Universidade de São Paulo
Instituto de Matemática e Estatística
Bacharelado em Ciência da Computação

Douglas Vasconcelos Cancherini

Desenvolvimento de um configurador gráfico para o ToPS

São Paulo
Janeiro de 2016

Desenvolvimento de um configurador gráfico para o ToPS

Monografia final da disciplina
MAC0499 – Trabalho de Formatura Supervisionado.

Supervisor: Prof. Dr. Alan Mitchell Durham

São Paulo
Janeiro de 2016

Resumo

Previamente desenvolvido pelo grupo, o ToPS (*Toolkit for Probabilistic Models of Sequences*) implementa com flexibilidade vários algoritmos eficientes para treinamento, decodificação e simulação de uma série de modelos probabilísticos com emissão de sequência de símbolos. Ele permite que estes modelos sejam combinados em um modelo oculto de Markov generalizado (GHMM), o que torna possível usá-lo para construir preditores gênicos de notável acurácia, como é o caso do MYOP (*Make Your Own Predictor*, também desenvolvido pelo grupo). Entretanto, a grande flexibilidade oferecida pelo ToPS na configuração e combinação de modelos, na prática, tem sido limitada pela dificuldade de um usuário editar seus numerosos e interdependentes arquivos de configuração. Este trabalho de formatura consistiu no desenvolvimento de um sistema cliente-servidor que facilitasse estas tarefas. O sistema cliente, desenvolvido para navegador de internet, proporciona ao seu usuário a visualização e a modificação da estrutura de submodelos de um GHMM para o ToPS, bem como a edição dos parâmetros de configuração de cada submodelo. No cliente, cada submodelo é representado como um polígono, com o tipo de polígono correspondendo ao tipo de modelo probabilístico. Ao redor de cada polígono existe um círculo ou elipse, que representa a duração do submodelo, ou seja, o comprimento da subsequência de símbolos emitida pelo submodelo, a qual também pode ser descrita por um modelo probabilístico. As transições entre submodelos são representadas por arestas com setas entre polígonos. A interação do usuário com estes elementos gráficos lança mão de ações intuitivas, seja para edição de parâmetros, seja para a disposição dos elementos na tela. Cores são usadas para destacar o compartilhamento de parâmetros entre submodelos ou especificações de duração. Além da cópia simples da especificação de submodelo, é possível uma cópia por referência, que torna a especificação compartilhada entre vários submodelos. Além da possibilidade de editar manualmente uma especificação de submodelo, o sistema admite obter uma especificação por meio do treinamento do submodelo, caso no qual o usuário é orientado a, de acordo com o tipo de modelo estatístico e o tipo de algoritmo de treinamento, escolher os parâmetros obrigatórios e opcionais para o treinamento, conforme o que é esperado pelo ToPS. Um arquivo de configuração do sistema permite modificar o conjunto destes parâmetros, facilitando a adaptação a modificações futuras do ToPS. O sistema servidor, por sua vez, usa as tecnologias Express e Node.js e deve residir na mesma máquina em que o sistema ToPS tiver sido instalado. Ele disponibiliza ao usuário a possibilidade de invocar o ToPS através da Internet para realizar, com a configuração escolhida, treinamento de submodelos e modelos de duração, bem como

decodificação de sequências, com base na completa especificação do GHMM.
Palavras-chave: predição gênica, configurador gráfico, modelos ocultos de Markov.

