordo com a pontuação obtida por cada droga.

Como visto, os quatro algoritmos citados possuem estrutura similar, baseada em regras que avaliam as mutações encontradas no vírus e geram relatórios de níveis de resistência para cada droga analisada. Como diferença, os algoritmos *ANRS, Rega* e o *Algoritmo Brasileiro* atribuem três níveis de resistência às drogas analisadas. São eles:

1. Suscetível – Sugere que não foram encontradas mutações que conferem resistência à droga analisada;
2. Intermediário – Indicativo que as mutações encontradas afetam parcialmente o funcionamento da droga no tratamento do paciente;
3. Resistente – Nível que representa o estágio no qual as mutações do vírus impedem a ação da droga, tornando-a ineficiente no combate à infecção pelo vírus.

Já as regras do algoritmo HIVdb classificam a resistência em cinco níveis, conforme a Tabela 3.1.

Outra diferença entre os algoritmos é a interpretação do impacto das mutações em relação à resistência às drogas. Conseqüentemente, tais algoritmos geram resultados

diferentes para o mesmo conjunto de mutações, o que motiva estudos de comparações entre eles, como: Ravela et al., 2003, e Zazzi et al., 2004.

Devido ao constante avanço das pesquisas, que descobrem as relações entre mutações e resistência, os algoritmos necessitam atualização constante para refletir as novas descobertas. Como a atualização de programas é um processo trabalhoso, demorado e sujeito a erros durante sua execução (Sommerville, 2006), surge a necessidade de programas para geração automática desses algoritmos. Uma iniciativa nesse sentido foi o desenvolvimento pela Universidade de Stanford do ASI - Algorithm Specification Interface - Interface para especificação de algoritmos (Betts e Shafer, 2003; Liu e Shafer, 2006) voltado para a especificação das regras dos algoritmos HIVdb, ANRS, Rega e variações desses algoritmos.

No ASI as regras são representadas no formato XML e, em seguida, um compilador utiliza tais regras para a geração do código do algoritmo especificado. A Figura 3.1 mostra um exemplo de uma regra do algoritmo Rega no formato XML, usado pelo programa ASI.

```
<ALGORITHM>
<ALGNAME>Rega v7.1.1</ALGNAME>
  <ALGVERSION>7.1.1</ALGVERSION>
    <DEFINITIONS>
      <LEVEL_DEFINITION>
        <ORDER>1</ORDER>
        <ORIGINAL>Susceptible GSS 1</ORIGINAL>
        <SIR>S</SIR>
      </LEVEL_DEFINITION>
        ...
      <LEVEL_DEFINITION>
        <ORDER>6</ORDER>
        <ORIGINAL>Resistant GSS 0</ORIGINAL>
        <SIR>R</SIR>
      </LEVEL_DEFINITION> <RULE>
        ...
    </DEFINITIONS>
          ...
  <DRUG>
   <NAME>AZT</NAME>
      <RULE>
        <CONDITION>
          SELECT ATLEAST 1 FROM (151M,69i)
        </CONDITION>
        <ACTIONS>
          <LEVEL>6</LEVEL>
        </ACTIONS>
          ...
      </RULE>
      ...
</ALGORITHM>
```

Figura 3.1: Exemplo de uma regra de resistência no formato XML usado pelo programa ASI (Betts e Shafer, 2003).

As linhas iniciais apresentam o nome do algoritmo e sua versão. Em seguida, são definidos os níveis de resistência e as regras referentes a cada uma das drogas analisadas avaliadas pelo algoritmo. A regra é formada pelo nome da droga, conjunto de mutações avaliado e o nível de resistência conferido por ela. Por sua vez, o conjunto de mutações é representado por comandos como: SELECT AT LEAST 1 FROM (151M, 69i), onde é indicada a ação a ser executada e o conjunto de mutações avaliado pela regra. Nesse exemplo, o comando indica que deve ser verificada a presença de pelo menos uma mutação das listadas entre parênteses. A estrutura da regra permanece a mesma para as demais regras, mudando apenas a quantidade de mutações selecionada e o conjunto de mutação. A representação das regras dos algoritmos ANRS e Rega é feita de forma similar à apresentada na Figura 3.1.

Apesar da flexibilidade oferecida pelo uso de arquivos no formato XML para a definição de regras, essa é uma abordagem informal e focada nas regras do algoritmo. Pois, o arquivo XML descreve as regras usadas e os respectivos níveis/pontuação de resistência, mas não descreve como o sistema deve funcionar. Ou seja, o sistema pode ser gerado de maneira automática, para os casos de inclusão e/ou exclusão de alguma mutação ou mudança no seu peso sobre a resistência. Porém, o algoritmo necessitará de alterações, além das mudanças no arquivo XML, caso o funcionamento do algoritmo seja modificado. Por exemplo, se novos parâmetros forem avaliados, como: subtipos e drogas usadas no tratamento do paciente. Nesse sentido, alguns trabalhos (Soares et al., 2007; Shafer, 2006; Wainberg, 2004; Zazzi et al., 2004; Wensing e Boucher, 2003) sugerem que as pesquisas e programas para a análise de resistência às drogas podem ser aprimorados com o uso de mais informações, além das mutações encontradas no vírus, como: seu subtipo e dados do tratamento do paciente.

Capítulo 4

DBCollHIV

4.1 Banco de dados

O banco de dados do DBCollHIV foi projetado para atender aos requisitos de integração de dados sobre o tratamento de pacientes HIV positivo e as respectivas análises necessárias. Devido às constantes mudanças nas necessidades de dados que caracteriza a área de pesquisa, o DBCollHIV foi projetado de forma modular para permitir que novos módulos sejam inseridos e que módulos existentes possam ser alterados e até mesmos removidos sempre que necessário. Para tanto, os módulos do DBCollHIV possuem baixa coesão entre eles o que torna mais simples a inclusão e exclusão de módulos. A Figura 4.1, apresenta o modelo do DBCollHIV e permite avaliar o acoplamento entre seus módulos. No canto superior direito está o módulo Paciente (Patient), principal módulo de coleta de dados do DBCollHIV.

Para manter os módulos com baixo acoplamento e conseqüente flexibilidade para alteração, eles devem possuir apenas ligações com o módulo de cadastro de paciente (Patient module). Porém, por motivo de eficiência e praticidade, alguns módulos podem ter ligações também entre eles. Nesse caso, os dados que compõem o relacionamento entre os módulos não são obrigatórios. Ou seja, o dado pode ou não ser cadastrado no banco de dados. Assim, caso um dos módulos necessite ser removido, existirá um baixo impacto no sistema como um todo. Como exemplo, o módulo de exames (Exam module) se relaciona também com o módulo de cadastro de amostras (Sample module). Tal relacionamento permite armazenar

informações das amostras usadas nos exames. Essa é uma informação importante, porém nem todos os projetos possuem dados sobre a amostra usada no exame. Nesse caso, não existe a obrigatoriedade de cadastro de dados da amostra. O mesmo ocorre com os relacionamentos de outros módulos como: seqüência e amostras que permitem armazenar dados da amostra da qual a seqüência foi obtida.

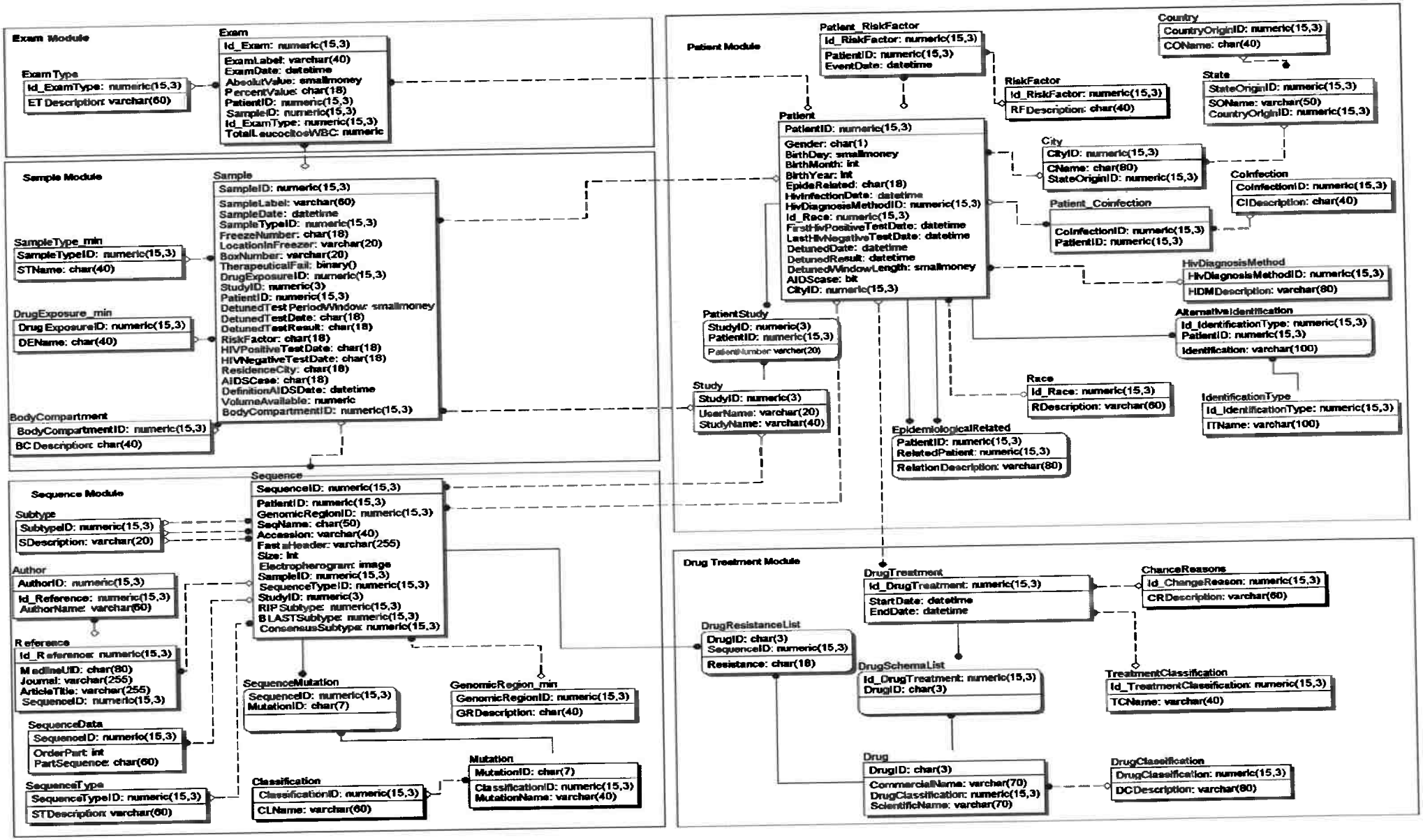

Figura 4.1: Modelo do DBCollHIV, com destaque para seus de módulos e relacionamentos.

Além da estrutura de dados modular, outra vantagem do DBCollHIV é a automação de tarefas de análise necessárias aos estudos de dados clínicos e moleculares. Durante o processo de análise de dados, o arquivo gerado por um passo pode ser usado pelos próximos passos mesmo que possua alguma incompatibilidade estrutural, o que é resolvido pela tradução de um arquivo XML (Achard et al., 2001). A integração proporcionada pelo DBCollHIV é baseada abordagem GEnflow (Oikawa et al., 2004). Nessa abordagem, a aplicação P_i (i= 1,2,...n; onde n é o número de aplicações instaladas e disponíveis para o sistema em questão) são colocadas em ordem seqüencial de acordo com suas tarefas. Essa propriedade é definida quando P_i é instalada no ambiente e de acordo com seus parâmetros de execução. Isso define a relação de precedência entre as aplicações.

Cada aplicação P_i está associada com um passo de execução E_k (k=1,2, ..., m; onde *m* é o número de tarefas de um workflow em particular). As cadeias de execução são construídas em série e são executadas utilizando arquivos de entrada e saída.

Todas as aplicações seguem a ordem de precedência e a compatibilidade semântica de seus arquivos de entrada e saída; elas são chamadas *cadeias*. A Figura 4.2, apresenta alguns tipos de cadeias de aplicações como: P_1 e P_3, P_2 e P_4, P_3 e P_4.

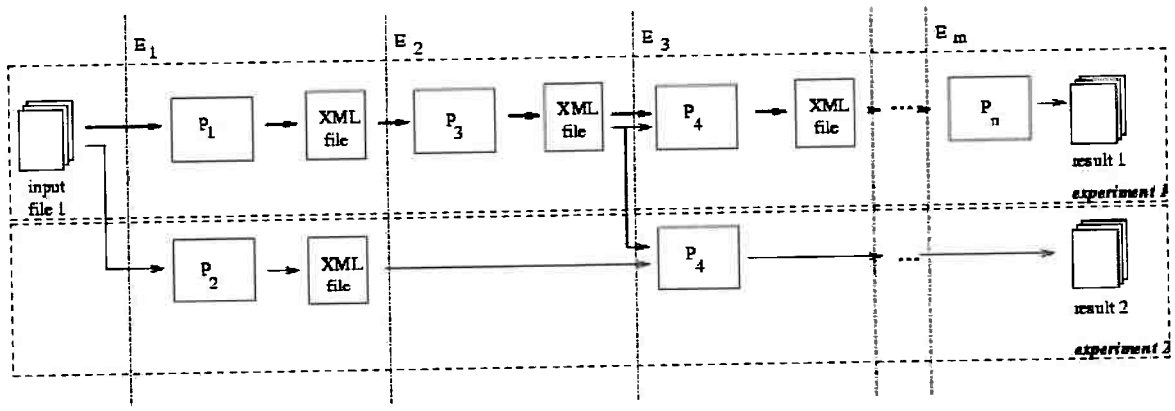

Figura 4.2: Esquema de integração usado pelo DBCollHIV.

4.1.1 Interface para cadastro de dados

Nessa seção são apresentados os conjuntos de dados coletados em cada um dos módulos do DBCollHIV e suas respectivas interfaces para cadastro de dados. Atualmente o DBCollHIV conta com 5 módulos:

1. Cadastro de Paciente – Módulo para coleta de dados relacionados ao paciente. Como requisito de segurança, não são coletados dados que permitam a identificação do paciente como: nome, endereço, e-mail, telefone e nome de parentes. A Figura 4.3 apresenta a tela de cadastro dos pacientes, cujos campos são apresentados a seguir:
   - Identity : Código do paciente no estudo;
   - Alternative Identification: Permite armazenar códigos de identificação atribuídos ao paciente em outros estudos que ele tenha participado;
   - Gender: Sexo;
   - Birth Location: Coleta o país, a região ou o estado de nascimento do paciente;
   - Residence Location: Coleta o país, a região ou o estado de residência do paciente;
   - Detuned test date: Data de realização do teste de Detuned. Teste que permite avaliar se a infecção é recente;
   - Detuned Result: Resultado do teste de Detuned;
   - Detuned test window length : Tamanho da janela usada no teste de Detuned;
   - AIDS Case: Indica se o paciente foi considerado como um caso de AIDS;
   - Coinfections: Coleta todas as co-infecções manifestadas no paciente.

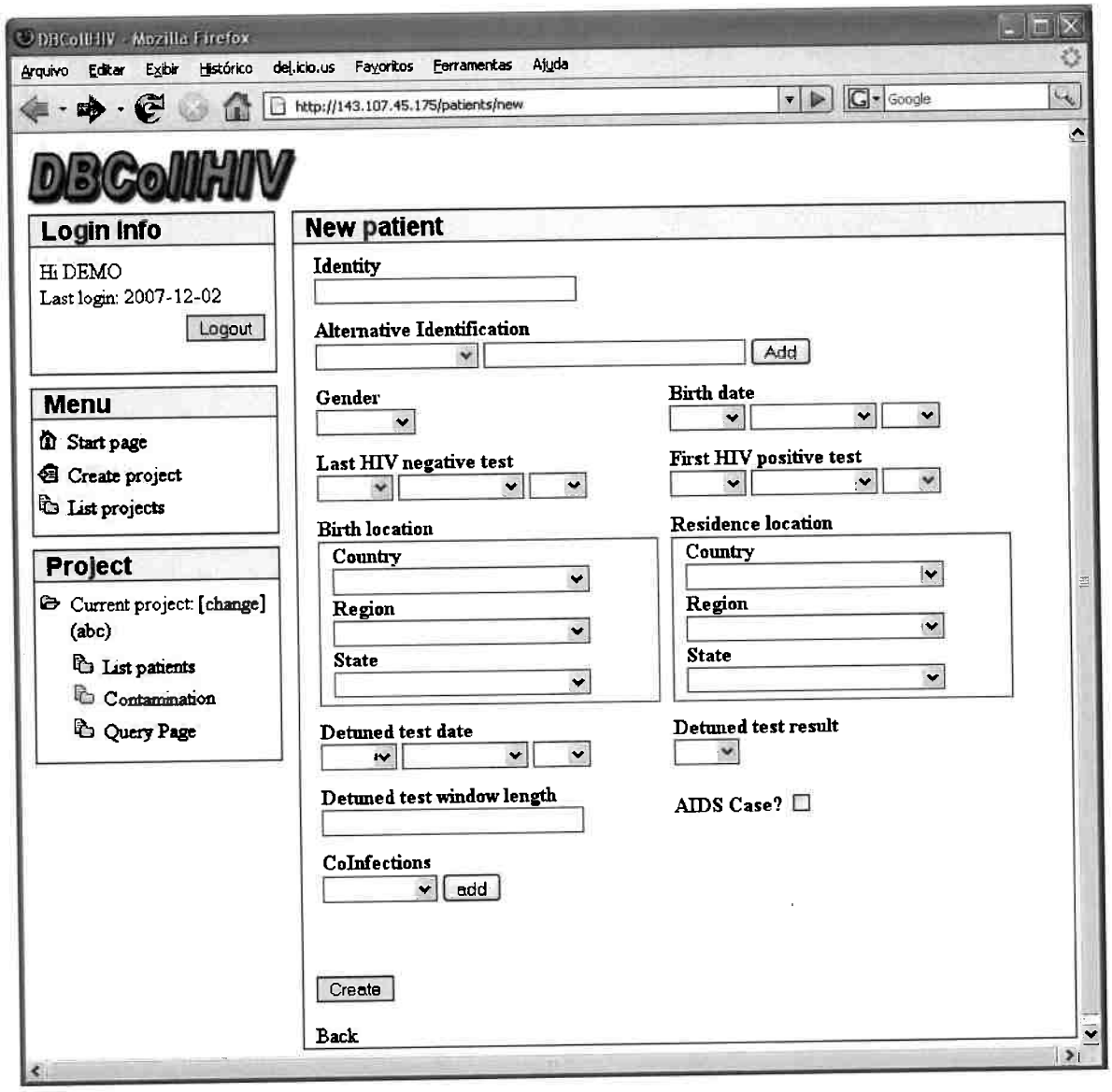

Figura 4.3: Tela para cadastro de pacientes.

2. Cadastro de amostras – Permite controlar informações sobre amostras usadas para realização de exames, incluindo informações sobre o armazenamento da amostra nos congeladores. A Figura 4.4 mostra a tela de cadastro de amostras. Nesse cadastro estão coletados os seguintes dados:
   - Label : Identificação da amostra;
   - Date: Data da coleta da amostra;
   - Freezer Number: Identificação do freezer onde está armazenada a amostra;

- Location in freezer: Em qual prateleira do freezer se encontra a amostra;
- Box number: Identificação do caixa onde a amostra esta guardada;
- Volume available: Indica a quantidade de amostra disponível;
- Sample type: Indica o tipo da amostra. Ex.: Plasma, Cultura, etc.;
- Body compartment: Indica de onde foi tirada a amostra. Ex.: Sangue, saliva, pele, etc.;
- Drug exposure: Armazena a exposição do paciente ao tratamento Ex.:
    - Uso de droga anti-retrovirais no momento do tratamento;
    - Ex.: Virgem de tratamento, Em tratamento, Já recebeu ;
    - Tratamento previamente.

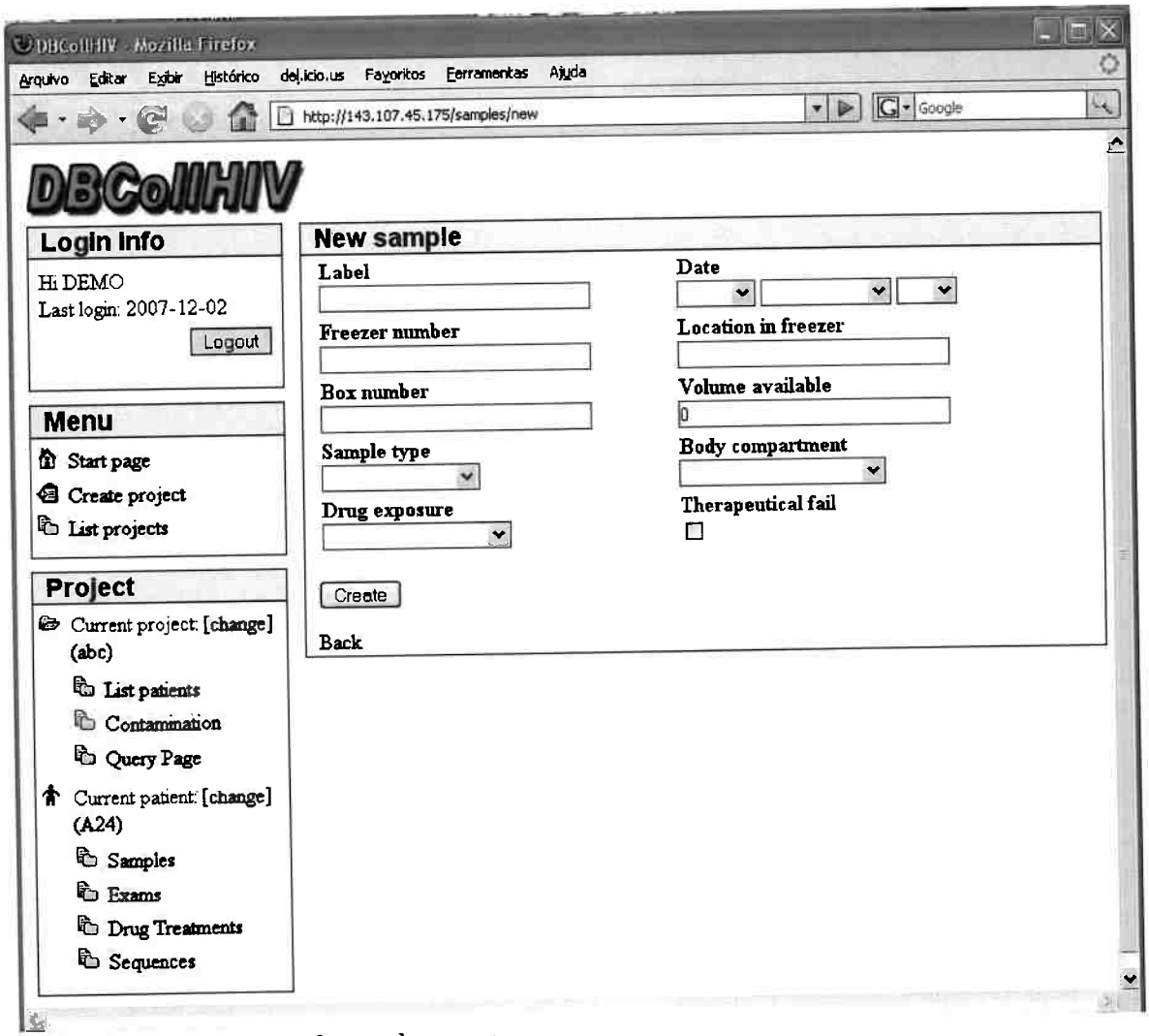

Figura 4.4: Tela para cadastro de amostras.