Abstract

Our research group previously developed ToPS (Toolkit for Probabilistic Models of Sequences), a flexible software that implements efficient algorithms for training, decoding and simulating a series of probabilistic models with symbol sequence emission. It allows these models to be combined in a GHMM (generalized hidden Markov model), what makes possible their use to construct very accurate gene predictors, like MYOP (Make Your Own Predictor, a previous creation of our group). However, considerable flexibility that ToPS offers concerning configuration and combination of models has been scarcely exploited by most users, because of numerous and interdependent configuration files are required by ToPS. This work has consisted in the development of a client-server system to make easier the tasks related to ToPS configuration. The client system for internet browser guides users in visualizing and modifying submodel structure of a GHMM for ToPS, as well as in editing configuration parameters of each submodel. The client system represents a submodel as a polygon, with polygon type corresponding to probabilistic model type. Around each polygon lies a circle or ellipse, representing submodel duration, that is, length of symbol subsequence emitted by submodel, which can also be given by a probabilistic model. Transitions between submodels are represented by arrowed edges between polygons. User interaction with these graphical elements resorts to intuitive actions, for parameter edition or element arrangement on the canvas. Colors are used in order to highlight parameter sharing among submodels or duration specifications. Besides being possible to manually edit a submodel specification, the system provides the means for obtaining one by model training. In this case, the user is guided to, according to statistical model and training algoritm, choose mandatory and optional parameters for training, exactly as it is expected by ToPS. A system configuration file allows modifying the parameter set, enabling adaptation to future modification in ToPS. Concerning the server system, it was developed using Express and node.js, and should reside in the same machine where ToPS has been installed. It offers users the possibility of invoking ToPS across the Internet for running, with the chosen configuration, submodel training and duration model training, as well as decoding of sequences based on the complete GHMM specification.

Keywords: gene prediction, graphical configuration editor, hidden Markov models.

Sumário

Capítulo 1

Introdução

Modelos ocultos de Markov (*hidden Markov models* ou HMMs) e suas extensões são apropriados para a modelagem de numerosos fenômenos, especialmente em reconhecimento de padrões no tempo, o que inclui análises de sinais de áudio, vídeo e de sistemas elétricos. Em bioinformática, devido à sua capacidade de reconhecer e modelar correlações entre símbolos próximos, eles encontram muitas aplicações relacionadas ao estudo de sequências de ácidos nucleicos e aminoácidos. Por exemplo, mostram-se adequados para reconhecimento de domínios proteicos, avaliação de erros de sequenciamento, alinhamento de duas ou múltiplas sequências e predição de estrutura secundária de proteína e RNA Yoon (2009).

O problema da predição gênica é mais um no qual tais modelos probabilísticos de longa data se mostraram muito úteis. De fato, boa parte dos modernos preditores gênicos tem sua estrutura, no nível mais alto, baseada em um modelo oculto de Markov generalizado (*generalized hidden Markov model* ou GHMM). Além disso, comumente outras extensões dos HMMs são usadas, nestes mesmos preditores gênicos, como submodelos do GHMM, para descrever o comportamento de subsequências correspondentes a elementos que, estrutural e funcionalmente, compõem os genes. Exemplos destes elementos são éxons e sítios doadores ou aceptores de íntrons.

O grupo de pesquisa do supervisor deste trabalho desenvolveu previamente o ToPS (*Toolkit for Probabilistic Models of Sequence*), um conjunto de ferramentas que explora muitas das capacidades que acabamos de descrever para estes modelos. O ToPS disponibiliza algoritmos eficientes para decodificação, treinamento e simulação de HMMs e várias de suas extensões. Ele oferece considerável flexibilidade na escolha de algoritmos e parâmetros de treinamento. Por meio de um GHMM, ele permite que vários modelos diferentes sejam combinados, com integração automática das configurações e algoritmos, sem que seja necessário escrever qualquer código. Todas essas virtudes foram muito apropriadamente exploradas na construção do MYOP (*Make Your Own Predictor*), uma ferramenta que facilita ao usuário a tarefa de, usando o ToPS, construir preditores gênicos eficientes e de elevada acurácia.

A motivação deste trabalho foi a percepção de que a flexibilidade oferecida pelo ToPS vinha sendo desperdiçada. Para uma tarefa maior, como justamente a construção de um preditor gênico, é necessário especificar completamente um GHMM com várias dezenas de submodelos. A própria compreensão da estrutura do GHMM tendia a ficar prejudicada nesta situação. Além disso, a especificação de cada submodelo envolvia a edição de um arquivo, que podia ser um arquivo propriamente dito de especificação do modelo, ou um arquivo contendo apenas seus parâmetros de treinamento, caso em que a especificação seria gerada automaticamente pelo treinamento. Em ambos os casos, tratava-se de uma tarefa tediosa, complicada pela dificuldade em verificar quais modelos compartilhavam, por exemplo, parâmetros de treinamento.

O objetivo do presente trabalho foi permitir a usuários do ToPS e do MYOP em uso de um modelo baseado em GHMM:

- compreender melhor a arquitetura do GHMM, bem como obter visualizações convenientes de tal arquitetura e ter a possibilidade de modificá-la;
- poder com facilidade editar os parâmetros dos submodelos do GHMM e/ou os parâmetros para seu treinamento, bem como os parâmetros dos modelos de duração correspondentes.

A proposta do trabalho foi o desenvolvimento de um sistema com arquitetura cliente-servidor. Tal sistema permitiria a um usuário em qualquer navegador de internet moderno executar um cliente, cuja interface gráfica permitiria-lhe realizar as tarefas de visualização e edição que descrevemos nos objetivos. Por outro lado, um sistema servidor em execução numa máquina com o ToPS instalado se comunicaria com o cliente para disponibilizar a ele a possibilidade de, com as configurações presentemente escolhidas para o ToPS, invocar o ToPS para realizar treinamento e simulação de modelos, bem como predição gênica.

Descrevemos a seguir a estrutura do texto desta monografia.

O capítulo *O problema da predição gênica em eucariotos* descreve um pouco dos elementos biológicos que constituem os genes eucarióticos e a partir dos mesmos apresenta uma das principais motivações de fundo para o desenvolvimento do ToPS, a saber, o problema da predição gênica. A seguir discorre-se um pouco sobre os sistemas computacionais de predição gênica.

O capítulo *Predição gênica* ab initio *baseada em modelos de Markov ocultos generalizados* apresenta inicialmente os modelos ocultos de Markov e os problemas da decodificação e do treinamento. A seguir apresenta uma série de extensões dos modelos de Markov, como as cadeias de Markov de ordem k, as cadeias de Markov de alcance variável, as cadeias ocultas de Markov inomogêneas e os GHMMs. Ao longo do capítulo, frequentemente é abordada também a relação destes modelos com o problema da predição gênica. Apresenta-se o software desenvolvido pelo grupo, o ToPS.

O capítulo *Desenvolvimento de um configurador gráfico para o ToPS* apresenta o desenvolvimento propriamente dito do sistema configurador gráfico. Detalha-se em especial as particularidades da interface que ajudam o configurador a realmente facilitar o usuário na tarefa de modificar configurações do ToPS.

O capítulo *Conclusões e perspectivas futuras* tem título suficientemente auto-explicativo.

Capítulo 2

O problema da predição gênica em eucariotos

Genes são regiões do DNA genômico que são transcritas em moléculas funcionais de RNA. Como ilustrado na Figura 2.1, sua estrutura frequentemente é complexa, especialmente em eucariotos. Um exemplo importante desta complexidade é a presença de múltiplas regiões do RNA, os íntrons, que são removidas durante seu processamento, denominando-se éxons as regiões que não são removidas e estão presentes no RNA maduro. Conhecida a sequência genômica, a precisa determinação das várias regiões funcionais de um gene, no chamado processo de anotação, é crucial para prever, entre outros aspectos, quais proteínas podem ser derivadas do gene, como ocorre a regulação de sua expressão e quais efeitos podem ter mutações com determinada localização. A anotação gênica é, deste modo, um problema central em genômica.