3. Cadastro de Exames – Permite o cadastramento dos valores de resultados de exames de CD3, CD4, CD8 e de carga viral. Os exames são cadastrados em duas telas. A tela de exames de CD3, CD4, CD8, Figura 4.5, possui os seguintes campos:
    - Label – Identificação do exame;
    - Exam Date – Data de realização do exame;
    - Leucocytes WBC – Contagem total de leucócitos;
    - CD3 absolute value – Valor absoluto de células T-CD3;
    - CD3 percent value – Percentual de células T-CD3;
    - CD4 absolute value – Valor absoluto de células T-CD4;
    - CD4 percent value – Percentual de células T-CD4;
    - CD8 absolute value – Valor absoluto de células T-CD8;
    - CD8 percent value – Percentual de células T-CD8.

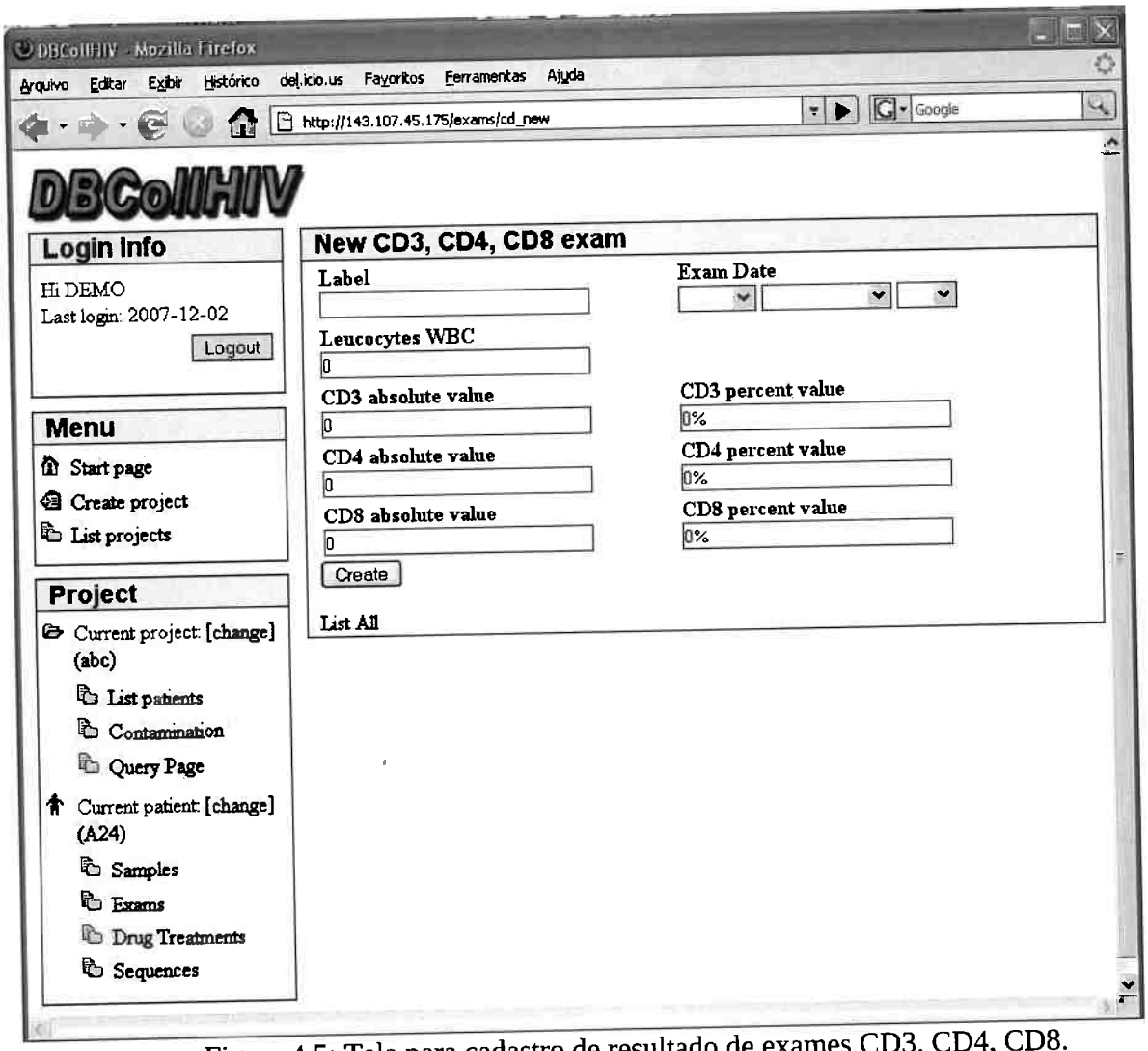

Figura 4.5: Tela para cadastro de resultado de exames CD3, CD4, CD8.

4. A tela de cadastro dos exames de Carga Viral, Figura 4.6, permite a coleta dos seguintes dados:
   - Label – Identificação do exame;
   - Exam Date – Data de realização do exame;
   - Absolute value – Valor absoluto de carga viral3;
   - Absolute value (logarithm scale)– Valor absoluto em escala logarítmica.

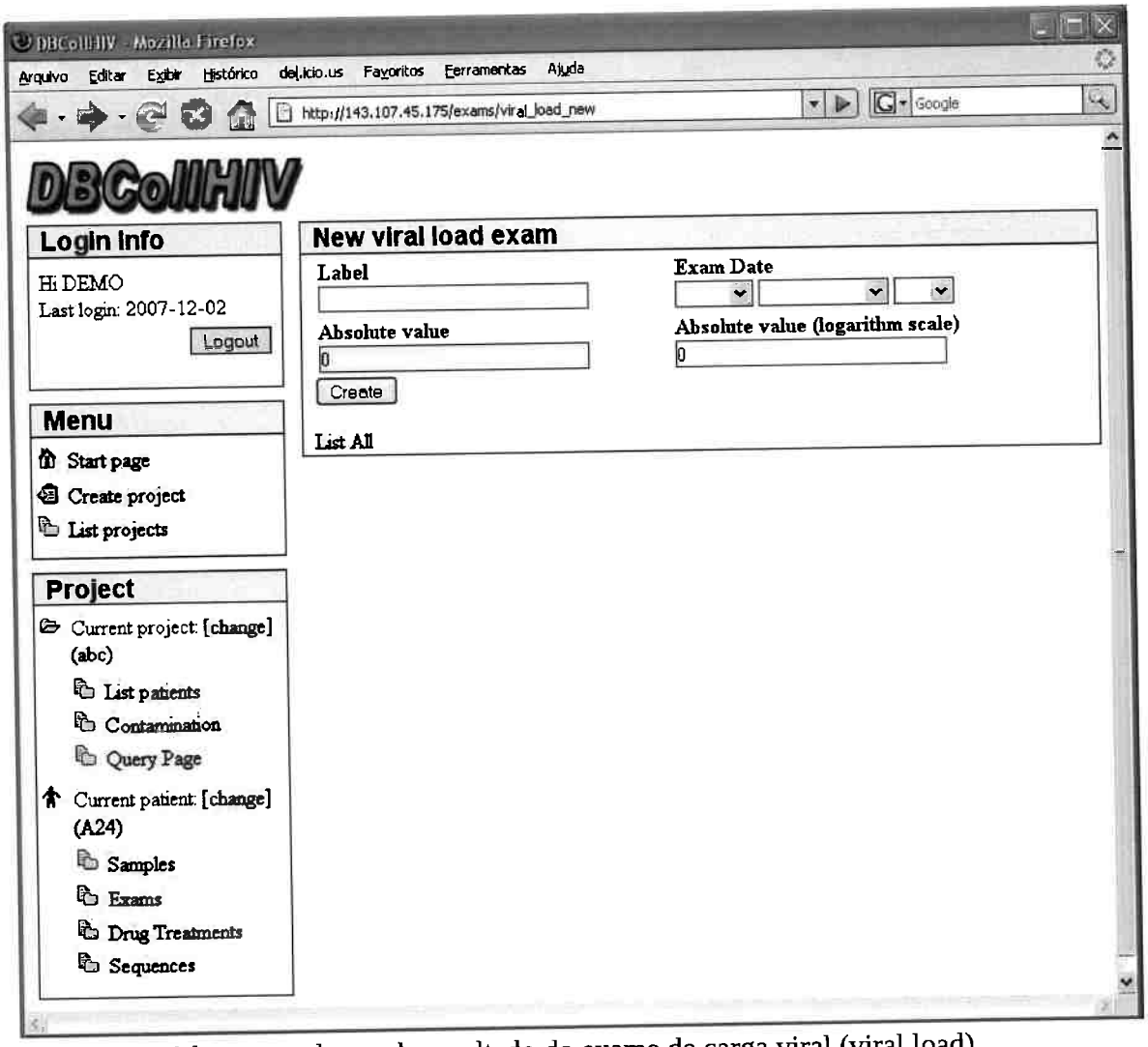

Figura 4.6: Tela para cadastro de resultado do exame de carga viral (viral load).

5. Cadastro de seqüências – O cadastro de seqüências apresenta os seguintes campos, como mostra a Figura 4.7.
    - Sample – Identifica a mostra de onde foi extraída a seqüência, preenchimento opcional;
    - Name – Nome atribuído à seqüência;
    - Tipo – Tipo da seqüência;
    - Genome Region – Região do genoma a qual pertence a seqüência;
    - Accession – Número de acesso, se a seqüência foi depositada no GenBank;
    - Subtype Blast – Subtipo definido por Blast;
    - Subtype Consensus – Subtipo consenso;
    - Fasta size – Tamanho da seqüência no arquivo fasta;
    - Fasta – Permite colocar a seqüência em formato Fasta. A seqüência também pode ser inserida no banco de dados usando a opção *Upload Fasta* que permite carregar a seqüência de um arquivo;
    - Reverse Transcriptase mutations – Cadastro das mutações encontradas na transcriptase reversa;
    - Protease mutations – Cadastro das mutações encontradas na protease.

Os demais campos do módulo de seqüência são obtidos a partir da análise da seqüência e de suas mutações. Ou seja, são dados secundários gerados por programas de análise e que também são armazenados no DBCollHIV. Tais campos são tratados na seção referente às ferramentas de análise de dados.

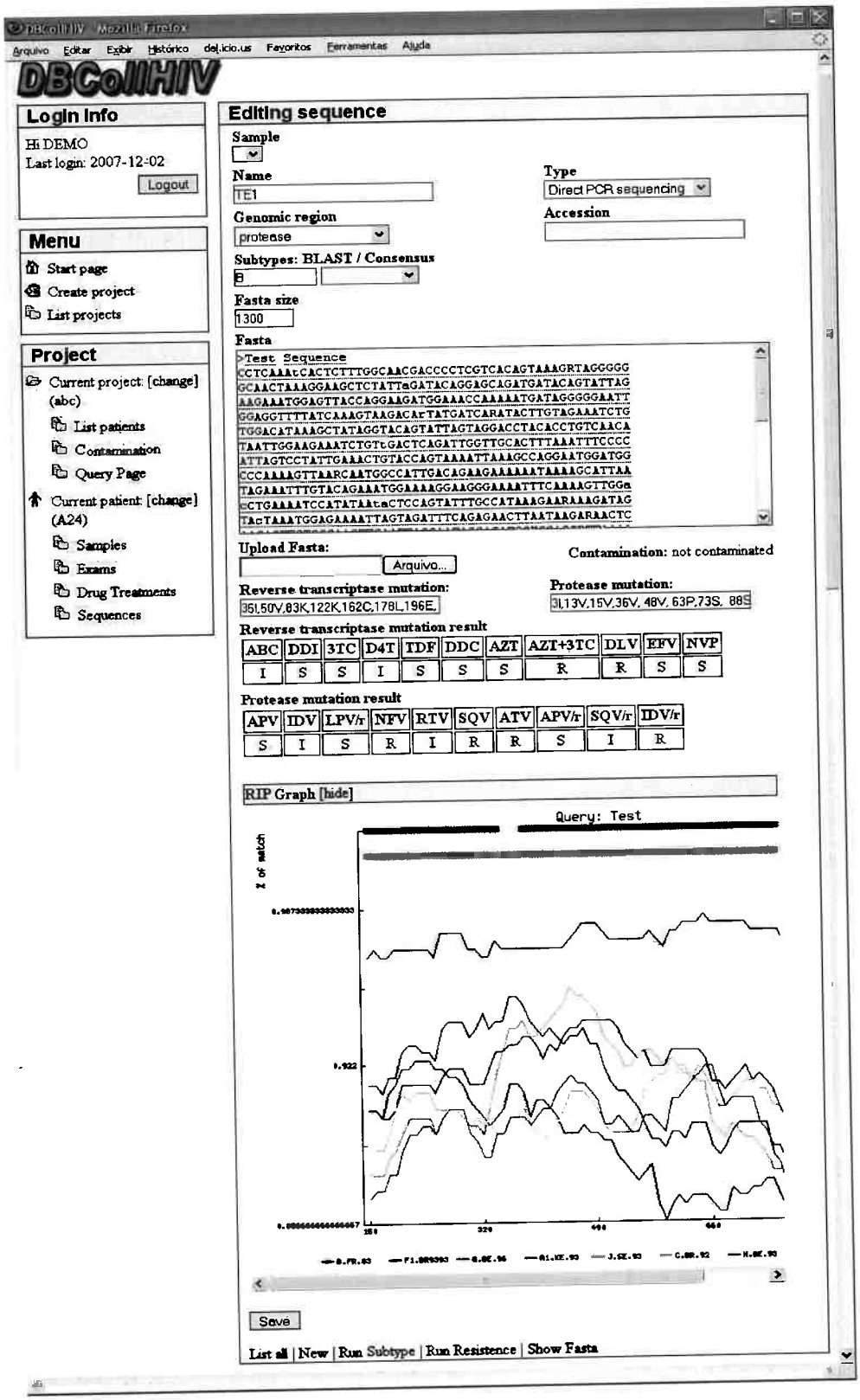

Figura 4.7: Tela para cadastro de seqüências.

6. Cadastro do Histórico de tratamento do paciente – Nesse módulo são cadastrados dados sobre as drogas usadas pelo paciente durante seu tratamento. São coletados os seguintes dados, como mostra a Figura 4.8.
   - Start Date – Data de início do tratamento;
   - End Date – Data do final do tratamento;
   - Reason for treatment change – Razão para a mudança do tratamento. Ex.: Efeitos colaterais, Novo medicamento, falha terapêutica;
   - Drugs – Lista de drogas usadas durante o período informado.

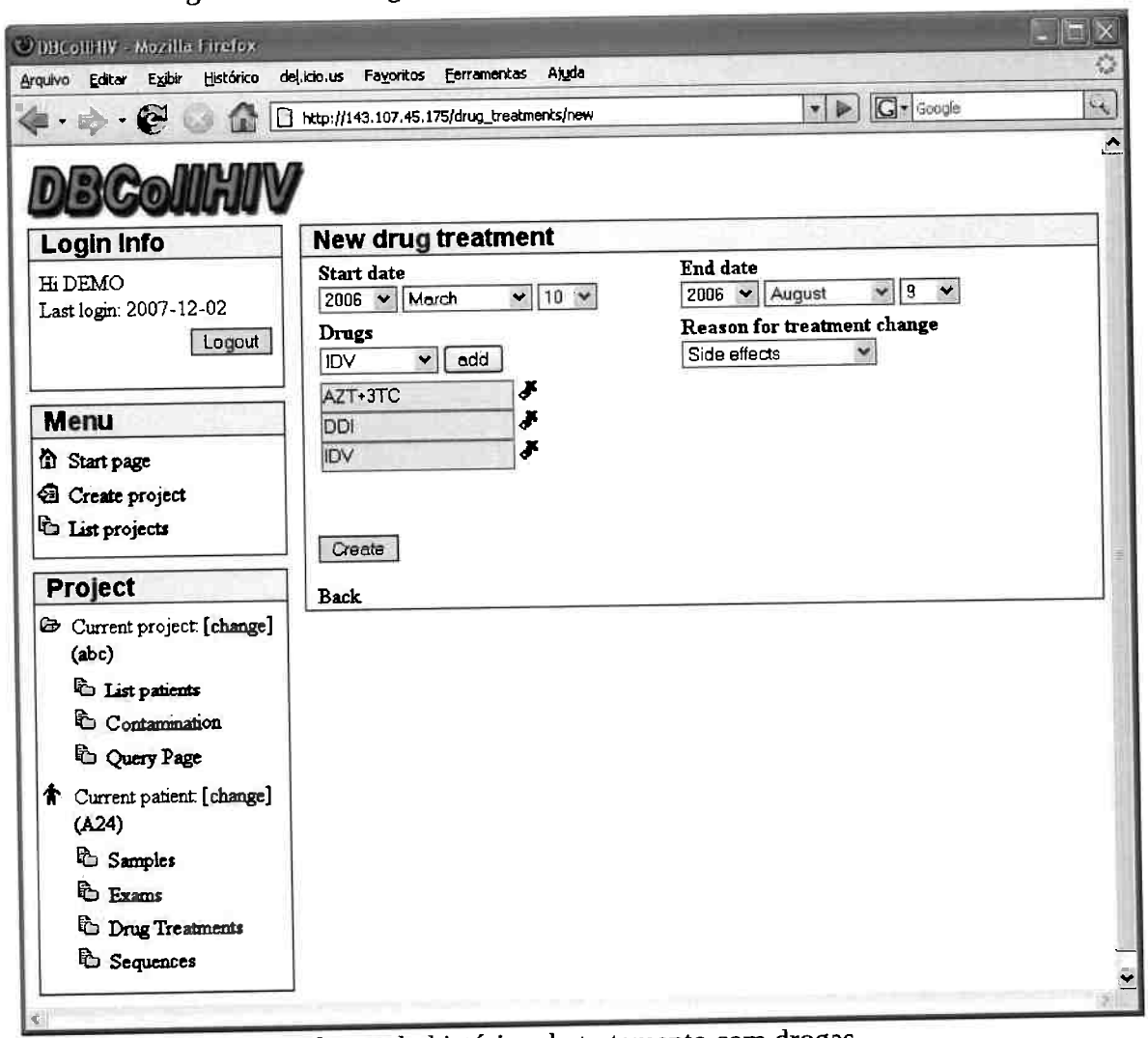

Figura 4.8: Tela para cadastro do histórico do tratamento com drogas.

A Figura 4.9 apresenta a tela de consulta aos dados armazenados no DBCollHIV, chamada de *query page*. Nela o usuário pode selecionar os dados que deseja visualizar e os

dados que deseja usar como condição de consulta. Os dados a serem visualizados ou exportados são definidos marcando o quadrado de seleção ao lado de cada nome de campo, quando são informados/selecionados valores para os campos. Tais valores são utilizados para definir condições para recuperação dos dados. Por exemplo: o usuário pode selecionar os projetos que deseja consultar, escolhe visualizar os campos sexo, país de origem, resultado do test detuned, amostra de pacientes virgens de tratamento e de determinado subtipo.

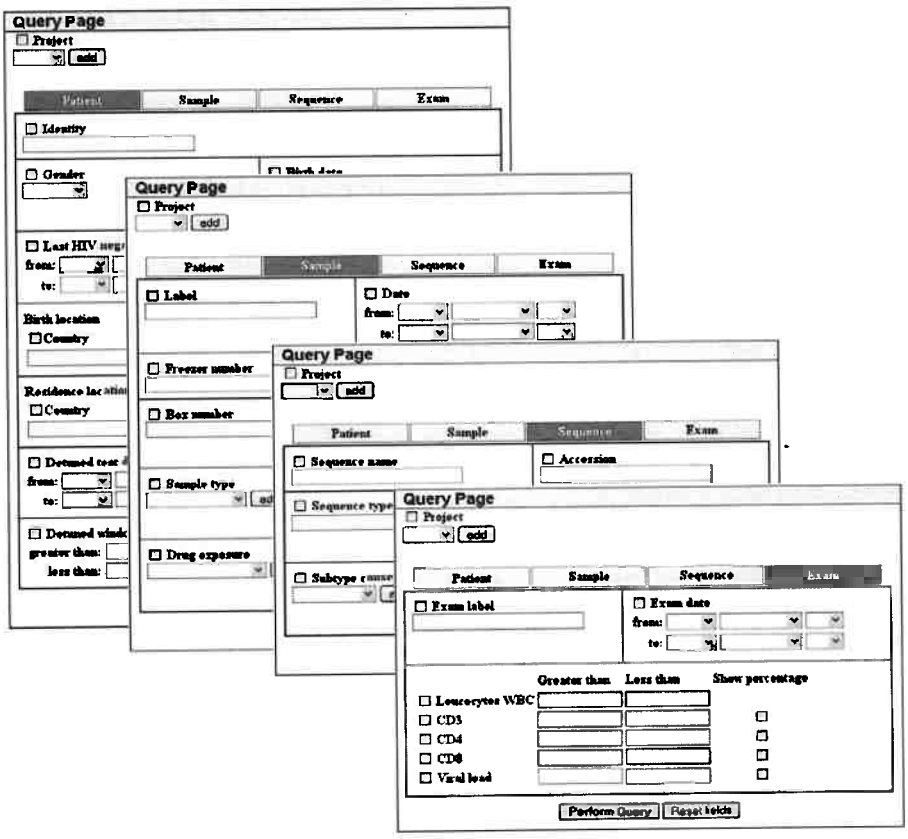

Figura 4.9: Tela para consulta de dados do DBCollHIV.