Diversos sinais biológicos presentes na sequência de DNA marcam as várias regiões do gene, com destaque para as regiões de início e fim de transcrição, início de fim de tradução, e as bordas éxon/íntron. Algoritmos de predição de genes ditos *ab initio* ou intrínsecos são aqueles que reconhecem estes sinais na sequência genômica e, com base apenas na mesma, preveem o número e os limites dos genes existentes. Comumente, estes algoritmos precisam ser treinados por sequências de genes conhecidos de outros organismos, dando-se preferência aqui ao uso de espécies próximas, pois os sinais biológicos nos genes tendem a ser também mais similares. Trata-se de um problema difícil: os métodos existentes têm alta taxa de sucesso em detectar a presença de um gene e/ou sua localização grosseira, mas são bastante falhos quando se deseja uma anotação precisa ao longo de todo o gene, incluindo todos os limites entre íntrons e éxons. Para ter-se uma ideia de como os preditores gênicos estão longe de incluir toda a riqueza do fenômeno biológico, é pertinente comentar o fato de que uma série de hipóteses simplificadoras são assumidas por todos eles. Exemplos comuns destas hipóteses simplificadoras incluem: em cada fita do DNA, inexistência de genes sobrepostos, aninhados, ou com sequência genômica incompleta; ausência de erros (experimentais) ou variações (populacionais) nas sequências genômicas; algum limite ao comprimento máximo do gene e de seus íntrons; não ocorrência do códon de iniciação (ou de parada) em éxons diferentes; suposição de que os íntrons de um determinado gene são sempre excisados do mesmo modo e/ou ausência de sítios doadores e aceptores de íntrons com sequências exóticas; ausência de códons TGA que não codifiquem parada de tradução, mas o aminoácido raro selenocisteína Majoros (2007).

Preditores de genes habitualmente são avaliados em função de medidas como sensibilidade, especificidade e acurácia. Após o treinamento do programa, um conjunto de teste cuja anotação manual seja confiável é submetido ao mesmo. A sensibilidade, que avalia a

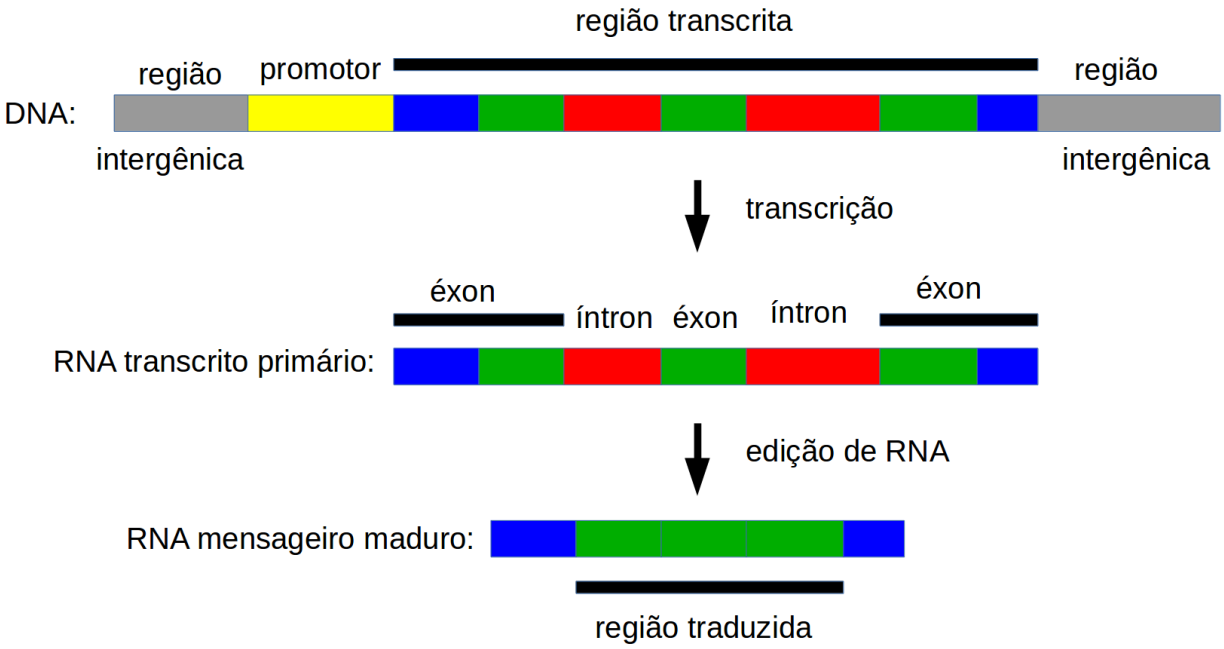

Figura 2.1: ***Representação esquemática da estrutura de um gene eucariótico.*** *Notar como nem toda a região transcrita tem aminoácidos correspondentes na proteína resultante. Trechos do RNA transcrito, os íntrons (em vermelho), são removidos e degradados. O RNA maduro é obtido por meio da junção das regiões vizinhas não removidas, os éxons, pela ação enzimática dos mesmos complexos nucleoproteicos que fazem a remoção dos íntrons. Além disso, a região cuja sequência é traduzida a proteína (em verde) é uma subsequência da sequência do RNA mensageiro maduro, havendo regiões não traduzidas a montante e a jusante da mesma (em azul). A determinação biológica dos pontos em que iniciam e terminam os processos de transcrição e tradução, bem como das fronteiras entre íntrons e éxons, é dada por uma série de sinais presentes na própria sequência da região transcrita e ao redor na mesma (como é o caso do promotor, em amarelo, que influencia o ponto de início da transcrição), sinais estes que os HMMs e suas extensões têm se mostrado bastante eficientes em reconhecer.*

capacidade do programa de reconhecer apropriadamente os objetos que procura, é avaliada dividindo-se o número de verdadeiros positivos pela soma dos mesmos ao número de falsos negativos. A especificidade, que avalia a porcentagem de acertos entre os objetos reconhecidos, é calculada dividindo-se o número de verdadeiros positivos pela sua soma ao número de falsos positivos. A acurácia, por sua vez, calcula-se pela divisão da soma dos verdadeiros positivos e negativos pelo total de objetos avaliados. Estas medidas podem ser aplicadas ao reconhecimento de nucleotídeos, de éxons ou de genes como um todo.

Além dos métodos *ab initio*, outra abordagem natural do problema de anotação de genes, dita extrínseca, consiste em obter não somente o genoma, mas também as sequências de RNA derivadas de sua transcrição, o chamado transcriptoma do organismo. Neste sentido, foram criados programas para mapear a sequência de fragmentos de RNA à sequência do DNA genômico, e com isso caracterizar a estrutura de íntrons e éxons dos genes. Esta abordagem é atualmente facilitada pelo grande número de organismos para os quais há genoma e transcriptoma conhecidos.

Por fim, existem preditores gênicos que reúnem metodologias usadas em preditores *ab initio* à metologia extrínseca, obtendo como resultado as mais confiáveis predições gênicas disponíveis atualmente. Uma situação comum na qual as metodologias baseadas em múltiplas informações beneficiam os preditores *ab initio* ocorre, descrita a seguir, quando se tem duas espécies próximas. Para uma das espécies, conhece-se o transcritoma, ou seja, o conjunto de todas as moléculas de RNA mensageiro que a mesma produz. Conseguem-se assim predições gênicas altamente confiáveis por meio dos métodos baseados em múltiplas informações. Já para a outra espécie, possui-se apenas o genoma. É comumente possível, num caso assim, obter-se uma predição razoavelmente confiável dos genes da segunda espécie, por meio de um preditor *ab initio* que tenha sido treinado por meio dos genes preditos na primeira espécie através de algum método baseado em múltiplas informações.

Capítulo 3

Predição gênica *ab initio* baseada em modelos ocultos de Markov generalizados

3.1 Modelos ocultos de Markov e alguns modelos derivados

3.1.1 Modelos ocultos de Markov

Várias definições similares e equivalentes são possíveis para modelo oculto de Markov. Adotaremos que modelo oculto de Markov é um modelo estocástico definido pelos seguintes elementos:

- um conjunto Q de estados, $Q = \{q_0, q_1, ..., q_{n-1}\}$, dentre os quais dois estados especiais: um estado inicial q^i e um estado final q^f;
- uma matriz T de probabilidades de transição entre estados, tal que T_{ij} é igual à probabilidade do sistema, estando no estado q_i, fazer logo a seguir a transição para o estado q_j;
- um alfabeto α, isto é, um conjunto de símbolos que o modelo pode emitir, $\alpha = \{\alpha_0, \alpha_1, ..., \alpha_{m-1}\}$;
- uma matriz E de probabilidades de emissão dos símbolos do alfabeto, tal que E_{ij} é igual à probabilidade do sistema, estando no estado q_i, emitir o símbolo α_j.