A consulta é gerada automaticamente e permite que o usuário receba os resultados na tela, exporte os dados para arquivos no formato XML, texto e FASTA para o caso de seqüências, como mostra a Figura 4.10.

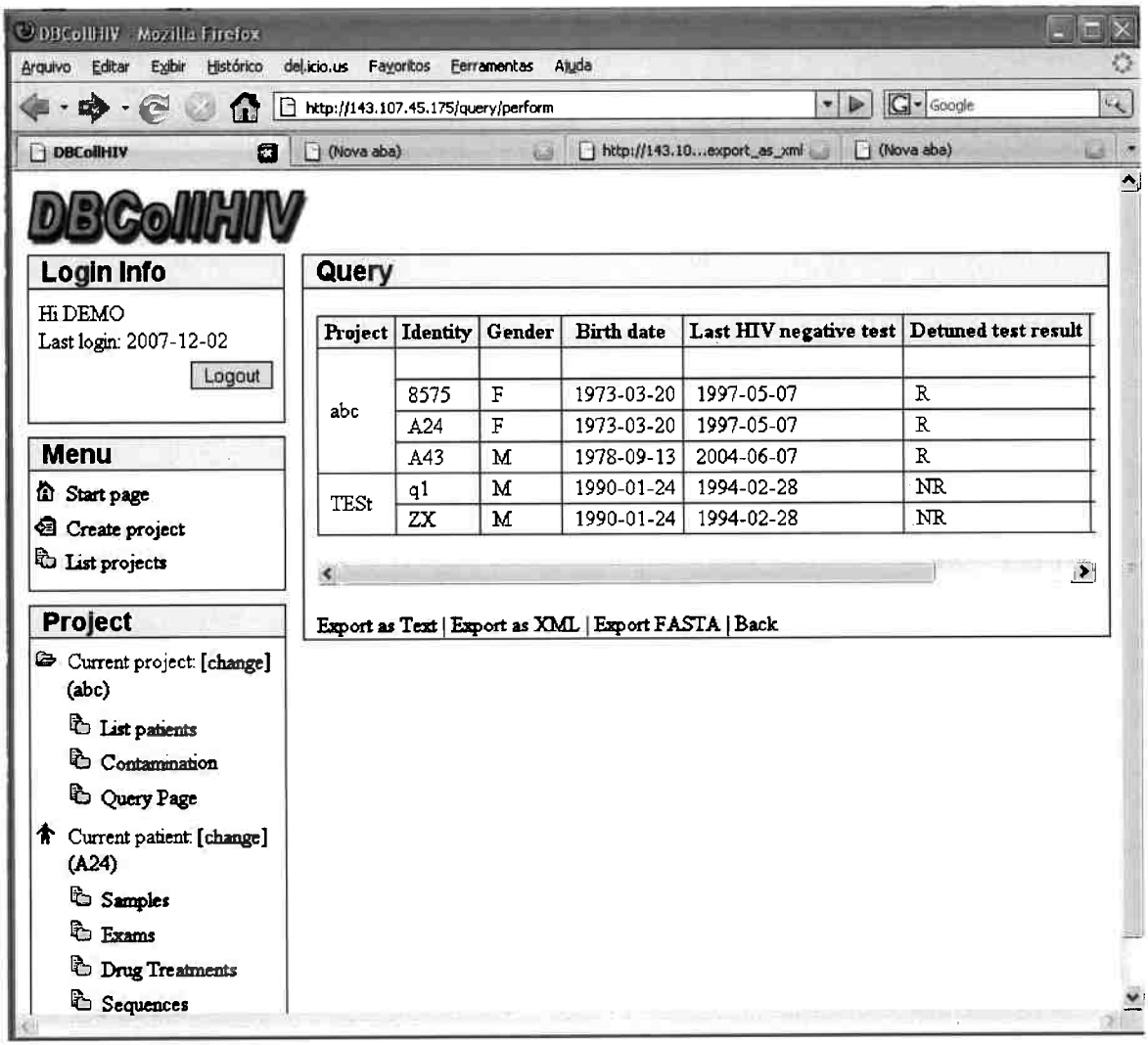

Figura 4.10: Tela para exportação de dados.

Como parte de suas funcionalidades, o DBCollHIV prevê a possibilidade de cada usuário cadastrar diferentes projetos e compartilhá-los da maneira que desejar. A busca por cooperação entre pesquisadores prevista na concepção do DBCollHIV permite ao usuário definir seus projetos e permitir que um grupo de usuário, escolhidos por ele, tenha acesso aos seus dados. O proprietário dos dados é o responsável pela liberação dos dados quando desejar. A Figura 4.11 mostra as opções de cadastro de projetos. Os campos disponíveis para configuração do projeto são:

- Name – Nome do projeto;
- Default Identity – Define o tipo de identificação será usado como padrão pelo Sistema. Ex.: Id;
- Include another user for this project – e-mail de usuários cadastrados que terão acesso aos dados;
- Country – Lista os países disponíveis para o projeto. Evita a apresentação de todos os nomes de países;
- Public information – Informação sobre o projeto que pode ser vista por todos os usuários;
- Private information – Informação disponível para os usuários do projeto.

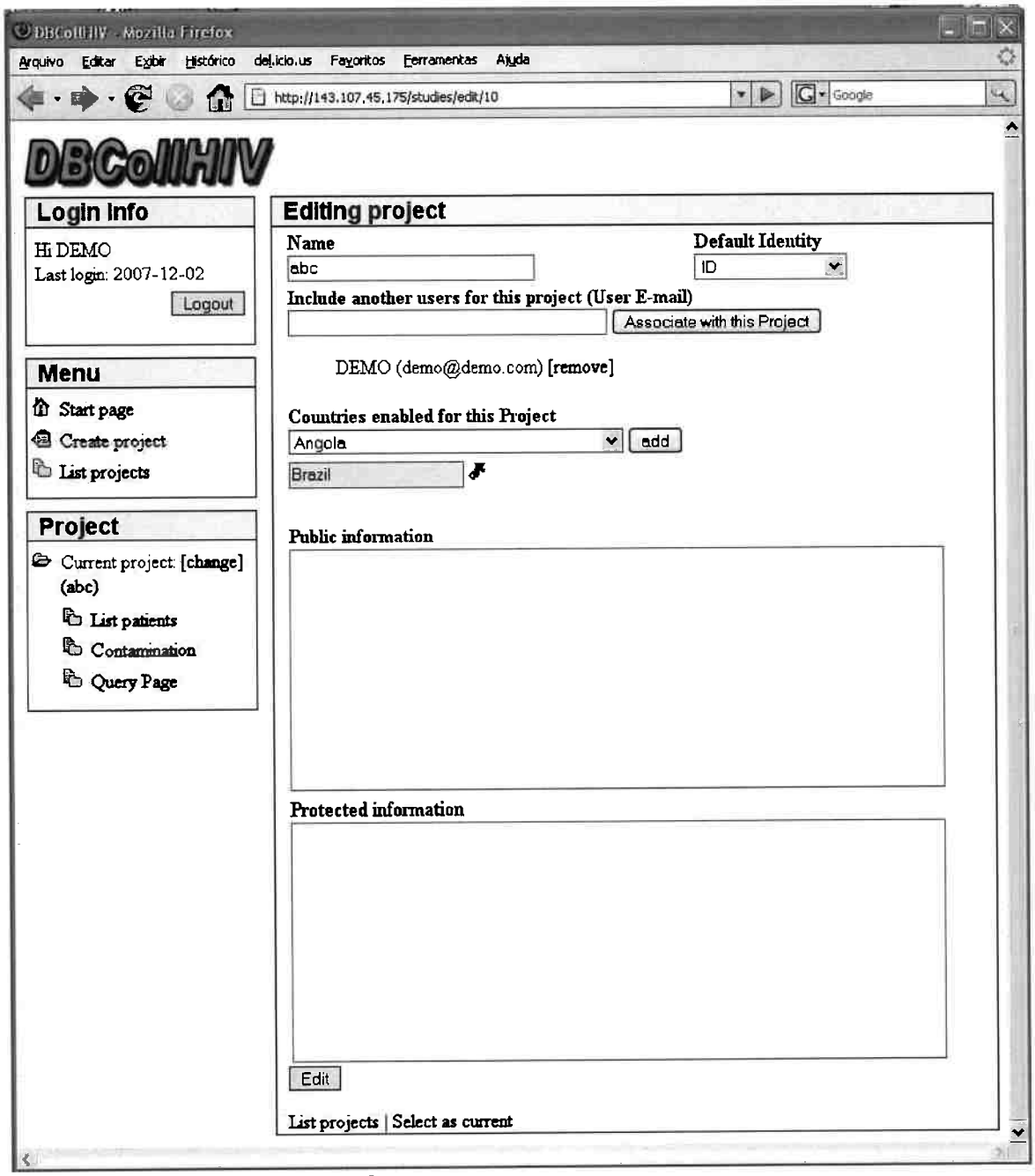

Figura 4.11: Tela para cadastro de projetos.

4.2 Programas para análise de dados

As ferramentas de análise disponíveis no DBCollHIV foram desenvolvidas para contemplar os seguintes requisitos: oferecer análises iniciais que compõem o ciclo de estudos do DNA do HIV, possuir capacidade de processar grande volume dados e diminuir a necessidade de interação entre o usuário e o sistema. Dessa forma, os dados armazenados no banco de dados podem ser analisados sempre que desejado ao custo de processamento de máquina, sem demandar esforço do usuário. Além disso, o desenvolvimento modular das ferramentas favorece a possibilidade de reutilização de código, como no caso de alinhamento de seqüências, presente em diferentes tipos de análises e que pode ser executada por um código único. As ferramentas de análise também podem ser utilizadas sem a necessidade de inserir os dados no DBCollHIV. Para isso, estão disponíveis páginas HTML específicas para cada ferramenta, nas quais os dados são enviados e analisados sem armazenamento de dados.

A escolha das ferramentas de análise foi motivada por necessidades encontradas em projetos do Ministério da Saúde do Brasil, como o RENAGENO, e para as quais não existiam ferramentas para atendê-las ou cujo uso não favorecia análises em larga escala. Foram desenvolvidos programas para o controle de contaminação das seqüências produzidas pelos laboratórios, a subtipagem de seqüências, a análise de mutações para identificação de resistência às drogas usadas no tratamento e a geração automática do software de identificação de resistência à droga e simulação do impacto de novas regras no quadro de pacientes brasileiros.

4.2.1 Controle de contaminação de seqüências obtidas por PCR

Avaliar a qualidade dos dados é uma necessidade constante em qualquer processo de análise, uma vez que dados sem qualidade comprometem os resultados obtidos. Junto com a popularização das técnicas de obtenção de seqüências genéticas como o PCR, cresce o volume de dados de seqüências genéticas disponíveis para análise e a necessidade de ferramentas tanto para analisá-los como para validá-los quanto à qualidade. Um dos

problemas relacionados à qualidade dos dados sobre seqüência genética é a contaminação das amostras que pode ocorrer durante seu processamento em laboratório.

Com o propósito de identificar possíveis contaminações de PCR, foi criado o software *PCR Contamination*. Sua função é analisar seqüências produzidas por um mesmo laboratório em busca de possíveis contaminações, dentro de um período de tempo em que se considera possível a contaminação entre as amostras.

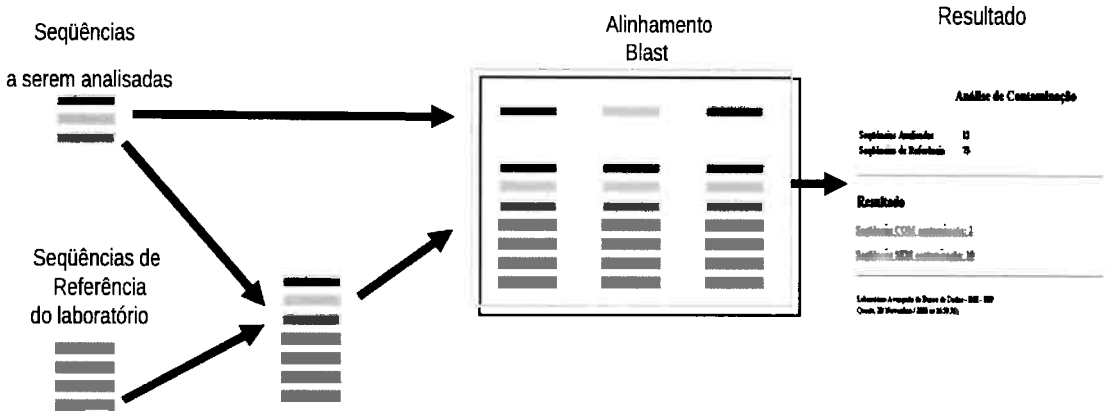

Figura 4.12: Diagrama da análise realizada pelo PCR Contamination.

A Figura 4.12 apresenta o funcionamento do *PCR Contamination*. Inicialmente, as seqüências submetidas para análise são colocadas junto com as seqüências de referência do laboratório para formar o banco de dados de referência do programa BLAST (Altschul, et al., 1990). No passo seguinte, as seqüências a serem analisadas são alinhadas usando o banco de dados de referência do BLAST. Em seguida, as seqüências são comparadas uma a uma entre si em busca das seqüências com possível contaminação. Ou seja, seqüências que apresentam um percentual de similaridade acima do esperado para o HIV. Esse percentual é definido pelo usuário e normalmente se encontra entre 98% a 100% de similaridade. As seqüências não contaminadas passam a formar o conjunto de *seqüências de referência* do laboratório. Assim, as novas seqüências produzidas devem ser comparadas com as seqüências de referência que estejam dentro de um período aceitável para a contaminação das novas seqüências.

A escolha das seqüências a serem usadas como seqüência de referência e a separação dos resultados obtidos tornam a análise de contaminação bastante trabalhosa e repetitiva, justificando a sua automatização para melhorar o desempenho do laboratório e evitar erros. A versão integrada ao DBCollHIV, Figura 4.13, faz a seleção das seqüências

de um mesmo estudo e as analisa automaticamente, registrando o resultado no banco de dados, Figura 4.14. Isso permite a seleção das seqüências de acordo com o rótulo recebido na análise, Figura 4.15.

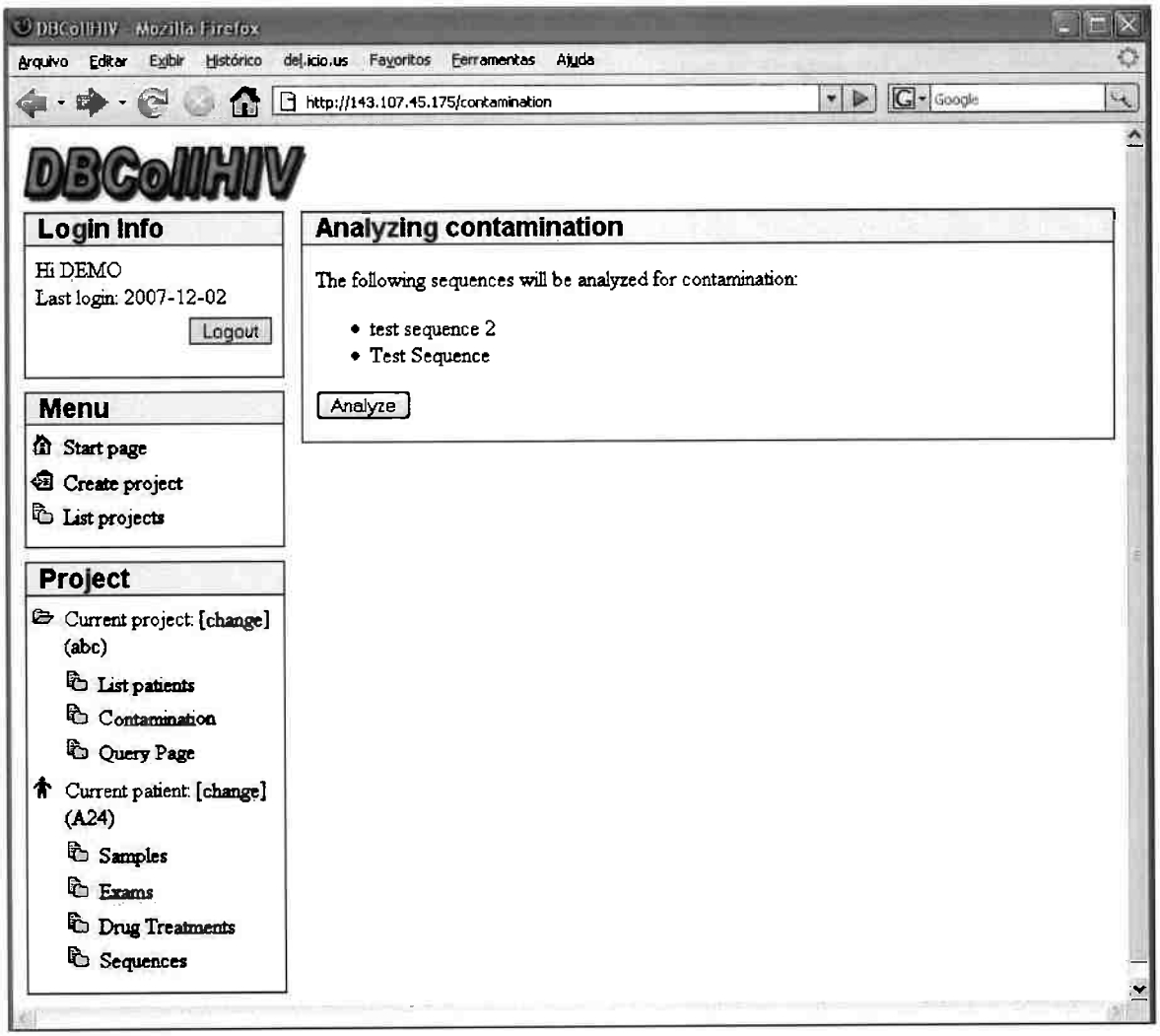

Figura 4.13: Análise de contaminação.

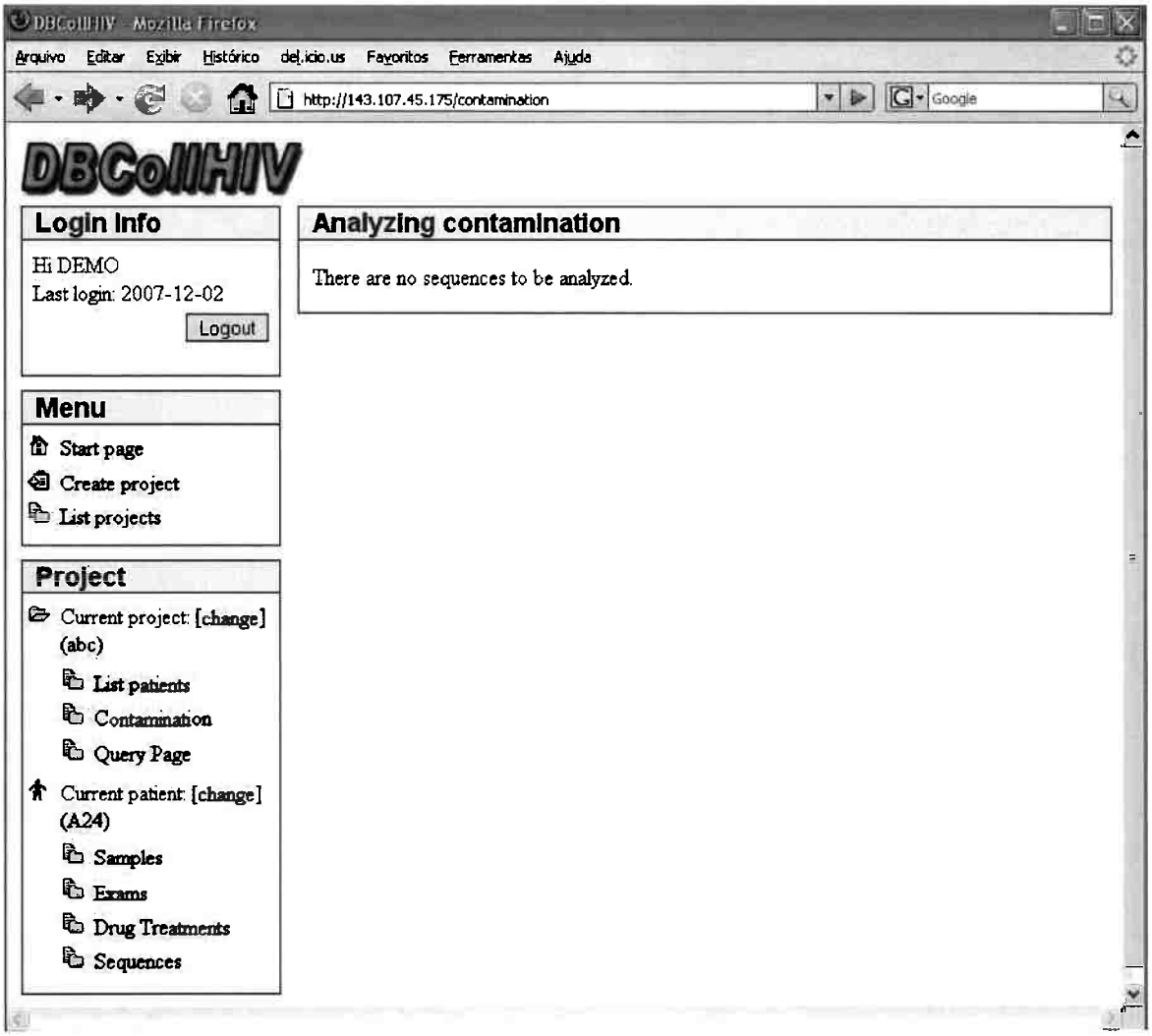

Figura 4.14: Resultado da análise de contaminação.