O modelo em questão sempre se inicia no estado q^i, ao qual nunca pode retornar. A cada passo de seu funcionamento antes de chegar ao estado q^f, ele emite um símbolo do alfabeto e faz uma transição de estado. A transição ao estado q^f ocorre uma única vez, sendo que então o sistema cessa sua operação, não ocorrendo mais transições ou emissões. Deste modo, reunindo-se na ordem o conjunto de emissões realizadas, temos que o sistema, enquanto funciona, emite uma cadeia de símbolos. Mais que isso, assume-se que o funcionamento do sistema é opaco, de modo a não ser possível ao observador determinar quais transições de estado ocorreram. Apenas a cadeia de símbolos emitidos pode ser observada.

O problema de decodificação em modelo oculto de Markov corresponde a determinar, dada uma sequência de símbolos emitida por um modelo com parâmetros conhecidos, qual a sequência de estados que o modelo percorreu. Frequentemente, os modelos são tais que existe um grande número de sequências de estados que poderiam resultar na emissão da mesma sequência de símbolos. Interessa, neste caso, qual a probabilidade de ocorrência de cada uma destas sequências de estados. O algoritmo de Viterbi usa programação dinâmica para

calcular sucessivamente qual a sequência mais provável de estados que resulta na emissão dos primeiros k símbolos da cadeia emitida, para valores incrementais de k. Um problema relacionado é o de, dada a sequência emitida, calcular o valor exato da probabilidade de uma sequência de estados, o que é realizado por um algoritmo que guarda considerável semelhança com o de Viterbi, conhecido habitualmente como algoritmo *forward*.

Já o problema de treinamento em modelo oculto de Markov, na sua versão mais simples, consiste em, dados a estrutura do modelo oculto de Markov (i. e., o número de estados, o alfabeto e frequentemente também quais transições e emissões podem ter probabilidade não nula) e um conjunto de referência de sequências de símbolos fornecendo informações quanto aos estados visitados a cada emissão de símbolo, obter os valores os parâmetros do modelo (i. e., as matrizes de transição e emissão) tais que seja maximizada a verossimilhança. Esta versão mais simples possui como solução o uso de estimadores de máxima verossimilhança que são simplesmente a probabilidade de ocorrência de cada emissão e cada transição, dado o estado, estimada a partir da amostra disponível de transições e emissões. Contudo, o mais habitual é que o problema surja num contexto mais complicado, no qual a informação disponível para o conjunto de referência é simplesmente um subconjunto de estados no qual o modelo deve estar ao realizar cada emissão, e não um único estado bem conhecido. Ou seja, existe um grau de ambiguidade de estados, variável a cada emissão de símbolo. Nesta situação, o treinamento requer o uso de um algoritmo mais sofisticado, sendo comumente usado o algoritmo de Baum-Welsch, que lança mão tanto do já citado algoritmo *forward*, mas ainda de uma versão similar que calcula a probabilidade de emissão dos símbolos à frente, conhecido como algoritmo *backward*.

Os modelos ocultos de Markov, mesmo na sua forma original, podem ser usados para modelar genes. Para tanto, é possível associar cada uma das partes do gene previamente descritas (íntrons, éxons, sítios doadores e aceptores, etc.) a um estado e considerar que ocorre a emissão do nucleotídeo correspondente àquela posição. Realizado o treinamento do modelo oculto de Markov por meio de um conjunto de sequências para as quais se conheçam a localização dos vários elementos gênicos, obtém-se um preditor gênico primitivo. Um refinamento possível e natural para tal preditor é criar estados separados para cada posição na sequência correspondente a certas partes, por exemplo, para cada um de vários nucleotídeos que antecedem ou sucedem o ponto de clivagem do ácido nucléico em um sítio doador de íntron. Também se podem tratar como estados diferentes os íntrons e éxons de fases diferentes. Tais modificações melhoram muito a capacidade de predição. Entretanto, somente elas se mostram insuficientes: o uso de algumas extensões dos modelos ocultos de Markov, que veremos a seguir, foi necessário para que os preditores *ab initio* melhorassem em sensibilidade e especificidade.

No contexto de predição gênica, o problema de decodificação passa a corresponder a determinar onde estão os genes, incluindo os limites de suas várias partes, dada uma sequência genômica sem nenhuma anotação, a qual pode ser, por exemplo, a recém-obtida e inédita sequência genômica de uma espécie animal ou vegetal. Já o problema de treinamento passa a ser, dados um modelo oculto de Markov com estados correspondentes aos vários elementos do gene e um conjunto de sequências gênicas de referência contendo informação sobre a localização de tais elementos, ajustar os parâmetros do modelo de modo a maximizar a verossimilhança. A supracitada ambiguidade de estados corresponde então ao fato de que, por exemplo, as sequências de referência apenas informam quais nucleotídeos ou emissões correspondem a íntrons e a éxons, ao passo que via de regra os modelos gênicos costumam ter vários estados correspondendo a íntron e vários estados correspondendo a éxon.

3.1.2 Algumas extensões úteis para predição gênica

Algumas extensões dos modelos ocultos de Markov mostraram-se particularmente convenientes para a construção de preditores gênicos. A primeira delas consiste apenas em tornar as transições e emissões dependentes não só do estado do sistema, mas também das últimas k emissões realizadas, o que se denomina modelo oculto de Markov de ordem k (o modelo oculto de Markov convencional é então dito como sendo de ordem zero). Trata-se de uma ideia natural, visto que muitos sinais biológicos apresentam sequências relativamente conservadas em cada uma de suas posições. O treinamento do modelo permite que mesmo sinais biológicos de significado desconhecido possam, em certo sentido, ser reconhecidos a cada vez que aparecerem.

O aumento da ordem k dos modelos de Markov usados em predição gênica em geral aumenta a acurácia da predição, para valores pequenos de k. Um grave problema surge, porém, quando se aumenta um pouco mais o valor de k: logo passa a haver amostras de tamanho insuficiente, pois o número de possíveis subsequências recém-emitidas de tamanho k é 4^k, considerando-se o alfabeto de quatro bases nitrogenadas. Se houver um total de N subsequências de tamanho k no conjunto de sequências de treinamento, haverá no máximo $N/4^k$, sendo que para algumas das mesmas poderá haver bem menos que isso. Sem uma amostragem suficiente, é inevitável que a acurácia da predição se reduza.

Um método que aproveita a pertinência de usar a informação dos últimos símbolos emitidos e evita, quando surge, o problema de amostragem insuficiente, são os chamados modelos ocultos de Markov de alcance variável. O que é feito nestes modelos é reduzir seletivamente o valor de k sempre que a frequência total de ocorrência de algum k-mero cai abaixo de algum limiar definido como necessário para garantir boa amostragem. As transições e emissões passam a depender, nesta situação, apenas dos últimos $k-1$ (ou $k-2$, $k-3$, ...) caracteres emitidos, até que a frequência para a amostra disponível supere o limiar estabelecido.

Outro importante modelo que tenta amenizar o problema da falta de amostragem suficiente é chamado de modelo oculto de Markov interpolado. Neste caso, haverá uma probabilidade de transição e emissão para cada estado atual e cada k-mero recém-emitido, de modo idêntico ao que se tem para um modelo de Markov de ordem k. A diferença está no modo como se faz o treinamento deste tipo de modelo: a estimativa da probabilidade destas transições e emissões no caso geral não leva em conta somente os dados de treinamento para o ocorrência dos k-meros recém-emitidos, mas usa um sistema de pesos para levar em conta potencialmente também os dados referentes a todos os j-meros recém-emitidos, com j variando de 1 a k. Os sistemas de pesos são arbitrários, mas desenhados de modo a dar o maior peso a uma faixa de valores de j que sejam os maiores possíveis que possuam uma frequência suficientemente alta para garantir boa amostragem.