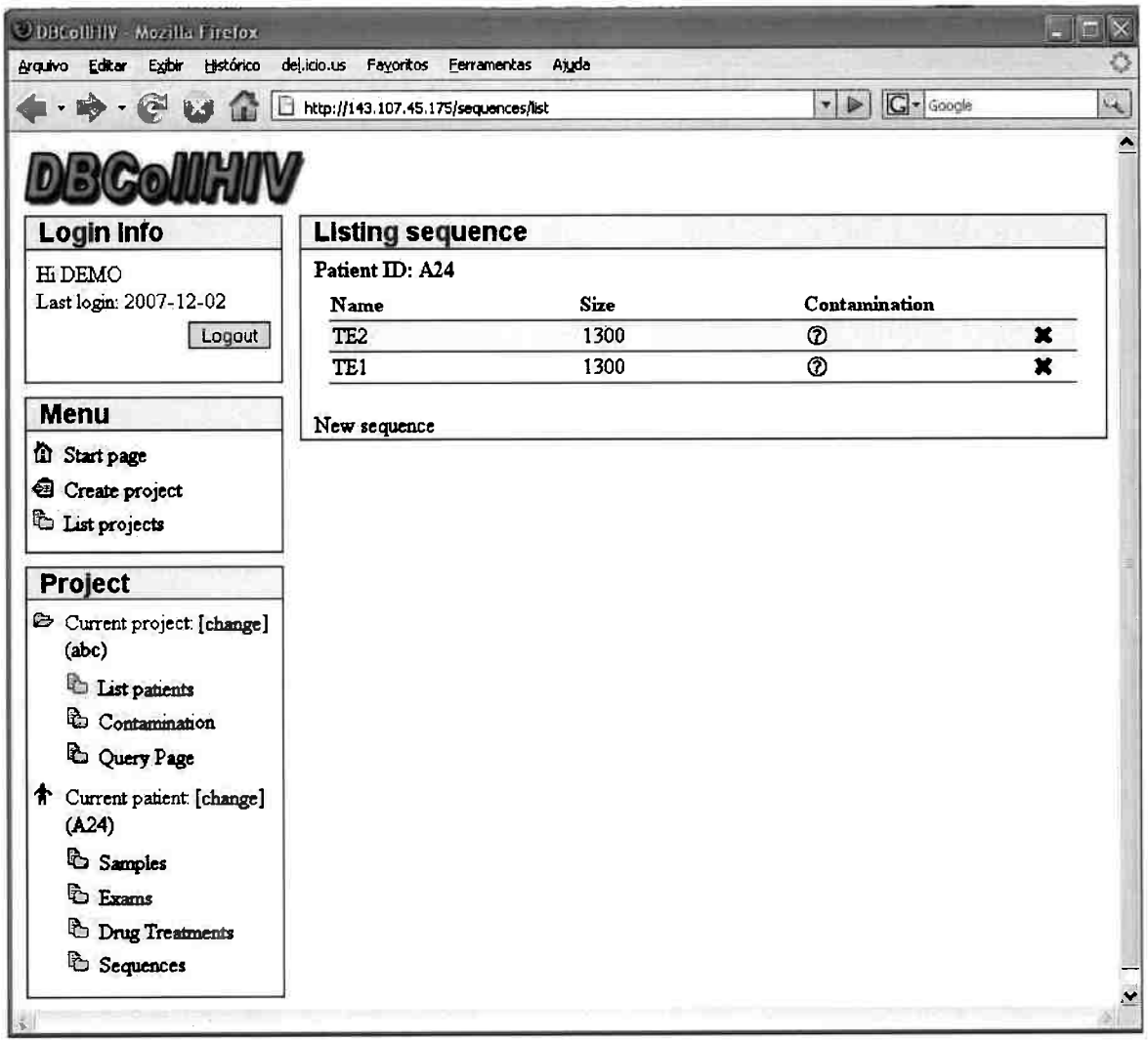

Figura 4.15: Seqüências sem análise de contaminação.

A figura 4.16, mostra a página HTML do *PCR contamination* na qual o usuário pode enviar os seus dados para a análise sem necessitar acesso ao DBCollHIV.

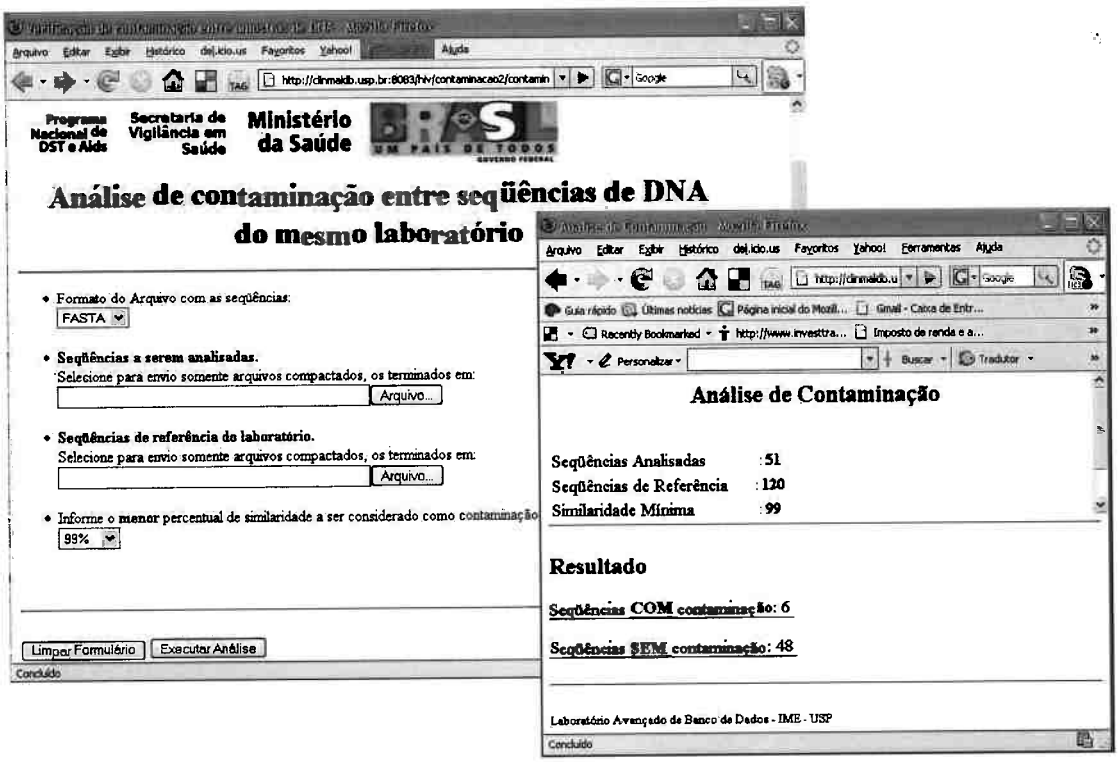

Figura 4.16: Página para acesso o programa PCR contamination e exemplo do resultado gerado.

O PCR Contamination, apesar de sua estrutura simples para preparação de seqüências e avaliação dos resultados do BLAST, desempenha o importante papel de garantir a qualidade dos dados por meio da análise das novas seqüências geradas em um laboratório e da eliminação da execução manual das tarefas necessárias para identificar a contaminação.

Como abordagem prática, o PCR Contamination tem sido utilizado pelo Ministério da Saúde do Brasil e por alguns laboratórios particulares para garantir a qualidade das seqüências geradas. Além disso, o PCR Contamination e os resultados da análise das primeiras 3927 seqüências geradas pelos 14 laboratórios do RENAGENO, distribuídos em regiões estratégicas pelo Brasil, formam apresentados na XV Conferência Internacional de AIDS 2004 (Brindeiro et al., 2004; Araújo et al., 2004).

4.2.2 HIVSetSubtype - Programa para Subtipagem de seqüência de HIV

Essa seção apresenta o programa *HIVSetSubtype*, programa que permite a subtipagem automática, em larga escala, de seqüências de HIV. Sua principal característica é realizar a subtipagem sem que o usuário tenha conhecimentos em filogenia ou necessite avaliar arquivos gerados para todas as seqüências analisada. Dessa maneira, a subtipagem pode ocorrer de uma forma direta, eficiente e atender à necessidade crescente de subtipagem de seqüências.

O *HIVSetSubtype* define o subtipo das seqüências submetidas por meio da comparação dos subtipos obtidos por dois métodos de subtipagem. A seqüência é subtipada quando ambos os métodos definem o mesmo subtipo. Caso contrário, a seqüência é rotulada como não subtipada.

A Figura 4.17 apresenta o diagrama da análise executada pelo *HIVSetSubtype*. No canto esquerdo da figura estão representados três conjuntos de seqüência que serão usados durante a análise, são eles:

1. Na parte superior, seqüências de subtipos puros: A1, A2, B, C, F1, F2, G, H, J, N, O e U;
2. Ao centro, seqüências sem subtipo e que foram submetidas identificação dos mesmos;
3. Na parte inferior, as seqüências de referência, ou seja, um conjunto formado por diversas seqüências com subtipos conhecidos, sejam eles puros ou recombinantes.

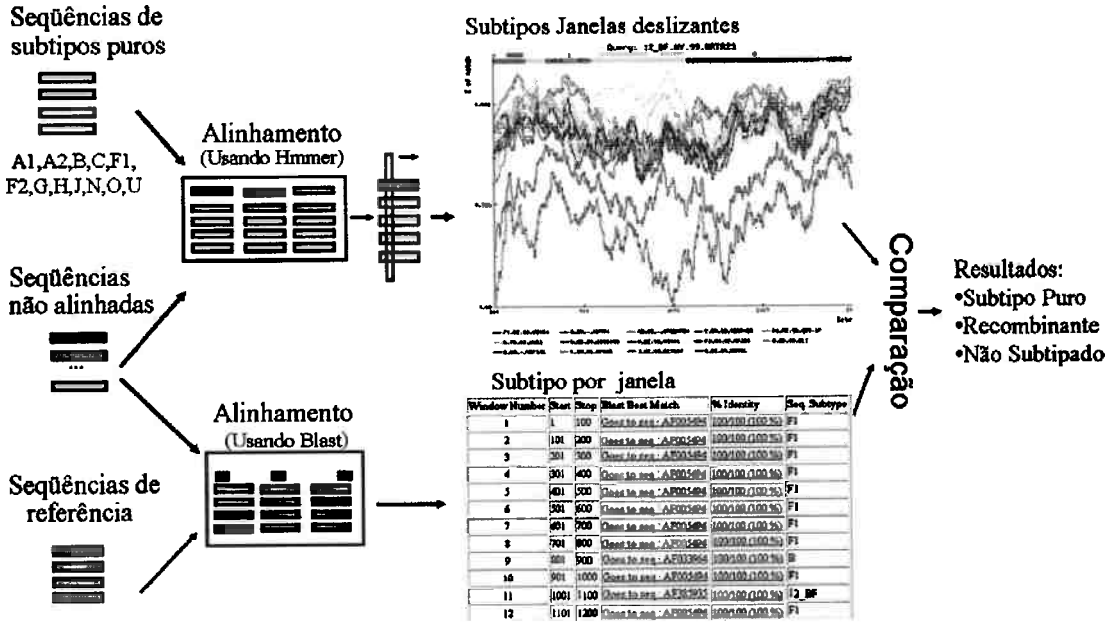

Figura 4.17: Diagrama das análises realizadas pelo HIVSetSubtype.

As seqüências sem subtipo passam por dois métodos de análise. O primeiro método está baseado no programa RIP desenvolvido no LANL (http://www.hiv.lanl.gov/content/hiv-db/RIPPER/RIP.html). O *HIVSetSubtype* alinha a seqüência a ser analisada contra o conjunto de seqüências de subtipo puro, usando o HMMER (Eddy SR, 1998). Em seguida, uma janela deslizante é usada para comparar os segmentos delimitados pela ela. Cada segmento de subtipo puro recebe uma pontuação referente à quantidade de bases similares à seqüência a ser subtipada. Em seguida, a pontuação dos segmentos mais similares é avaliada, pelo teste estatístico Z, para verificar se a diferença entre elas é significante. Quando a diferença é estatisticamente significante, o subtipo da seqüência mais similar define o subtipo da janela. Do contrário, o subtipo mais similar é armazenado, porém com a indicação de que a diferença não é estatisticamente significante.

Para dar continuidade à análise da seqüência, a janela é movida de acordo com o tamanho do passo definido pelo usuário e o processo de avaliação das janelas prossegue por toda a extensão da seqüência. Ao final da análise, se todas as janelas apresentam o mesmo resultado, a seqüência é rotulada com o subtipo puro encontrado. Caso alguma janela possua subtipo diferente, a seqüência recebe o subtipo recombinante. Ou seja, a seqüência é formada por um mosaico de subtipos, como mostra o resultado da análise de uma seqüência de subtipo recombinante BF, Figura 4.18. Nela, a linha no topo do gráfico mostra a cor do

subtipo mais similar encontrado em cada janela, a parte não contínua indica que o trecho não apresentou diferença estatisticamente significante. A segunda linha apresenta a cor do subtipo mais similar em cada janela, independente da diferença dele para o segundo subtipo mais similar. As linhas abaixo das duas linhas do topo do gráfico apresentam a similaridade de cada seqüência de subtipo puro por toda a extensão da seqüência.

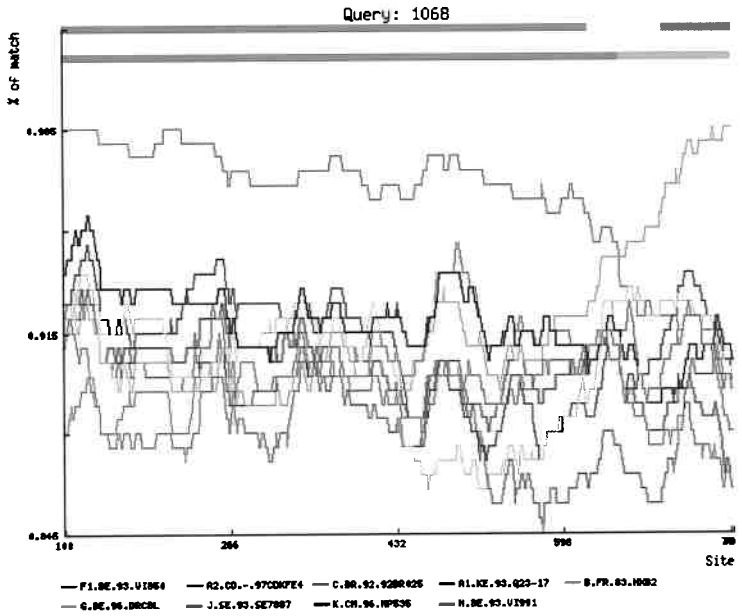

Figura 4.18: Resultado da análise de uma seqüência de subtipo recombinante.

O resultado da análise de um subtipo puro pode ser visto na Figura 4.19. A linha mais ao topo apresenta uma só cor, o que indica que todas as janelas tiveram o subtipo B, linha amarela, como o mais similar. A segunda linha do gráfico apresenta mais de uma cor, porém a diferença de similaridade entre o subtipo B e os demais subtipos não foi significante, portanto, o subtipo B é definido com o subtipo da seqüência.

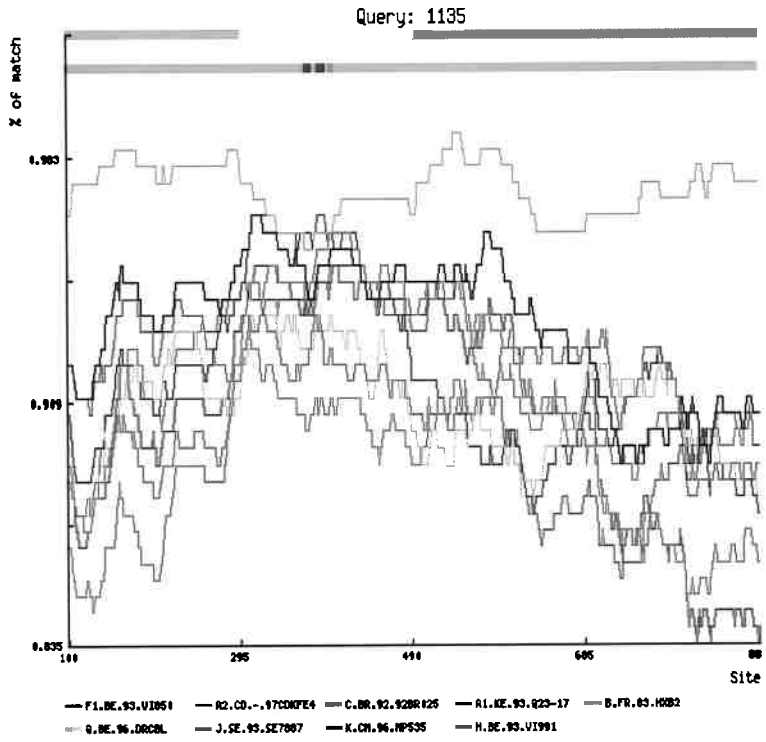

Figura 4.19: Resultado da análise de uma seqüência de subtipo puro.

O segundo método usado pelo *HIVSetSubtype*, divide a seqüência a ser subtipada em tamanhos iguais. Os fragmentos obtidos são alinhados e comparados contra um conjunto de seqüência de subtipos puros e recombinantes, usando o programa BLAST. Similar ao primeiro método, a comparação dos subtipos recebidos por cada fragmento define o subtipo da seqüência, seja ele puro ou recombinante. A Figura 4.20 mostra o resultado da análise de uma seqüência do subtipo recombinante BF. A Figura apresenta as seguintes colunas:

- Window Number - Indica o número da janela ou fragmento analisado;
- Start e Stop - Apresentam, respectivamente, a posição de início e fim da janela;
- Blast Best Match - Informa a seqüência mais similar ao fragmento limitado pela janela e o seu número de acesso no GenBank;
- % Identity - Mostra o percentual de similaridade do segmento da seqüência de referência com o fragmento da seqüência submetida à subtipagem;
- Seq. Subtype - Subtipo atribuído à janela.

Window Number	Start	Stop	Blast Best Match	% Identity	Seq. Subtype
1	1	100	Goes to seq : AF005494	100/100 (100 %)	F1
2	101	200	Goes to seq : AF005494	100/100 (100 %)	F1
3	201	300	Goes to seq : AF005494	100/100 (100 %)	F1
4	301	400	Goes to seq : AF005494	100/100 (100 %)	F1
5	401	500	Goes to seq : AF005494	100/100 (100 %)	F1
6	501	600	Goes to seq : AF005494	100/100 (100 %)	F1
7	601	700	Goes to seq : AF005494	100/100 (100 %)	F1
8	701	800	Goes to seq : AF005494	100/100 (100 %)	F1
9	801	900	Goes to seq : AF033964	100/100 (100 %)	B
10	901	1000	Goes to seq : AF005494	100/100 (100 %)	F1
11	1001	1100	Goes to seq : AF385935	100/100 (100 %)	12_BF
12	1101	1200	Goes to seq : AF005494	100/100 (100 %)	F1

Figura 4.20: Resultado da análise de usando o programa BLAST.

Ao final, os subtipos identificados por cada método são comparados como forma de confirmação do resultado. Assim, se os resultados forem iguais, o subtipo é atribuído à seqüência, seja ele puro ou recombinante. Por outro lado, se subtipos forem diferentes, a seqüência é rotulada como de subtipo indefinido e necessitará de uma análise mais detalhada para definir seu subtipo.

As imagens apresentadas nas figuras 4.18 a 4.20 não precisam ser analisadas pelo usuário. Elas somente são geradas se forem solicitadas. De outra forma, somente o subtipo final da seqüência é informado.

4.2.2.1 Avaliação para definição dos melhores parâmetros para uso do HIVSetSubtype.

Como forma de testar a capacidade de identificação de subtipo do *HIVSetSubtype*, foram analisados dois conjuntos de seqüências, 9287 seqüências, são eles:

1. 1397 seqüências reais de subtipo puros e recombinantes conhecidos:
   - 468 seqüências obtidas do site de Los Alamos e do GenBank ;
   - 929 do laboratório da USP.
2. 7890 seqüências de subtipo recombinantes artificialmente montadas para testar a sensibilidade do programa para a detecção de recombinantes com fragmentos de diversos tamanhos e em diferentes regiões do genoma.

As seqüências foram analisadas com o uso de diferentes tamanhos de janela, para ambos os métodos RIP e BLAST, variando de 50 a 450 bp. Como seqüências de referência para o método RIP, foram usadas seqüências representativas de cada subtipo do HIV-1, exceto os subtipos D e K, pois, os testes mostraram que devido à sua similaridade na região do gene Pol, com os subtipos B e F, respectivamente, os subtipos D e K reduzem a capacidade do *HIVSetSubtype* de detectar recombinantes com o método RIP e aumenta sua taxa de erro.

Das 468 seqüências públicas, 83.8 % contêm seqüências parciais de nucleotídeos que cobrem ambas as regiões da PR e da RT (Barreto, et al., 2005; Gordon et al., 2003), enquanto 16.2 % (76/468) são seqüências com o genoma completo do HIV-1, incluindo 55 não recombinantes A-K seqüências de referência de Los Alamos para o HIV-1 e 21 seqüências com o genoma completo de subtipos puros e recombinantes brasileiros de alguns estudos recentes (Sanabani, et al., 2006 a, b).

Para descrever a capacidade de identificar subtipos recombinantes do HIVSetSubtype, usando diferentes tamanhos de janelas, foi criado um conjunto de 7890 seqüências de subtipo recombinantes a partir da substituição de fragmentos da seqüência receptora de subtipo B (GenBank, número de acesso AY173959) por fragmentos de igual tamanho da seqüência doadora de subtipo F (GenBank, número de acesso AF005494). Foram escolhidos subtipos B e F encontrados no Brasil por serem os subtipos dominantes no Brasil.

Para a criação das seqüências, foi desenvolvido um programa que recebe como parâmetros, duas seqüências, uma receptora e uma doadora, o tamanho da janela de substituição e tamanho do passo da janela. A janela deslizante percorre toda a extensão da seqüência e a região delimitada na seqüência receptora, de subtipo B, é substituída por fragmento de igual tamanho e da mesma região da seqüência doadora, de subtipo F. O recombinante gerado é gravado, a janela é avançada em uma base e o processo repetido até o final da seqüência. Esse programa foi executado para janelas substituição de tamanho variando de 50 a 400 bases e com incremento de 50 bases. Dessa maneira, foram gerados nove grupos de recombinantes, identificados pelo tamanho do fragmento inserções (50, 150, 200, ..., 350, 400).

O *HIVSetSubtype* foi executado para avaliação de cada um dos nove grupos, com tamanho de janela variando de 50 a 400 bases, tanto para a análise usando o BLAST quanto usando o RIP. O que resultou em nove análises de cada grupo. Ou seja, as 7890 seqüências foram analisadas uma vez para cada tamanho de janela, o que equivale à análise de 71.010 seqüências.

Os resultados da análise dos recombinantes artificiais, apresentados na Figura 4.21, mostram que mais de 60% das seqüências com a inserção de fragmentos maiores que 250bp foram corretamente identificadas como recombinantes, quando usadas janelas de análise de tamanho maior ou igual a 150 nucleotídeos. Além disso, pouco mais de 20% das seqüências com inserção de fragmento de até 200 bp foram classificadas como recombinantes com o uso de uma janela de tamanho 150 bp

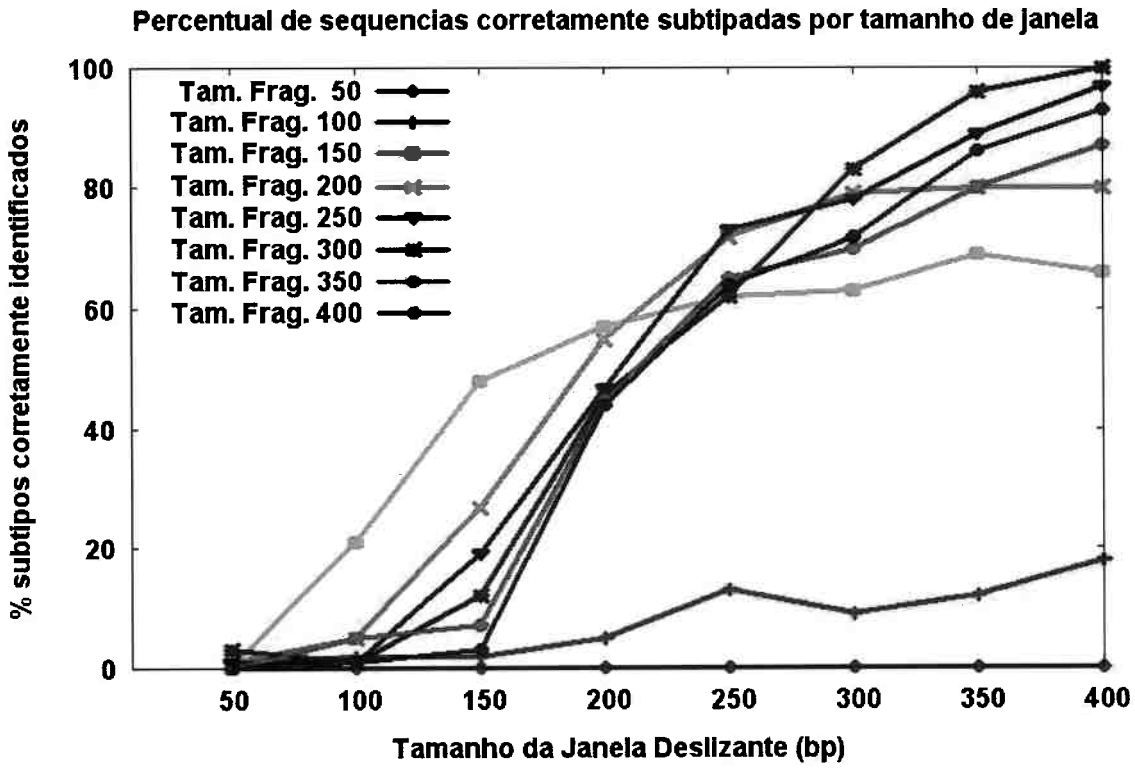

Figura 4.21: Percentual de seqüências corretamente subtipadas por tamanho de janela e fragmento de recombinação.

A Figura 4.22 apresenta a perspectiva dos erros de subtipagem. Ou seja, o percentual de seqüências que foram classificadas como sendo de subtipo puro. Ela mostra que os erros de subtipagem começam a surgir com o uso de janelas de análise com tamanho maior que os fragmentos de inserção. Assim, a maioria das seqüências BF com fragmentos entre 50 e 150 bp foram erroneamente classificadas como subtipos puros para janelas com tamanho superiores a 250 bp.

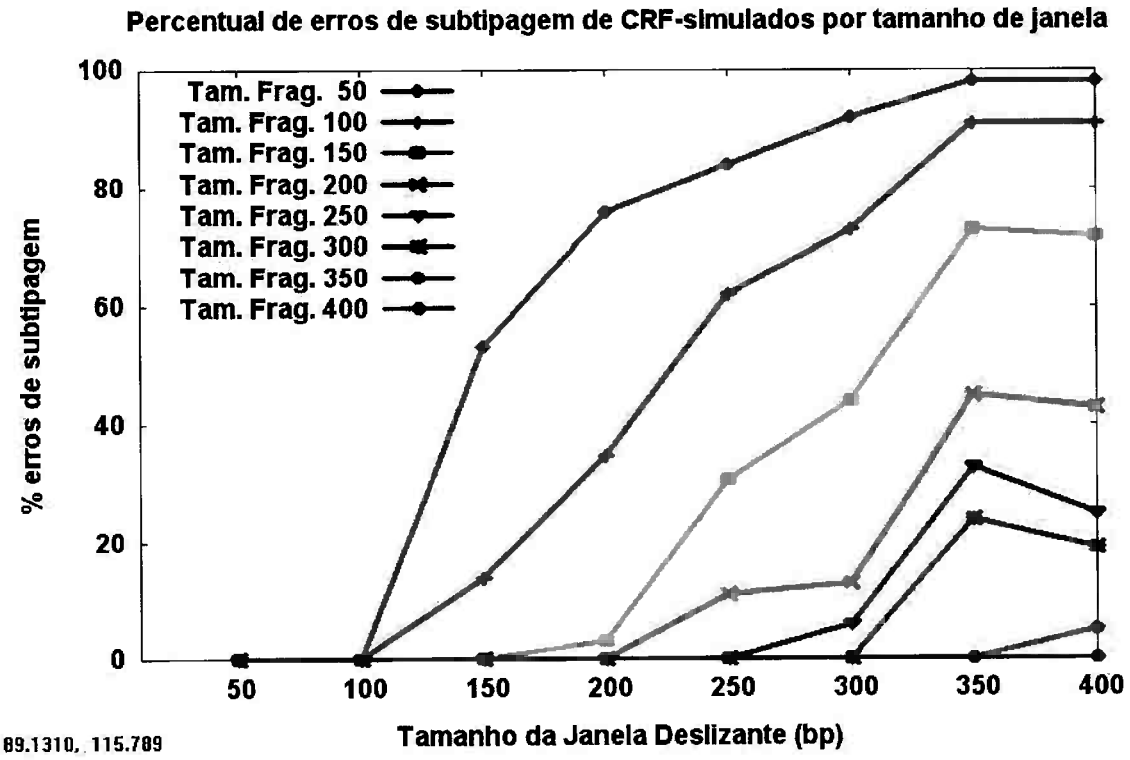

Figura 4.22: Percentual de seqüências erroneamente subtipadas por tamanho de janela e fragmento de recombinação.

Em relação à análise das seqüências reais, conforme o esperado, os resultados apresentados na Figura 4.23 mostraram uma relação inversamente proporcional entre o percentual de seqüências não subtipadas e o tamanho da janela utilizado. Mais de 90 % das seqüências reais foram classificadas como de subtipo indefinido com o uso de uma janela de tamanho de 50 bases. Esse percentual diminuiu para 30% e 4.7% quando o tamanho da janela foi elevado para 100 e 200, respectivamente.

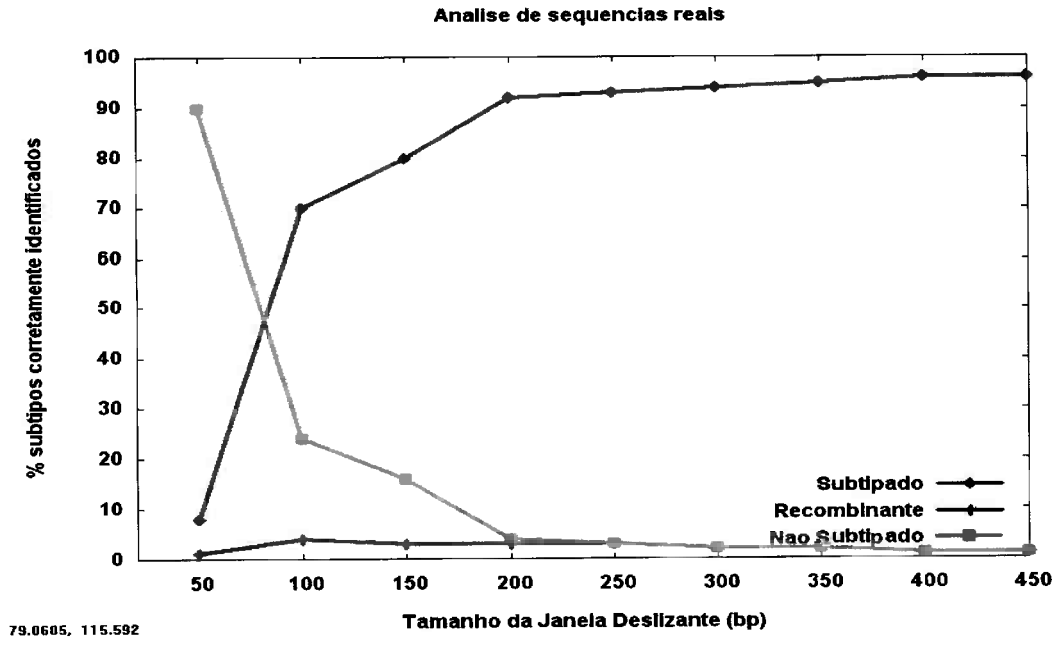

Figura 4.23: Resultado da análise de seqüências reais usando o HIVSetSubtype.

Embora seja interessante ter um programa que consiga identificar corretamente 100% das seqüências recombinantes, essa é uma meta difícil de ser atingida na prática. Uma alternativa é possuir um programa que consiga separar os recombinantes dos não recombinantes. Os testes com HIVSetSubtype avaliaram a chance de seqüências recombinantes serem classificadas de forma errada. Isso foi feito com a análise dos resultados gerados pela execução do programa usando diferentes tamanhos de janela para análise dos diferentes tamanhos de fragmento de recombinação. A Figura 4.22 mostra que fragmentos recombinantes de 50 e 100 bp são difíceis de serem detectados e tais seqüências são classificadas de forma errada como subtipo puro em 80 e 40% dos casos, respectivamente, quando uma janela de análise de tamanho 200 bp é utilizada. Para recombinações maiores, de tamanho 150 e 200 bp, a taxa de erro foi de 3.4% e 0% respectivamente, também com o uso de uma janela de 200 bp. Baseado nesses dados, o melhor tamanho de janela para análise das seqüências é 200 bp, pois apresenta a menor taxa de erro de classificação das seqüências recombinantes. Portanto, a janela de 200 bp é o parâmetro ótimo para uso do HIVSetSubtype. Os dados sobre a avaliação do parâmetro

ótimo são importantes para definir o parâmetro para execuções automáticas quanto para os casos nos quais existe maior conhecimento sobre os possíveis tamanhos dos fragmentos de recombinação.

De 929 seqüências de subtipo submetidas à subtipagem, usando o HIVSetSubtype, 95.37 % foram corretamente subtipadas e somente 4.63% foram rotuladas como de subtipo indefinido. Na reavaliação de subtipo de 468 publicadas e com subtipo conhecido, o HIVSetSubtype apresentou taxa de 99.37% de concordância com os subtipos originais. Somente 3 seqüências (Número de acesso no GenBank AY727526, AY727527 e AY136974) apresentaram um subtipo BC com um padrão de mosaico como relatado em (Gordon et al., 2006; Sanabani et al., 2006). Em busca de confirmação desses resultados, essa seqüências foram novamente analisadas usando métodos de bootscan como implementado no SIMPLOT v3.2 beta mantendo os parâmetros originais, como descrito nos seus estudos iniciais, exceto pelo tamanho de janela configurado para 200bp.

O resultado mostrou que essas seqüências realmente são formadas por um padrão mosaico de subtipos BC, como sugerido pelo HIVSetSubtype. Isso mostra que os resultados do HIVSetSubtype estão 100% de acordo com os resultados originais apresentados por essas seqüências.

4.2.2.2 Constatações sobre o HIVSetSubtype.

O *HIVSetSubtype* foi desenvolvido para abordar o problema de subtipagem do crescente número de seqüências de PR e RT, geradas como parte da rotina para avaliação do tratamento de pacientes. Com o objetivo de aumentar seu desempenho, foram feitos experimentos para definir o melhor tamanho de janela a ser usado, para tanto, foram usadas seqüências reais e seqüências que simularam subtipos recombinantes.

Os resultados dos experimentos mostraram que assim como em outros programas para subtipagem do HIV (Gifford et al., 2006, de Oliveira, et al., 2005), a precisão *HIVSetSubtype* cai para seqüências cujo fragmento de recombinação seja menor que 150 bp. No caso do *HIVSetSubtype*, um pequeno percentual de 3% das seqüências recombinantes simuladas foi erroneamente classificado como sendo de subtipo puro quando foi utilizado um fragmento de tamanho 150 bp.

Apesar do *HIVSetSubtype* usar análises de similaridade, que são menos sofisticada que as análises com uso de árvores filogenéticas, ele foi capaz de classificar 95.37 % das seqüências de PR e RT geradas como rotina em nosso laboratório. Somente 4.63% das seqüências apresentaram resultados discrepantes entre a análise usando o método RIP e BLAST, o que resultou na classificação como subtipo indefinido. Esse pequeno percentual de seqüências com subtipo indefinido pode ser analisado usando métodos filogenéticos mais sofisticados e que demandam cálculos e processamento intensos como Hidden Markov Model (HMM) implementados em HMM-HIV (Schultz et al., 2006).

Para aumentar o poder de detecção de subtipo do HIV SetSubtype, não foram incluídas seqüências de subtipo D e K no conjunto de seqüência de referência para o método RIP, durante a análise de seqüências isoladas no Brasil, devido a razões citadas anteriormente, tais como a similaridade da região do gen Pol de subtipos B e F. Contudo, os usuários são cuidadosamente aconselhados a selecionarem as seqüências de referência que eles desejarem utilizar, tanto para o método RIP quanto para o BLAST, levando em consideração as cepas circulantes na sua região.

A comparação entre os resultados de dois métodos de subtipagem permite validar os resultados obtidos, ao invés de simplesmente considerá-los corretos ou não, e diminui a taxa de erro da classificação, pois seqüências com resultados divergentes não são subtipadas e sim classificadas como de subtipo indefinido. Essa idéia pode ser aplicada usando diferentes métodos ou programas de subtipagem, desde que a execução dos mesmos não aumente o tempo de execução a ponto de inviabilizar o seu uso.

Como aplicação prática, o *HIVSetSubtype* têm sido usado pelo Ministério da Saúde do Brasil para a identificação de subtipo de seqüências HIV-1 submetidas para análise quanto à resistência à droga. O *HIVSetSubtype*, além de integrado ao DBCollHIV, como mostra a figura 4.24, também esta publicamente disponível para uso e sem a necessidade de armazenamento de dados, no site :

http://clinmaldb.usp.br:8083/hiv/subtype/hivsetsubtype.html.

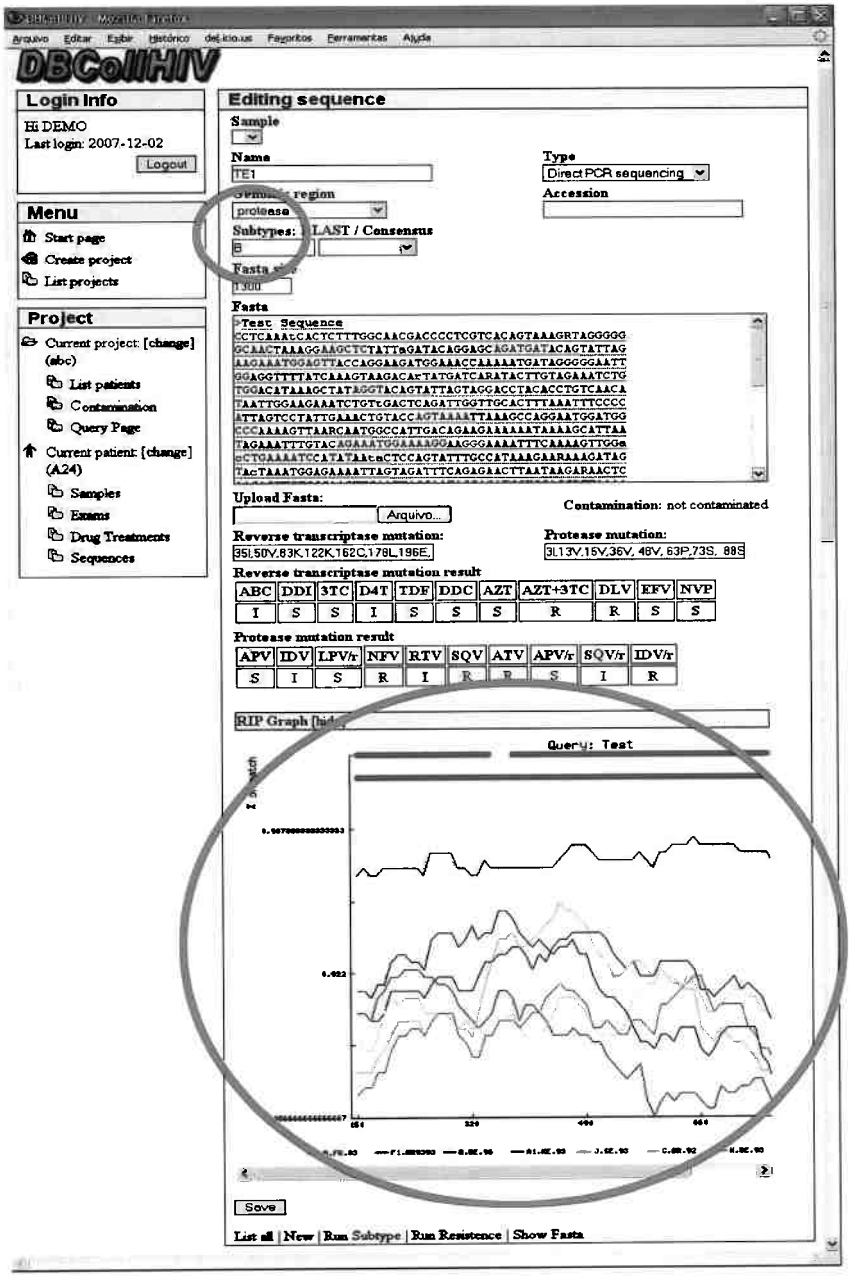

Figura 4.24: Em destaque o resultado da análise de subtipo usando o HIVSetSubtype no DBCollHIV.

4.2.3 HIVdag - Programa para análise de resistência à droga.

Interpretar testes de genotipagem não é uma tarefa simples, pois, entre outros motivos, existem muitas mutações relacionadas à resistência a drogas e uma variedade de medicamentos disponível. Esse cenário permitiu o surgimento de diferentes algoritmos para resistência à droga. Cada um deles baseado em um conjunto de regras que buscam representar a relação entre as mutações e a resistência às drogas disponíveis para tratamento.

Essa seção apresenta o Algoritmo Brasileiro Para Interpretação do Teste de Genotiapagem. O Algoritmo Brasileiro foi desenvolvido para atender a necessidade de um programa para interpretar os testes de genotipagem fornecidos pelo Ministério da Saúde e que fosse baseado não somente nos conhecimentos da comunidade internacional, mas também que contasse com a experiência de médicos e pesquisadores brasileiros e cuja atualização e evolução fosse definida de acordo com as necessidades nacionais.

O Ministério da Saúde do Brasil apoiou as pesquisas para o desenvolvimento do programa para análise de resistência baseado em regras desenvolvidas pelos representantes do comitê do projeto RENAGENO, o que originou o Algoritmo Brasileiro.

As regras para interpretação de resistência às drogas, independente do algoritmo usado, são obtidas de forma empírica baseada nos dados de fenotipagem e de tratamentos de pacientes, portanto não é objetivo desse trabalho discutir as regras que compõem o algoritmo, tais regras estão disponíveis no site do Ministério da Saúde do Brasil no endereço:

http://www.aids.gov.br/final/tratamento/politicas/regras_renageno.htm ou

http://www.aids.gov.br/data/Pages/LUMIS1F16A8CFITEMIDE34AD7CDF6274444A943B0FF968F2E6DPTBRIE.htm

O Algoritmo Brasileiro está disponível para acesso pela Internet pelo site do Ministério da Saúde (http://www.aids.gov.br/data/Pages/LUMIS1F16A8CFPTBRIE.htm) ou com acesso direto ao servidor do Instituto de Matemática e Estatística da USP que hospeda o serviço (http://clinmaldb.usp.br:8083/hiv/resistencia/resistencia.html).

A Figura 4.25 apresenta o site para análise de resistência às drogas utilizadas pelos médicos e laboratórios brasileiros. No site, podem ser submetidos arquivos contendo as mutações geradas pelos testes de genotipagem e a seqüência do vírus para análise do subtipo. Caso o usuário não possua tais arquivos, as mutações podem ser digitadas no site. Após a submissão dos dados, é gerado um relatório contendo os seguintes dados:

- Número identificador do paciente – Informação opcional;
- Subtipo do vírus, caso o arquivo com a seqüência do vírus seja enviado;
- Mutações do vírus classificadas de acordo com sua relação de resistência a droga;
- Drogas inibidoras da Transcriptase reversa e os respectivos níveis de resistência:
    - R – Vírus resistente à droga;
    - I – Vírus com resistência intermediária;
    - S – Vírus suscetível à droga.
- Drogas inibidoras da Protease e os seus níveis de resistência.

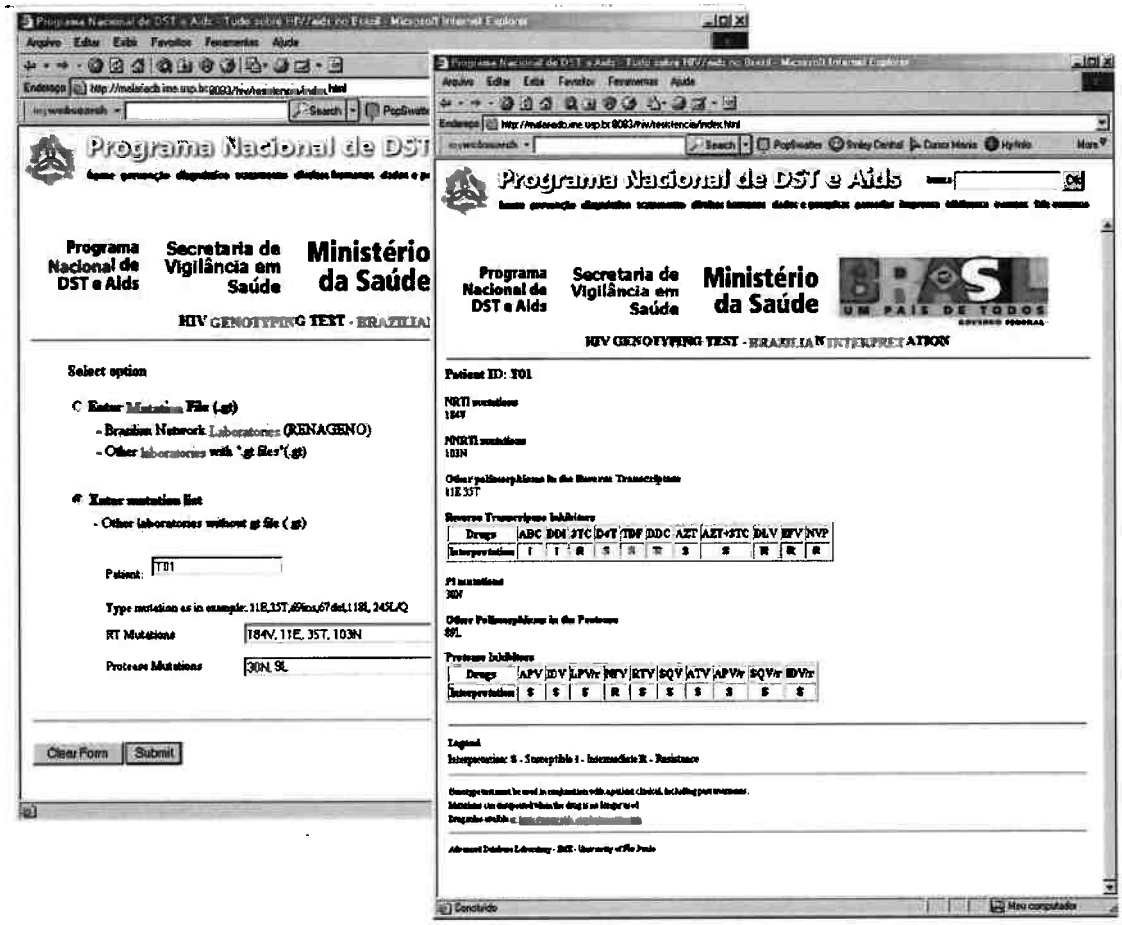

Figura 4.25: Página de acesso e resultado do Algoritmo Brasileiro de Genotipagem.

O Algoritmo Brasileiro de Genotipagem também esta integrado e disponível no DBCollHIV, como mostra a figura 4.26.

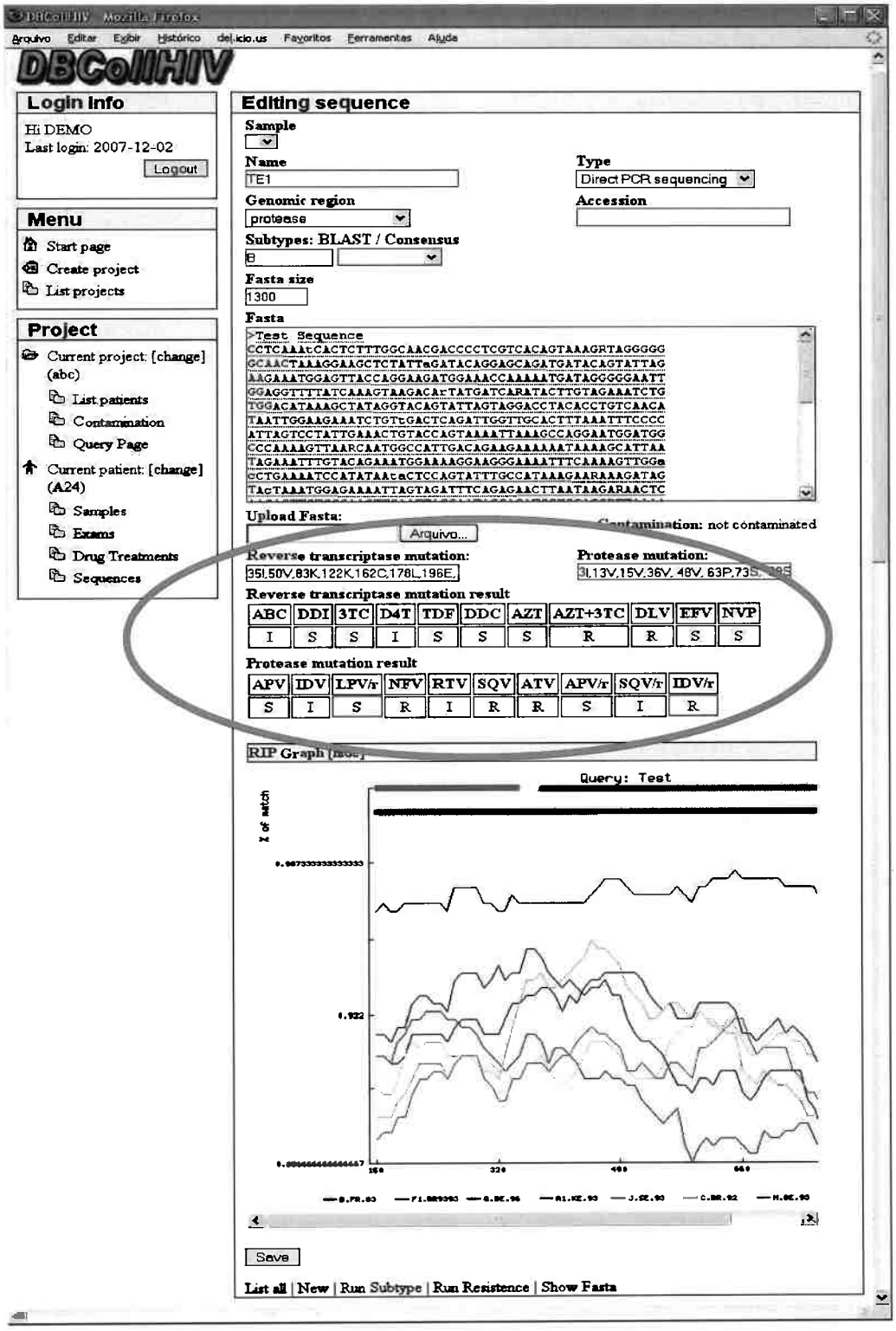

Figura 4.26: Resultado da análise de resistência integrado ao DBCollHIV.

Atualmente, mais de 500 médicos brasileiros estão treinados para desempenhar a função de médicos de referência de genotipagem, ou seja, esses são médicos hábeis para interpretar o relatório gerado pelo Algoritmo Brasileiro de Genotipagem e indicar o tratamento mais adequado ao paciente. A cada ano, mais de 5000 pacientes realizam o teste de genotipagem como parte do tratamento oferecido pelo Ministério da Saúde e os com resultados de resistência gerados pelo Algoritmo Brasileiro de Genotipagem.

4.2.4 Geração automática de programas para análise de resistência à droga.

Inicialmente, o Algoritmo Brasileiro foi desenvolvido na linguagem de programação Perl, porém, a experiência obtida durante as pesquisas e o desenvolvimento do Algoritmo Brasileiro mostraram que a avaliação de regras como as de resistência a drogas, que necessitam de constante de atualização e que podem no futuro contemplar outros dados, além das mutações do vírus, demandam um processo para geração e validação automática do mesmo. Essa constatação motivou a criação de um mapeamento das regras de resistência à droga em expressões da álgebra de processos com o objetivo de gerar, comparar os algoritmos automaticamente e contar com os recursos da álgebra de processos para garantir a corretude e futuras análises das regras.

4.2.4.1 Mapeamento das regras de resistência à droga para expressões da álgebra de processos.

As regras dos algoritmos de resistência à droga apresentam uma estrutura similar, ou seja, cada algoritmo é composto por um conjunto de regras, em que cada uma representa um conjunto de mutações associadas a um nível/penalidade de resistência à determinada droga. Essa estrutura formada por tarefas bem definidas, que podem ser repetidas e cujos resultados influenciam na execução das próximas ações, permite o mapeamento das regras em expressões da álgebra de processos. Suas expressões são baseadas em ações atômicas e

operadores que indicam a ordem de execução de cada uma das ações que compõem a expressão.

As ações atômicas podem ser entendidas como programas/métodos que executam uma tarefa indivisível. Assim, o primeiro passo para o mapeamento das regras de resistência às drogas é a definição de ações atômicas que expressem tarefas executadas pelas regras.

De forma resumida, o processo de análise de mutações consiste na comparação entre as mutações encontradas no vírus, as combinações de mutações definidas pelas regras e atribuição de níveis/penalidades de resistência. Este trabalho definiu 8 ações atômicas que associadas a 4 operadores da NPDL/AP permitem o mapear as regras de resistência a drogas em expressões da AP. As 8 ações atômicas são:

1. ***MS* (MutationSearch)** – Seja ***MS*** uma ação atômica representada por uma função booleana que avalia as mutações encontradas na lista de mutações do vírus e gera o valor verdadeiro, caso seja encontrada uma quantidade de mutações previamente definida por seus parâmetros. O valor falso é gerado caso tal quantidade de mutação não seja encontrada. A ação ***MS*** recebe como parâmetro: uma lista de mutações que deve ser comparada com as mutações encontradas no vírus, um par de valores inteiros representando a quantidade mínima e máxima de mutações que devem ser encontradas na lista de mutações do vírus para que a função booleana retorne um valor verdadeiro. Se o valor máximo for igual a zero, o conjunto de mutações deve estar ausente no vírus para que a ação ***MS*** retorne valor verdadeiro. Caso o valor máximo seja vazio ou *Null* indica que não existe um valor máximo de mutações encontradas para que MS gere um valor verdadeiro.
2. ***SS* (SetScore) –** Seja ***SS*** uma ação atômica que incrementa a soma total de penalidades de uma droga. O incremento é feito de acordo com o valor inteiro passado como parâmetro.
3. ***SC* (ScoreClassification)** - Seja ***SC*** uma ação atômica que recebe como parâmetro um valor inteiro representando o total da pontuação de penalidades de uma droga e retorna ao nível de resistência à droga correspondente.
4. ***RS* (ResultsSynchronization) –** Seja ***RS*** uma ação atômica que sincroniza o resultado de diferentes regras de uma droga. Ou seja, ela recupera os resultados de

todas as regras de uma droga e retorna o resultado mais significativo como o nível de resistência associado à droga.

5. ***SRL* (SetResistanceLevel)** – Seja ***SRL*** uma ação atômica que atribui a droga o nível de resistência passado por parâmetro.
6. ***RE* (ResultsEquivalence)** – Seja ***RE*** uma ação atômica que estabelece a relação de equivalência entre os resultados dos algoritmos por meio dos parâmetros recebidos. Seus parâmetros são: identificação do algoritmo, valor do resultado, identificação do algoritmo a ser comparado e resultado equivalente.
7. ***ARC* (AlgorithmsResultComparison)** – Seja ***ARC*** uma ação atômica que recupera e compara o resultado de cada algoritmo executado.
8. ***GO* (Go On)** – Seja ***GO*** uma ação silenciosa que simula o comportamento de uma ação silenciosa da álgebra de processos. Ou seja, ela não avalia resultados, apenas dá continuidade a execução da próxima ação da expressão da álgebra de processos.

Os operadores da NPDL usados no mapeamento das regras de resistência a drogas são:

1. Composição alternativa "+" – Define que os termos unidos pelo operador são ambos executados, porém um de cada vez. Ex.: A + B indica que ambos A e B serão executados, podendo ser na ordem (A e B) ou (B e A);
2. Composição seqüencial "." – Executa os termos unidos por ele de forma seqüencial, ou seja, na expressão A.B, o termo B só é executado ao final da execução de A;
3. Composição paralela "||" – Executa seus termos de forma simultânea;
4. Execução condicional "**% *r***" - Executa a ação associada a ela somente se a regra booleana ***r*** gerar um valor verdadeiro. O complementar da execução condicional é representado por "**% ! *r***", nesse caso, a ação associada somente será executada se a regra ***r*** gerar um valor falso. O operador % pode ser associado a uma *ação silenciosa* que não executa nenhuma tarefa, ela apenas indica que a avaliação da expressão deve continuar.

Considerando a lista de operadores da NPDL {•, +, ||, **%r**} e as ações atômicas {***MS, SS, SC, RS, SRL, RE, ARC, GO***} é possível o mapeamento das regras de resistência a drogas em expressões da álgebra de processos conforme disponível na linguagem NPDL.

Para exemplificar o mapeamento, são usadas regras do Algoritmo Brasileiro e, em especial, as regras para análise de resistência à droga DLV, apresentadas a seguir:

1. *DLV1 = Presença de 1 ou mais de (100I, 181C/I/L, 188L, 230L, 236L) E ausência de 190A - R (resistente);*
2. *DLV2= Presença de 1 ou mais de (225H, 227L) OU Presença de 1 ou mais de (106A, 103N) – I (Intermediário);*
3. *DLV3= Presença de 1 ou mais de (106A/M, 103N/H/T/S V) E ausência de (190A,225H,227L) - R;*
4. *DLV4= Presença de 1 ou mais de (190E) - R.*

O mapeamento da primeira regra da droga DLV pode ser observado a seguir:

DLV1 = % MS(100I,181C/I/L,188L,230L,236L;1;null) GO •
% MS(190A;0;0) GO •
SRL(R)

Essa expressão representa duas execuções seqüênciais da ação ***MS*** indicada pelo operador composição seqüencial •, seguida pela atribuição do nível de resistência **R** (resistente) ao final da expressão.

A ação ***MS*** recebe três parâmetros separados por ponto-e-vírgula. O primeiro parâmetro contém a lista de mutações a ser procurada, separadas por vírgula. O segundo parâmetro indica a quantidade mínima a ser encontrada e o último parâmetro indica a quantidade máxima a ser encontrada. Dessa maneira, a primeira execução de ***MS*** verifica a presença das mutações *100I,181C/I/L,188L,230L,236L* entre as mutações do vírus. Se, no mínimo, uma das mutações for encontrada, sem um limite máximo, a ação ***MS*** gerará um valor verdadeiro. Caso contrário, o resultado de ***MS*** será falso. Após a execução de ***MS***, o operador execução condicional **%** verificará o resultado gerado. Se o resultado de ***MS*** for

verdadeiro, a ação silenciosa ***GO*** será liberada para execução. Devido a sua definição, a Ação ***GO*** simplesmente permitirá que o restante da expressão seja avaliado.

Em seguida, o operador composição seqüencial • definirá qual o próximo passo a ser executado. Caso a primeira execução de ***MS*** gere um valor falso, a execução da regra será finalizada, pois a primeira condição não foi satisfeita. Na hipótese da primeira execução de ***MS*** gerar resultado verdadeiro, a segunda execução de ***MS*** é liberada para execução. Ela verificará a presença da mutação 190A. Porém, como seus parâmetros máximo e mínimo possuem valor 0 (zero), a ação ***MS*** verificará a ausência da mutação 190A. Ou seja, se o vírus não possuir a mutação 190ª, a ação retorna o valor verdadeiro, do contrário, o valor será falso.

Ao final da segunda execução da ***MS***, o operador execução condicional % avalia o resultado gerado. Se for verdadeiro, a execução da ação ***SRL*** é liberada, senão, a execução da regra é terminada, pois a segunda condição não foi satisfeita. Nesse caso, a droga permanecerá com o nível de resistência atual. Caso a ação ***SRL*** *seja executada*, a droga ***DLV*** receberá o nível de resistência R (resistente).

A segunda regra da *DLV* apresenta o uso de uma composição condicional. Ou seja, as análises das mutações poderão ser executadas em qualquer ordem. Nesse caso, tais análises devem ser colocadas entre parênteses para evitar erros na composição da expressão da NPDL. A regra ***DLV2*** é mapeada para a expressão a seguir.

DLV2 = (%MS(225H, 227L; 1; null) GO + %MS(106A, 103N; 1 ;null) GO) •
SRL(I)

Nessa expressão, as duas execuções de **MS** podem acontecer em qualquer ordem e basta que uma delas gere um valor verdadeiro para que a execução da ação ***SRL*** seja liberada para execução e atribua o nível de resistência I para a droga *DLV*. Caso nenhuma gere valor verdadeiro, a avaliação da expressão é finalizada sem a execução da ação ***SRL.***

Com base no mesmo processo de mapeamento, as duas regras restantes são mapeadas para as seguintes expressões da NPDL.

DLV3 = %MS(106A/M, 103N/H/T/S/V;1; null) GO •
%MS(190A, 225H, 227L; 0; 0) GO •
SRL(I)

DLV4 = %SM(190E; 1; null) GO • SRL(R)

Depois do mapeamento de todas as regras para a droga ***DLV***, o processo que representa avaliação de todas as regras para avaliar o nível de resistência para a droga ***DLV*** deve ser mapeado. Para tanto, é importante entender se as regras são ou não independentes. Ou seja, se a ordem de execução das regras influencia o resultado final. Regras dependentes devem ser encadeadas usando os operadores • e + que controlarão sua ordem de execução. Já as regras independentes podem ser executadas de forma simultânea e devem ser encadeadas pelo operador ||. Por serem independentes, as regras para a droga *DLV* podem ser executadas de forma paralela, como mostra a expressão a seguir.

DLV = (DLV1 || DLV2 || DLV3 || DLV4) • RS

Nessa expressão, cada uma das regras é executada de forma simultânea e, ao final, os resultados gerados são recuperados e avaliados pela ação ***RS***. A ação ***RS*** atribui à droga **DLV** o nível de resistência mais significativo entre os valores encontrados, considerando que o nível **R** (Resistente) é mais significativo que o nível *I* (Intermediário) que por sua vez é mais significativo do que o nível **S** (Suscetível).

Após a criação de regras e processos para análise de todas as drogas, deve ser criado o processo que define a análise de um algoritmo. De maneira similar, a criação do processo de análise para a droga *DLV*, os processos de análise das drogas de um algoritmo podem ser executados de forma independente e paralela. Logo, o processo pode ser mapeado como na expressão a seguir, que representa o Algoritmo Brasileiro em sua versão 4, composta por 21 processos para analisar 21 drogas.

BRASIL_V4 = (ABC ||DDI||3TC || D4T || TDF || DDC || AZT || AZT_3TC || DLV || EFV || NVP || APV || IDV || LPV/r || NFV || RTV || SQV || ATV || APV_r || SQV_r || IDV_r).

Tal representação permite que os sub-processos, que avaliam a resistência para cada droga, sejam executados em de forma simultânea e que cada regra seja avaliada de acordo com os operadores da NPDL.

O mapeamento dos algoritmos ANRS e REGA seguem os mesmos passos e usam as mesmas ações usadas para o algoritmo brasileiro. Entretanto, o mapeamento do algoritmo HIVdb necessita das ações ***SS*** **(SetScore)** e **SC ScoreClassification)** para mapear a atribuição de pontos e para classificar o nível de resistência à droga de acordo com a totalidade de pontos, respectivamente.

A Figura 4.27 apresenta o XML da regra para a droga DLV do algoritmo HIVdb.

```
<DRUG>
<NAME>DLV</NAME>
        <RULE>
        <CONDITION>
        SCORE FROM (98G => 5,
                100I => 40,
                100V => 10,
                ...
                318F => 50
                )
        </CONDITION>
</RULE>
</DRUG>
```

Figura 4.27: Regra do algoritmo HIVdb para a droga DLV.

Com o uso das ações **SC** e **SS**, a regra é mapeada para a seguinte expressão.

DLV = (%MS(98G;1;null) GO • SS(5) + %MS(100I;1;null) GO • SS(40) + %MS(100V;1;null) GO • SS(10) + %MS(101E/P;1;null) GO • SS(5)+ ... + %MS(318F;1;null) GO • SS(50)) • SC

Após cada execução de ***MS*** com resultado verdadeiro, a ação ***SS*** incrementa ao total de pontos de penalidade da droga com o valor passado por parâmetro. Ao final da execução

da expressão, a ação SC atribui à droga o nível de resistência baseado no total de pontos acumulados.

Como resumo, o mapeamento das regras de resistência à droga para expressões da álgebra de processos é feito por meio de ações atômicas ordenadas por operadores da álgebra de processos. Depois disso, as várias regras criadas são combinadas em um processo que representa a análise de resistência de uma droga. Por fim, os processos para análise de cada droga são encadeados usando operadores da AP em um processo principal que representa a análise a ser realizada pelo algoritmo.

A comparação dos resultados dos algoritmos compreende a execução concorrente dos algoritmos analisados seguida de um sincronismo realizado pela ação **ARC**, como na expressão:

$$P = (HIVdb||ANRS||Brazilian\_Algorithms) \bullet ARC$$

Para a avaliação de outros dados que não sejam mutações, como: subtipo do vírus, aderência ao tratamento, drogas usadas no tratamento do paciente, tempo de tratamento, resultados de exames laboratoriais, etc. Basta definir ações atômicas capazes de avaliar cada um dos novos parâmetros e usá-las na definição das expressões.

4.2.4.2 HIVdag – Programa para geração de testes de genotipagem.

Após o mapeamento das regras dos algoritmos para análise de resistência, foi criado um programa chamado de *HIVdag*, HIV drug resistance analysis generator. O *HIVdag* permite ao usuário definir regras, criar algoritmos para testes de genotipagem e comparar os resultados gerados por eles com o uso de uma interface simples que dispensa a necessidade de conhecimentos sobre a álgebra de processos.

O *HIVdag* está baseado nos conceitos da álgebra de processo que formalmente descreve o comportamento do sistema e usa a linguagem NPDL para gerenciar as expressões da álgebra de processos que representa as regras dos algoritmos. Ele foi implementado usando a linguagem de programação Ruby on Rails (www.rubyonrails.org), o banco de dados PostgreSQL (www.postgresql.org) e o web services integrado à linguagem NPDL chamado de NavigationPlanTool (Braghetto et al., 2007).

Conforme apresenta a figura 4.28, o *HIVdag* oferece uma interface amigável para que o usuário possa definir as regras de seu algoritmo. Na parte superior da interface, o usuário informa o algoritmo e a droga a qual a regra pertence, o tipo de resultado atribuído pela regra, nível de resistência ou pontuação de penalidade e o resultado conferido pela regra. Na parte inferior da interface, o usuário define as mutações a serem analisadas e suas associações. O que é feito por meio dos campos:

- Begin Block – Permite ao iniciar um bloco dentro da regra, usando o símbolo abre parênteses. O bloco é necessário para encadear condições onde existam operadores OR e AND;
- Occurrence type – Define se a regra deve avaliar presença ou ausência do conjunto de mutações;
- Min Mutation – Indica a quantidade mínima de mutações que devem ser encontradas na lista de mutações do vírus;
- Max Mutation – Indica a quantidade máxima a ser encontrada;
- Mutation set – Conjunto de mutações que deve ser procurado entre as mutações do vírus;
- End Block – Finaliza um bloco iniciado com o Begin Block. É representado pelo símbolo – fecha parênteses;

- Operator – Operadores utilizados para encadear blocos de mutações para análise. Os operadores AND e OR estão disponíveis. O campo não é obrigatório;
- Options - Apresenta as opções de incluir ou remover uma linha ao conjunto e linhas à regra.

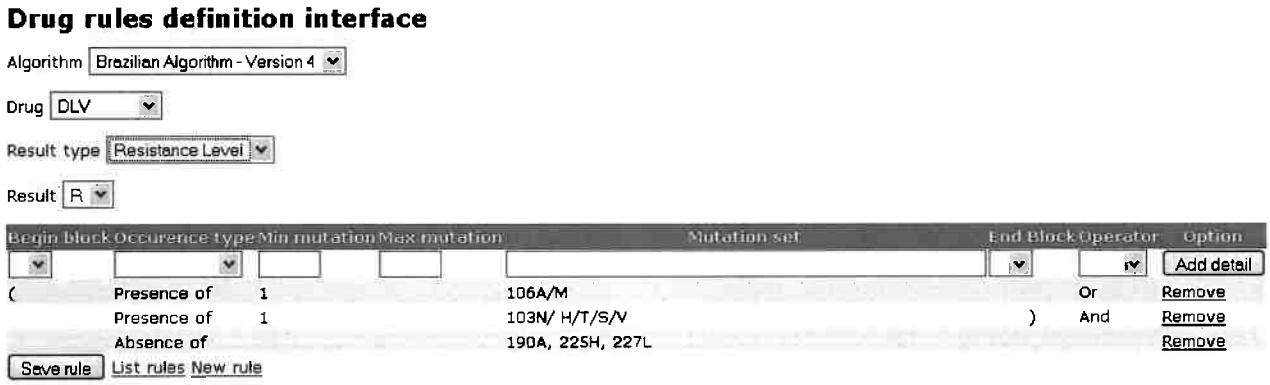

Figura 4.28: Tela para definição de regras de resistência à droga.

A regra que aparece na Figura 4.28 é a regra para a droga **DLV** da versão 4 do Algoritmo Brasileiro. Após sua definição, ela é mapeada pelo *HIVdag* para a expressão:

DLV1 = ((%MS(106A/M, 1, null) GO + %MS(103N/ H/T/S/V,1,null) GO) • %MS(190A, 225H, 227L, 0,0) GO) • SRL(R)

Dessa maneira, o usuário pode criar suas regras sem possuir conhecimentos sobre a álgebra de processos. Nela, a definição das regras é feita de maneira similar ao modo como as regras são normalmente descritas. Após a definição de todas as regras, o algoritmo pode ser gerado automaticamente e liberado para uso.

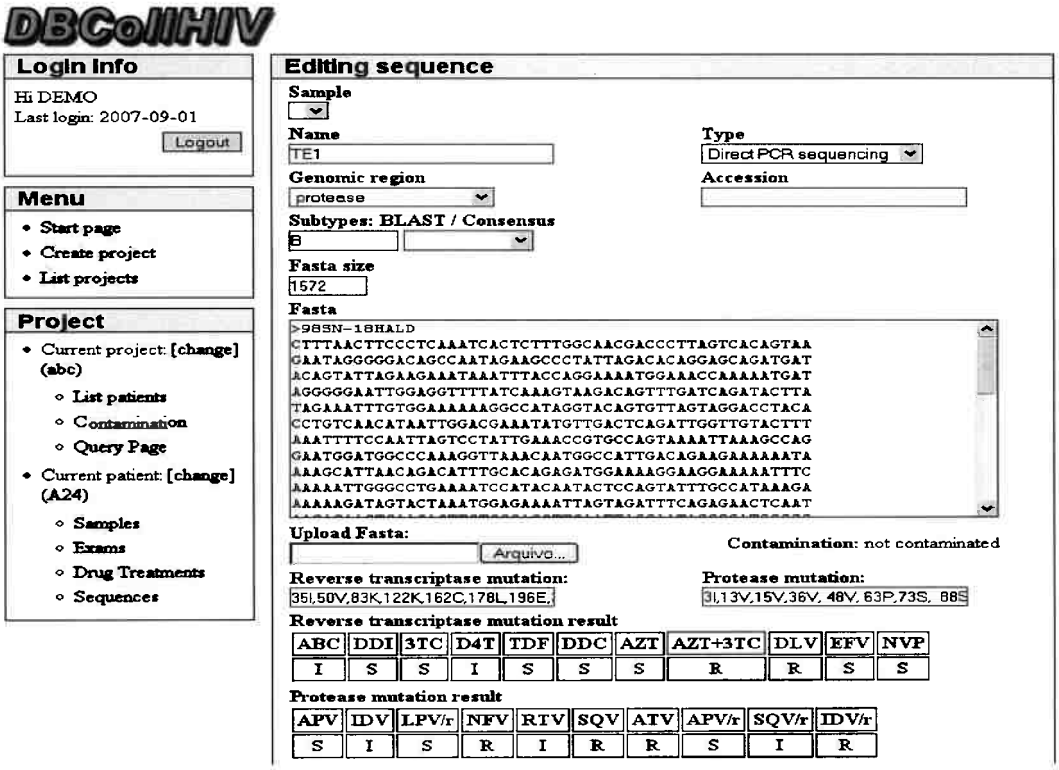

Figura 4.29: Tela de resultado do HIVdag no DBCollHIV.

A apresentação dos resultados obtidos segue o padrão do Algoritmo Brasileiro, como na integração ao DBCollHIV, como mostra a Figura 4.29.

O *HIVdag* também realiza análise de grupos de seqüências, com o objetivo de gerar o perfil de resistência dos pacientes analisados como mostra a figura 4.30. Nela é apresentada a quantidade e o percentual de pacientes em cada nível de resistência por droga. As drogas são apresentadas em dois grupos: inibidoras da transcriptase reversa e da protease.

Drug Resistance distribution per drug

Reverse Transcriptase Inhibitors

Drug	Resistant		Intermediate		Susceptible	
	Qty	(%)	Qty	(%)	Qty	(%)
ABC	231	51.11	153	33.85	68	15.04
DDI	191	42.26	191	42.26	70	15.49
3TC	352	77.88	27	5.97	73	16.15
D4T	254	56.19	42	9.29	156	34.51
TDF	189	41.81	0	0.00	263	58.19
TDF+3TC	59	13.05	121	26.77	272	60.18
DDC	344	76.11	22	4.87	86	19.03
AZT	295	65.27	32	7.08	125	27.65
AZT+3TC	268	59.29	25	5.53	159	35.18
DLV	164	36.28	25	5.53	263	58.19
EFV	265	58.63	8	1.77	179	39.60
NVP	273	60.40	0	0.00	179	39.60

Protease Inhibitors

Drug	Resistant		Intermediate		Susceptible	
	Qty	(%)	Qty	(%)	Qty	(%)
APV	171	37.83	47	10.40	234	51.77
IDV	223	49.34	58	12.83	171	37.83
LPV	102	22.57	51	11.28	299	66.15
NFV	258	57.08	28	6.19	166	36.73
RTV	206	45.58	56	12.39	190	42.04
SQV	216	47.79	26	5.75	210	46.46
ATV	175	38.72	28	6.19	249	55.09
APV/R	123	27.21	37	8.19	292	64.60
SQV/R	166	36.73	44	9.73	242	53.54
IDV/R	184	40.71	39	8.63	229	50.66
ATV/R	132	29.20	43	9.51	277	61.28
TPV	5	1.11	29	6.42	418	92.48
TPV/R	0	0.00	34	7.52	418	92.48

Figura 4.30: *HIVdag* relatório de perfil de resistência.

A análise do perfil de resistência de pacientes é útil para monitoramento das resistências, tomada decisões logísticas quanto à compra/produção de medicamentos e para medir o impacto de novas versões de algoritmo no perfil dos pacientes. Todas essas aplicações permitem a antecipação de ações que podem colaborar para um melhor tratamento do paciente. Tais informações são úteis para países, como o Brasil, que possuem um programa de distribuição de drogas e tratamento.

Capítulo 5

Conclusão

O DBCollHIV foi desenvolvido com o objetivo de oferecer apoio computacional às tarefas de análise e armazenamento de dados necessários às pesquisas sobre o HIV. Para tanto, foram identificadas necessidades tanto de organização e armazenamento de dados quanto de programas para realização de análise dos mesmos. Além de manter a perspectiva de um ambiente que permita a reavaliação constante das análises realizadas, estímulo à cooperação entre pesquisadores e facilidade de uso, tais soluções foram baseadas em recursos computacionais voltados para os requisitos de oferecer ferramentas simples de usar, com capacidade para trabalhar com volume de dados e com a filosofia de software livre, para que sua expansão possa ser contínua e com participação de outros colaboradores.

Como projeto de bioinformática, o DBCollHIV apresenta soluções para problemas biológicos com o uso de recursos computacionais avançados. A abordagem do problema biológico demandou a criação de um banco de dados que pudesse captar tanto dados primários, informados pelo usuário, como dados secundário, gerados pelas ferramentas de análise. Além de oferecer facilidades para modificações e expansão do mesmo, a integração dos dados, ferramentas e resultados proporciona ao pesquisador recursos importantes para planejar suas pesquisas, gerenciá-las e garantir a qualidade de seus dados com as validações e reavaliações de seus dados.

Quanto às ferramentas de análise, o DBCollHIV oferece um conjunto de análises e validações necessárias para a maioria dos estudos de HIV. Em especial, o *HIVdag* permite que o pesquisador não só analise seus dados como também desenvolva suas pesquisas sobre as regras para definição de resistência à droga.

A seguir são detalhadas as principais contribuições deste trabalho.

5.1 Contribuições

5.1.1 Ambiente integrado para análise de dados.

O DBCollHIV é um ambiente de análise e armazenamento de dados sobre HIV que integra um conjunto de recursos úteis ao estudo do HIV, baseado em software livre e que apresenta conceitos atuais na concepção de sua estrutura de banco de dados e de ferramentas de análise. Um ambiente integrado como o DBCollHIV, evita que os dados sejam acumulados de forma desorganizada, facilita o armazenamento dos resultados obtidos pelas análises e favorece a reutilização em outras pesquisas, ao invés de serem esquecidos em arquivos com formatos heterogêneos que inviabilizam sua reutilização.

O arcabouço de recursos oferecidos pelo DBCollHIV permite ao pesquisador realizar pesquisas com o HIV sem a necessidade de investir tempo e recursos financeiros com o desenvolvimento de programas como os que estão disponíveis. Mesmo no caso em que o DBCollHIV não atenda completamente às necessidades de uma determinada pesquisa, ele pode ser ampliado a partir dos recursos existentes.

5.1.2 Ambiente cooperativo para estudos de HIV.

A cooperação entre grupos de pesquisas para compartilhar dados e conhecimento é uma tendência que acompanha o sucesso de iniciativas de software livre. Quanto mais dados de qualidade estiverem disponíveis para análise, maior a possibilidade de avanço da ciência. Alinhado a essa perspectiva, o DBCollHIV oferece a possibilidade de cooperação em duas áreas. A primeira delas é a cooperação por meio do compartilhamento de uma infra-estrutura computacional que permite a realização de uma variedade de pesquisas de forma imediata. Assim, essa infra-estrutura permite que pesquisas possam ser realizadas por grupos que não possuem recursos para sua criação e manutenção. Além disso, usuários do ambiente podem patrocinar ou desenvolver e integrar novas funcionalidades ao DBCollHIV.

A segunda área para cooperação envolve os dados armazenados. O compartilhamento de dados entre grupos permite uma visão cada vez mais abrangente do objeto de estudo, no caso o HIV. Por exemplo, um banco de dados com maior volume de

dados permite a confirmação de padrões de comportamentos e mesmo a avaliação de casos raros.

Apesar de recomendado, o uso do DBCollHIV não obriga o compartilhamento de dados. Além disso, a decisão de qual dado compartilhar, com quem compartilhar e por quanto tempo pertence ao proprietário dos dados.

5.1.3 Ferramentas para análise de dados.

5.1.3.1 Análise de contaminação.

A análise para identificação de contaminação, apesar de apresentar uma solução computacional simples, desempenha o importante papel de garantir a qualidade das seqüências geradas pelos laboratórios. Antes dela, a identificação de contaminação demandava conhecimento sobre o uso e avaliação dos resultados de programas de alinhamento de seqüência como o BLAST e manipulação de diferentes conjuntos de seqüências, o que resultava em uma tarefa trabalhosa, repetitiva e sujeita ao erro.

Com o uso do *PCR contamination* as seqüências geradas diariamente por um laboratório podem ser comparadas com o conjunto de seqüências já produzidas por ele em busca de possíveis contaminações.

A análise feita pelo *PCR contamination* foi incorporada com parte do procedimento dos laboratórios vinculados à rede RENAGENO do Ministério da Saúde Brasileiro e de alguns laboratórios particulares no Brasil. Todas as seqüências geradas por esses laboratórios devem ser analisadas antes que laudos sejam liberados para os pacientes.

5.1.3.2 Análise de identificação de subtipo.

Devido à importância de conhecer o subtipo do HIV, o *HIVSetSubtype* contribui com uma forma simples e eficiente de subtipar seqüências do HIV sem a necessidade de conhecimentos de filogenia.

O *HIVSetSubtype* tem contribuído de forma prática com a subtipagem das seqüências geradas pelo projeto RENAGENO e de seqüências submetidas para análise de resistência a drogas.

5.1.3.3 Análise de resistência à droga.

O desenvolvimento do programa para análise de resistência à droga atendeu a demanda de um teste de genotipagem nacional ajustado às características locais. Hoje mais de 500 médicos já foram treinados para interpretar seus resultados e decidir sobre o melhor tratamento a ser ministrado ao paciente. Anualmente, mais de 5000 mil análises de resistência a drogas são realizadas. Além disso, a avaliação do perfil da resistência à droga no país é um importante apoio para a tomada de decisões logísticas de compra e distribuição de medicamentos.

5.1.3.4 Mapeamento de regras de resistência à droga para expressões da álgebra de processos.

O mapeamento de regras de resistência à droga para expressões da álgebra de processos introduz uma nova abordagem para o tratamento computacional dessas regras. Esse mapeamento permite uma verificação formal e validação das regras, principalmente no que se refere à identificação de relações implícitas entre as mutações.

Além do mais, a definição de ações finitas e independentes que compõem a execução das regras contribui com uma visão mais clara do funcionamento das regras e facilita a inclusão de novos parâmetros a serem avaliados pelas regras.

5.1.3.5 Gerador de algoritmos para análise de resistência à droga.

O gerador de algoritmos para análise de resistência ou *HIVdag* contribui com um ambiente para criação e teste de algoritmos de avaliação de resistência à droga. Com ele, o pesquisador pode criar e/ou alterar regras e testá-las rapidamente. Assim, o pesquisador pode simular o impacto de novas relações entre mutações e resistência a drogas, verificar a urgência da atualização dessas regras e atualizá-las sempre que desejado. A geração automática de regras não só aumenta a velocidade de seus testes e de atualizações dos algoritmos, como elimina a dependência dos programadores para sua manutenção.

5.1.4 Publicações.

Este trabalho produziu publicações relacionadas ao seu banco de dados e às ferramentas de análise, além de ter sido utilizado para avaliação de seqüências de uma publicação na área biológica.

Artigos publicados:

1. **Araújo LV, Soares M. A., Oliveira S.M., Chequer P., Tanuri A., Sabino E.C., Ferreira J.E.** (2006). DBCollHIV: a database system for colaborative HIV analysis in Brazil. Genet Mol Res.;5(1):203-15. PMID: 16755511.

   Menção Honrosa – Prêmio de Incentivo em Ciências e Tecnologia para o SUS 2006.

2. **Barreto CC, Nishyia A, Araújo LV, Ferreira JE, Busch MP, Sabino EC** (2005). Trends in Antiretroviral Drug Resistance and Clade Distributions among HIV-1 - Infected Blood Donors in São Paulo, Brazil. J Acquir Immune Defic Syndr.; 41(3):338-341. PMID: 16540943.

Artigo aceito para publicação:

1. **Araujo LV, Sabino EC, Ferreira JE** (2008). HIV drug resistance analysis tool based on process algebra. ACM SAC 2008, Fortaleza. **Proceedings of ACM** SAC-CAHC.

Artigo submetido:

1. **Araújo LV, Ferreira JE**, **Sanabani SS, Sabino EC,** Evaluation of HIVSetSubtype program for subtype classification of HIV-1 sequence.

Posters em congressos:

1. **Araújo LV, Ferreira JE**, **Sanabani SS, Sabino EC,** (2006) *Evaluation of Brazilian web based program for classification of HIV-1*- Scientific Poster – ISMB 2006, 14th. International Conference on Intelligent Systems for Molecular Biology (ISMB), Fortaleza, Ceará, Brazil.
2. **Araújo, L. V; Brindeiro, R. M; Oliveira, S. M.; Ferreira, J. E.; Sabino, E. C.**(2004), Software for Quality Control of HIV Sequences Obtained by PCR. ICOBICOBI - International Conference on Bioinformtics and Computational Biology
3. **Araújo, L.V., Sabino, E.C., Brindeiro, R.M., Tanuri, A., Ferreira, J.E.** and **Dantas, M.C.S.** (2004). *Bioinformatic tools for HIV-1 sequences analyses developed for the Brazilian STD/AIDS program network for genotype testing.* In MedGenMed, T., (ed.), *XV International AIDS Conference.* Thailand, Vol. 11, p. MoPeB3125.

4. **Brindeiro, R.M., Diaz, R.S., Sabino, E.C., Araújo, L.V., Levy, J.E.** and **Tanuri, A.** (2004). *Implementation of the quality control program (QC) for HIV-1 resistance genotyping testing network - RENAGENO of Brazilian STD/AIDS program.* In MedGenMed., (ed.), *The XV International AIDS Conference.* Thailand, eJIAS. 2004 Jul 11;1(1):MoPeB3126.

5. **Sabino EC, Araújo LV, Ferreira JE (2003)**, Multidimensional data analysis for HIV drug resistence, ICOBICOBI - International Conference on Bioinformtics and Computational Biology.

5.2 Futuras pesquisas

As seguintes futuras pesquisas estão previstas para expansão e aperfeiçoamento do ambiente do DBCollHIV :

- Criação de uma estrutura de data warehouse e recursos para realização de análises multidimensionais;
- Usar técnicas de regras associativas para identificar possíveis associações entre resistência a drogas e parâmetros como subtipos, resultados de exames laboratoriais e histórico de drogas usadas no tratamento do paciente;
- Uso da álgebra de processos para integração de novos módulos e ferramentas ao DBCollHIV.

Como funcionalidades extras para o ambiente, estão previstas:

- Estatística descritiva dos dados armazenados por projeto ou por usuário;
- Ambiente de comunicação e divulgação de pesquisas para facilitar o contato entre os usuários;
- Recursos para publicação dos dados em bancos de dados como o GenBank;
- Criação de módulo de importação de dados contidos em planilhas Excel;
- Entrada de dados off-line, com posterior envio dos dados para o banco de dados central;
- Gráficos com distribuição geográfica de subtipo, mutações e resistência à droga;
- Gráficos sobre comparativo entre drogas usadas, resultados de exames laboratoriais, mutações encontradas no vírus e resistência à droga;
- Integração de outros programas de subtipagem para que o usuário possa escolher os dois métodos/programas que serão utilizados para identificar o subtipo das seqüências.

Referências Bibliográficas

Achard, F., Vaysseix, G. and **Barillot, E.** (2001). XML, bioinformatics and data integration. *Bioinformatics* 17:115-125.

Alter, G., G. Hatzakis, et al. (2003). Longitudinal assessment of changes in HIV-specific effector activity in HIV-infected patients starting highly active antiretroviral therapy in primary infection. J Immunol 171(1): 477-88.

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. and **Lipman, D.J.** (1990). Basic local alignment search tool. *J Mol Biol* 215: 403-10.

Araujo LV, Soares M. A, Oliveira SM, Chequer P, Tanuri A, Sabino EC, Ferreira JE (2006). DBCollHIV: a database system for colaborative HIV analysis in Brazil. Genet Mol Res.;5(1):203-15. PMID: 16755511

Araújo, L.V., Sabino, E.C., Brindeiro, R.M., Tanuri, A., Ferreira, J.E. and **Dantas, M.C.S.** (2004). Bioinformatic tools for HIV-1 sequences analyses developed for the Brazilian STD/AIDS program network for genotype testing. In MedGenMed, T., (ed.), *XV International AIDS Conference,*. Thailand, Vol. 11, p. MoPeB3125.

Aslanzadeh, J. (2004). Preventing PCR amplification carryover contamination in a clinical laboratory. Ann Clin Lab Sci 34(4): 389-96.

Baeten JM, Chohan B, Lavreys L, Chohan V, McClelland RS, Certain L, Mandaliya K, Jaoko W, Overbaugh J. (2007) HIV-1 subtype D infection is associated with faster disease progression than subtype A in spite of similar plasma HIV-1 loads. J Infect Dis;195:1177-80.

Bakshi, S. S., S. Tetali, et al. (1995). "Repeatedly positive human immunodeficiency virus type 1 DNA polymerase chain reaction in human immunodeficiency virus-exposed seroreverting infants." Pediatr Infect Dis J 14(8): 658-62.

Baltimore, D. 1970, *RNA-dependent DNA polymerase in virions of RNA tumour viruses.* Nature. 226(5252): p. 1209-11

Barreira, D., Teixeira, P.R. and **Tanuri, A.** (2003). Brazilian Network for HIV Drug Resistance Surveillance (HIV-BResNet): a survey of chronically infected individuals. *Aids* 17: 1063-9.

Barrera, J., Cesar, R.M., Jr., Ferreira, J.E. and **Gubitoso, M.D**. (2004). An environment for knowledge discovery in biology. *Comput Biol Med* 34**:** 427-47.

Barreto CC, Nishyia A, Araújo LV, Ferreira JE, Busch MP, Sabino EC (2005). Trends in Antiretroviral Drug Resistance and Clade Distributions among HIV-1 - Infected Blood Donors in Sao Paulo, Brazil. J Acquir Immune Defic Syndr.;41(3):338-341. PMID: 16540943

Bergstra, J. A.; Ponse, A.; Smolka, S. A.(2001) Handbook of process algebra. Amsterdã: Elsevier Science Inc.

Best, E.; Devillers, R.; Koutny, K.(2001) A unified model for nets and process algebras. In: BERGSTRA, J. A.; PONSE, A.; SMOLKA, S.A. Handbook of process algebra. Amsterdã: Elsevier Science Inc.

Betts, B.J., and R.W. Shafer (2003) Algorithm specification interface for human immunodeficiency virus type 1 genotypic interpretation. J. Clin. Microbiol. 41:2792-2794.

Bradley Efron (1979) Bootstrap Methods: Another Look at the Jackknife. The Annals of Statistics 7 (1): 1-26

Braghetto, K. R.; Ferreira, J. E.; Pu, C., (2007) Using Control-Flow Patterns for Specifying Business Processes in Cooperative Environments. In: The 22nd Annual ACM Symposium on Applied Computing, Seoul. The 22nd Annual ACM Symposium on Applied Computing, 2007.v. 2. p.1234-1241.

Bryson, Y. J., S. Pang, et al. (1995). "Clearance of HIV infection in a perinatally infected infant." N Engl J Med 332(13): 833-8.

Brindeiro, R.M., Diaz, R.S., Sabino, E.C., Araújo, L.V., Levy, J.E. and **Tanuri, A.** (2004). Implementation of the quality control program (QC) for HIV-1 resistance genotyping testing network - RENAGENO of Brazilian STD/AIDS program. In MedGenMed., (ed.), *The XV International AIDS Conference.* Thailand, eJIAS. 2004 Jul 11;1(1):MoPeB3126

Brindeiro, R.M., Diaz, R.S., Sabino, E.C., Morgado, M.G., Pires, I.L., Brigido, L., Dantas, M.C., Barreira, D., Teixeira, P.R. and **Tanuri, A.** (2003). Brazilian Network for HIV Drug Resistance Surveillance (HIV-BResNet): a survey of chronically infected individuals. *Aids* 17**:** 1063-9

CDC.(1993) 1993 revised classification system for HIV infection and expanded surveillance case definition for AIDS among adolescents and adults, Centers for Disease Control. Disponível em: (http://www.cdc.gov/MMWR/preview/ MMWRhtml/00018871.htm). Acesso em:25/02/2008

DAFTG - Database Architecture Framework Task Group (1986). "Reference model for DBMS standardization." SIGMOD Rec. 15(1): 19-58.

Dalgleish, A.G., et al. (1984) The CD4 (T4) antigen is an essential component of the receptor for the AIDS retrovirus. Nature. 312(5996): p. 763-7

de Oliveira, T., K. Deforche, et al. (2005). "An automated genotyping system for analysis of HIV-1 and other microbial sequences." Bioinformatics 21(19): 3797-800.

Deeks, S. G. (2003). Treatment of antiretroviral-drug-resistant HIV-1 infection. Lancet 362(9400): 2002-11.

Durbin,R., Eddy,S., Krogh,A., Mitchison G., (1998) Biological sequence analysis: probabilistic models of proteins and nucleic acids,1998, Cambridge University Press

Eddy SR (1998) Profile hidden Markov models. Bioinformatics, 14:755-763.

Ferreira, J.E. and **Busichia, G**. (1999). Database modularization design for the construction of flexible information systems. *Proceedings IEEE for the IDEAS99.* Montreal -Canada, pp. 415-422.

Ferreira J.E., **Takai** O.K., **Braghetto,** K.R., **Pu C.** Large Scale Order Processing through Navigation Plan Concept - accepted to SCC2006. In: IEEE International Conference on Services Computing (SCC 2006), 2006, Chicago.

Fokkink , W. J. (2000) Introduction to Process Algebra**:** Texts in Theoretical Computer Science. Berlin: Springer-Verlag New York, Inc., 163 p

Frenkel, L. M., Mullins, J. I., Learn, G. H., Manns-Arcuino, L., Herring, B. L., Kalish, M. L.,Steketee, R. W., Thea, D. M., Nichols, J. E., Liu, S. L., Harmache, A., He, X., Muthui, D.,Madan, A., Hood, L., Haase, A. T., Zupancic, M., Staskus, K., Wolinsky, S., Krogstad, P.,Zhao, J., Chen, I., Koup, R., Ho, D., Roberts, N. J., Jr., and et al. (1998) Genetic evaluation of suspected cases of transient HIV-1 infection of infants. Science 280, 1073-7.

George, J.R. and Schochetman G. Detection of HIV infection using serological techniques., in AIDS testing: a compreensive guide to technical, medical, social, legal

and management issues, G. Schochetman and J.R. George, Editors. 1994, Springer Verlag: New York. p. 62-102.

Gifford, R., de Oliveira, T., Rambaut, A., Myers, R.E., Gale, C.V., Dunn, D., Shafer, R., Vandamme, A.M., Kellam, P. and Pillay, D. (2006) Assessment of automated genotyping protocols as tools for surveillance of HIV-1 genetic diversity, Aids, 20 1521-9.

Gordon, M., De Oliveira, T., Bishop, K., Coovadia, H.M., Madurai, L., Engelbrecht, S., Janse van Rensburg, E., Mosam, A., Smith, A. and Cassol, S., (2003)Molecular characteristics of human immunodeficiency virus type 1 subtype C viruses from KwaZulu-Natal, South Africa: implications for vaccine and antiretroviral control strategies, J Virol, 77 2587-99.

Greene, W., The molecular biology of human immunodeficiency virus type 1 infection, 1991 New England Journal of Medicine, vol. 324, pp. 308-17.

Gürtler L. Difficulties and strategies of HIV diagnosis. Lancet 1996; 348:176-9

Hammer, S. M., D. A. Katzenstein, et al. (1996). A trial comparing nucleoside monotherapy with combination therapy in HIV-infected adults with CD4 cell counts from 200 to 500 per cubic millimeter. AIDS Clinical Trials Group Study 175 Study Team. N Engl J Med 335(15): 1081-90.

Hartley, J. L., A. Rashtchian (1993). "Dealing with contamination: enzymatic control of carryover contamination in PCR." PCR Methods Appl. 3(2): S10-14.

Hoare, C. A. R. (1978) Communicating sequential processes. Communications of the ACM, v. 21, n. 8, p. 666-677.

Holland, J.J., De La Torre, J.C. and Steinhauer, D.A. (1992) *RNA virus populations as quasispecies.* Curr Top Microbiol Immunol. **176**: p. 1-20.

J.M. Coffin. (1979). Structure, replication, and recombination of retrovirus genomes: Some unifying hypotheses. J. Gen. Virol. 42: 1-26.

J.M. Coffin. (1992a). Structure and classification of retroviruses. In The retroviridae (ed. J.A. Levy), pp. 19–49. Plenum Press, New York.

J.M. Coffin. (1992b). Genetic diversity and evolution of retroviruses. Curr. Top. Microbiol. Immunol. 176: 143-164.

J.M. Coffin. (1996). Retroviridae and their replication. In Virology (ed. B.N. Fields et al.), pp. 1767–1848. Raven Press, New York.

Kalmar, **E.M.N.**, **Chen. S.**, **Ferreira**, **S.**, **Barreto**, **C.C.**, **McFarlan, W.**, **Sabino**, **E,C**,(2005). Drug resistance among HIV patients who discontinued ARV treatment in Brazil, *3rd International AIDS Society Conference on HIV Pathogenesis and Treatment*, WePe 4.4.C13.

Kanki P.J., Donald J. Hamel, Jean-Louis Sankalé, Chung-cheng Hsieh, Ibou Thior, Francis Barin, Stephen A. Woodcock, Aïssatou Guèye-Ndiaye, Er Zhang, Monty Montano, Tidiane Siby, Richard Marlink, Ibrahima NDoye, Myron E. Essex, and Souleymane MBoup (1999) Human Immunodeficiency Virus Type 1 Subtypes Differ in Disease Progression, Journal of Infectious Diseases Volume 179 Number 1.

B. T. Korber, B. F. Foley, C. I. Kuiken, S. K. Pillai, and J. G. Sodroski, (1998) "Numbering Positions in HIV Relative to HXB2CG," in Human Retroviruses and AIDS. Report LA-UR 99-1704, B. T. Korber et. al., Ed. Los Alamos, NM: Los Alamos National Laboratory, pp. III-102;III-111.

Kuiken, C., Korber, B. and **Shafer, R.W**. (2003). HIV sequence databases. *AIDS Rev*, **5,** 52-61.

Kwok, S. and R. Higuchi (1989). "Avoiding false positives with PCR." Nature 339(6221): 237-238.

Laeyendecker O, Li X, Arroyo M, McCutchan F et al.(2006) The Effect of HIV Subtype on Rapid Disease Progression in Rakai, 13th Conference on Retroviruses and Opportunistic Infections, Uganda,(abstract no. 44LB),

Lamers S, Beason S, Dunlap L, Compton R, Salemi M.(2004) HIVbase: a PC/Windows-based software offering storage and querying power for locally held HIV-1 genetic, experimental and clinical data.Bioinformatics;20(3):436-8.

Learn, G. H., Jr, Korber, B. T., Foley, B., Hahn, B. H., Wolinsky, S. M., and Mullins, J. I. (1996) Maintaining the integrity of human immunodeficiency virus sequence databases. J.Virol. 70, 5720-5730.

Lee, Thomas and Pouliot, Yannick and Wagner, Valerie and Gupta, Priyanka and Stringer-Calvert, David and Tenenbaum, Jessica and Karp, Peter (2006).

BioWarehouse: a bioinformatics database warehouse toolkit, BMC Bioinformatics, 7(1) 170. PubMedID:16556315.

Liu TF, Shafer RW (2006). Web Resources for HIV type 1 Genotypic-Resistance Test Interpretation. Clin Infect Dis 42(11):1608-18. Epub 2006

Lole KS, Bollinger RC, Paranjape RS, Gadkari D, Kulkarni SS, Novak NG, Ingersoll R, Sheppard HW, Ray SC.(1999) Full-length human immunodeficiency virus type 1 genomes from subtype C-infected seroconverters in India, with evidence of intersubtype recombination. J Virol;73(1):152-60.

Maurer, Michael and Molidor, Robert and Sturn, Alexander and Hartler, Juergen and Hackl, Hubert and Stocker, Gernot and Prokesch, Andreas and Scheideler, Marcel and Trajanoski, Zlatko (2005), MARS: Microarray analysis, retrieval, and storage system, BMC Bioinformatics, 6(1)170, PubMedID:15836795.

Martin, DP, Williamson C, Posada, D. (2005). RDP2: Recombination detection and analysis from sequence alignments. Bioinformatics 21: 260-262.

Meynard, J. L., M. Vray, L. Morand-Joubert, et al. (2002) Phenotypic or genotypic resistance testing for choosing antiretroviral therapy after treatment failure: a randomized trial. AIDS 16:727-736.

Milner, R. A (1982) Calculus of Communicating Systems. Secaucus: Springer-Verlag New York, Inc., 260 p.

Morgado, M.G., Sabino, E.C., Shpaer, E.G., Bongertz, V., Brigido, L., Guimaraes, M.D., Castilho, E.A., Galvao-Castro, B., Mullins, J.I., Hendry, R.M. and et al., (1994) V3 region polymorphisms in HIV-1 from Brazil: prevalence of subtype B strains divergent from North American/European prototype and detection of subtype F, AIDS Res Hum Retroviruses, 10 569-76.

Mullis, K., F. Faloona, et al. (1986). Specific enzymatic amplification of DNA in vitro: the polymerase chain reaction. Cold Spring Harb Symp Quant Biol 51 Pt 1: 263-73.

Murata, T. (1989) Petri Nets: Properties, Analysis and Applications. Proceedings of The IEEE, v. 77, n. 4, p. 541-580.

Myers RE, Gale CV, Harrison A, Takeuchi Y, Kellam P. (2005) A statistical model for HIV-1 sequence classification using the subtype analyser (STAR). Bioinformatics; 21:3535–3540.

Oikawa, M. K., Broinizi, M. E., Dermagos, A., Armelin, H. A., Ferreira, J. E. (2004), Genflow: generic flow for integration, management and analysis of molecular biology data. *Journal of Genetics and Molecular Research*. Special edition for Icobicobi, 27(4): 690-697.

Ortiz, G. M., J. Hu, et al. (2002). Residual viral replication during antiretroviral therapy boosts human immunodeficiency virus type 1-specific CD8+ T-cell responses in subjects treated early after infection. J Virol 76(1): 411-5

Oxenius, A., D. A. Price, et al. (2000). Early highly active antiretroviral therapy for acute HIV-1 infection preserves immune function of CD8+ and CD4+ T lymphocytes. Proc Natl Acad Sci U S A 97(7): 3382-7.

Özsu, M. T., Valduriez P.,(1999), Principles of Distributed Database Systems, Second Edition, Prentice Hall, ISBN 0-13-659707-6

Petropoulos, C. J., N. T. Parkin, et al. (2000). "A novel phenotypic drug susceptibility assay for human immunodeficiency virus type 1." Antimicrob Agents Chemother 44(4): 920-8.

Posada D and Crandall KA. (2002). The effect of recombination on the accuracy of phylogeny reconstruction. Journal of Molecular Evolution 54: 396-402

Posada D and Crandall KA. (2001). Evaluation of methods for detecting recombination from DNA sequences: computer simulations. Proceedings of the National Academy of Sciences , 13751-13756.

Posada D, Crandall KA and Holmes EC. (2002). Recombination in evolutionary genomics. Annual Review of Genetics 36: 75-97.

Preston, B.D., B.J. Poiesz, and L.A. Loeb. (1988) *Fidelity of HIV-1 reverse transcriptase.* Science. 242(4882): p. 1168-71

Proffitt, M.R. and Yen-Lieberman B., Laboratory diagnosis of human immunodeficiency virus infection. Infect Dis Clin North Am, 1993. 7(2): p. 203-19.

Ravela J, Betts BJ, Brun-Vezinet F, et al. (2003) HIV-1 protease and reverse transcriptase mutation patterns responsible for discordances between genotypic drug resistance interpretation algorithms. J Acquir Immune Defic Syndr;33:8-14

Robertson, D.L., Anderson, J.P., Bradac, J.A., Carr, J.K., Foley, B., Funkhouser, R.K., Gao, F., Hahn, B.H., Kalish, M.L., Kuiken, C., Learn, G.H., Leitner, T., McCutchan, F., Osmanov, S., Peeters, M., Pieniazek, D., Salminen, M., Sharp, P.M., Wolinsky, S. and Korber, B., (2000) HIV-1 nomenclature proposal, Science, 288 55-6.

Sabino, E.C., Shpaer, E.G., Morgado, M.G., Korber, B.T., Diaz, R.S., Bongertz, V., Cavalcante, S., Galvao-Castro, B., Mullins, J.I. and Mayer, A., (1994) Identification of human immunodeficiency virus type 1 envelope genes recombinant between subtypes B and F in two epidemiologically linked individuals from Brazil, J Virol, 68 6340-6.

Sanger, F., Nicklen, S. and Coulson, A. R.,(1977) DNA sequencing with chain-terminating inhibitors, PNAS (Proc Natl Acad Sci) U S A.; 74(12): 5463–5467.

Saravolatz, L. D., D. L. Winslow, et al. (1996). Zidovudine alone or in combination with didanosine or zalcitabine in HIV-infected patients with the acquired immunodeficiency syndrome or fewer than 200 CD4 cells per cubic millimeter. Investigators for the Terry Beirn Community Programs for Clinical Research on AIDS. N Engl J Med 335(15): 1099-106.

Schultz, A.K., Zhang, M., Leitner, T., Kuiken, C., Korber, B., Morgenstern, B. and Stanke, M. (2006) A jumping profile Hidden Markov Model and applications to recombination sites in HIV and HCV genomes, BMC Bioinformatics, 7 265.

Sanabani, S., Neto, W.K., de Sa Filho, D.J., Diaz, R.S., Munerato, P., Janini, L.M. and Sabino, E.C. (2006a) Full-length genome analysis of human immunodeficiency virus type 1 subtype C in Brazil, AIDS Res Hum Retroviruses, 22 171-6.

Sanabani, S., Neto, W.K, Kalmar, E.M., Diaz, R.S., Janini, L.M. and Sabino, E.C. (2006b) Analysis of the near full length genomes of HIV-1 subtypes B, F and BF recombinant from a cohort of 14 patients in Sao Paulo, Brazil, Infect Genet Evol, 6 368-77.

Schuurman, R., Demeter, L., Reichelderfer, P., Tijnagel, J., de Groot, T., and Boucher, C.(1999b) Worldwide evaluation of DNA sequencing approaches for

identification of drug resistance mutations in the human immunodeficiency virus type 1 reverse transcriptase. J.Clin.Microbiol. 37, 2291-2296.

Shafer RW, Stevenson D, Chan B. (1999) Human immunodeficiency virus reverse transcriptase and protease sequence database. Nucleic Acids Res.;27:348- 352.

Shafer, R.W. (2002). Genotypic testing for human immunodeficiency virus type 1 drug resistance. *Clin Microbiol Rev* 15**:** 247-77.

Shafer RW. (2006) Rationale and Uses of a Public HIV Drug-Resistance Database. Journal of Infectious Diseases 194 Suppl 1:S51-8.2006.

Siepel, A.C., Halpern, A.L., Macken, C. and **Korber, B.T**. (1995). A computer program designed to screen rapidly for HIV type 1 intersubtype recombinant sequences. *AIDS Res Hum Retroviruses* 11**:** 1413-6.

Simon Fiddy , David Cattermole , Dong Xie , Xiao Yuan Duan and Richard Mott (2006), An integrated system for genetic analysis, BMC Bioinformatics, 7:210, PubMedID:15723693.

Soares, E. A., A. F. Santos, et al. (2007). Differential drug resistance acquisition in HIV-1 of subtypes B and C. PLoS ONE 2(1): e730.

Sohrab P Shah , Yong Huang , Tao Xu , Macaire MS Yuen , John Ling and BF Francis Ouellette (2005), Atlas – a data warehouse for integrative bioinformatics, BMC Bioinformatics, 6:34

Sommerville I. (2006), Software Engineering, 8 Ed., Addison Wesley.

Temin, H.M. and S. Mizutani (1970), *RNA-dependent DNA polymerase in virions of Rous sarcoma virus.* Nature. 226 (5252): p. 1211-3.

Van Laethem, K., A. De Luca, A. Antinori, A. Cingolani, C. F. Perno, and A.-M. Vandamme (2002) A genotypic drug resistance algorithm that significantly predicts therapy response in HIV-1 infected patients. Antivir.Ther. 7:123-129

Wainberg, Mark A. (2004), HIV-1 subtype distribution and the problem of drug resistance, AIDS 2004 Volume 18 Supplement 3

Wensing, A.M.J., Boucher, C.A.B (2003) Worldwide Transmission of Drug-resistant HIV. AIDS Review;5:140-155

Watson, J.D, Baker, T. A., Stephen Bell P., Gann, A., Levine M., Losic R.(2003) Molecular Biology of the Gene, 5ª Ed., Benjamin Cummings Publishing Company.

WHO. (2005) Intrem WHO Clinical Staging of HIV/AIDS and HIV/AIDS Case Definitions for Surveillance, World Health Organization.
Disponível em (http://www.who.int/hiv/pub/guidelines/casedefinitions/en/index.html) Acesso em: 25/02/2008

Zazzi, M.; Romano, L.; Venturi,G.; Shafer,R.; Reid, C.; Dal Bello,F.; Parolin,C; Palù, G.; Valensin, P (2004) Comparative evaluation of three computerized algorithms for prediction of antiretroviral susceptibility from HIV type 1 genotype. Journal of Antimicrobial Chemotherapy 53, 356-360 DOI: 10.1093/jac/dkh021