Em predição gênica, existe o fenômeno de, nas regiões que codificam para proteína, haver uma influência forte da fase dos nucleotídeos, por certo refletindo o fato de o primeiro, o segundo e o terceiro nucleotídeos de um códon apresentarem influência marcantemente diferente sobre os aminoácidos que estarão presentes na proteína correspondente. Devido a isso percebeu-se ser bastante conveniente o uso de modelos estatísticos que tratam separadamente os nucleotídeos nas três fases: usam-se na verdade três modelos ocultos de Markov de ordem k treinados independentemente, mantendo apenas a dependência das emissões e transições de cada um deles aos k-meros recém-emitidos, contendo obviamente, para $k >= 3$, nucleotídeos de todas as fases. Estes modelos, que podem ter outra periodicidade na sua dependência da posição que não apenas as trincas de caracteres, recebem o nome geral de modelos ocultos de Markov inomogêneos.

3.2 Modelos ocultos de Markov generalizados

Um modelo estatístico de especial destaque para predição gênica são os chamados modelos ocultos de Markov generalizados. Eles são definidos de modo semelhante a um modelo oculto de Markov, apenas com o acréscimo de um modelo estatístico que determine a chamada duração de cada estado, que corresponde simplesmente ao comprimento da sequência emitida ao passar por aquele estado. Ou seja, enquanto nos modelos ocultos de Markov tradicionais cada estado levava à emissão de apenas um símbolo do alfabeto, aqui será possível a emissão de um número arbitrariamente grande de símbolos.

Na verdade, é extremamente comum no caso dos modelos ocultos de Markov tradicionais haver a emissão de múltiplos símbolos, um por vez, com o sistema permanecendo no mesmo estado através de uma auto-transição. O problema é que nestes casos haverá uma probabilidade fixa p desta auto-transição e a probabilidade do sistema emitir uma sequência de tamanho k antes de fazer a transição para outro estado será igual a $(1-p)p^k$, com o comprimento da sequência emitida seguindo portanto necessariamente uma distribuição geométrica. Elementos gênicos como íntrons e éxons apresentam comprimento variável, mas em particular os éxons apresentam distribuições de comprimento que não são de modo algum bem modeladas por uma distribuição geométrica. O uso de modelos de Markov generalizados permite associar a cada estado quaisquer distribuições de comprimento de sequência emitida, resolvendo este problema.

Um grande número de preditores gênicos de sucesso atualmente usa um modelo deste tipo para modelar o gene como um todo, explorando o fato de que é possível deste modo usar um submodelo distinto para cada parte do gene, com poucas restrições aos tipos possíveis de submodelo.

O ToPS (*Toolkit for Probabilistic Models of Sequence*) é um conjunto de ferramentas desenvolvido pelo grupo do Prof. Alan Durham para dar suporte à análise de sequências biológicas, sejam de nucleotídeos ou aminoácidos, com a proposta de disponibilizar a um só tempo:

- um conjunto variado de alguns dos modelos estatísticos mais relevantes e flexíveis, que possam ser usados sem necessidade de conhecer linguagens de programação;
- algoritmos apropriados para problemas como treinamento e decodificação de cada modelo, além de algoritmos de classificação de sequências;
- integração entre os vários modelos e algoritmos, de modo a ser possível ao usuário construir um GHMM com vários submodelos, sem se preocupar com interfaces e conversão de formatos;
- otimizações apropriadas para a decodificação do GHMM de modo a tornar viável seu uso para predição gênica Kashiwabara (2012); Kashiwabara *et al.* (2013).

Capítulo 4

Um configurador gráfico para o ToPS

O desenvolvimento deste configurador foi proposto a partir de uma série de dificuldades que imaginamos que poderiam afetar algum profissional ou pesquisador com formação na área biológica que se propusesse a compreender e/ou modificar as configurações do ToPS. Assumimos que o sistema a ser modelado por meio do ToPS pode ser descrito por meio de um GHMM, cujos estados sejam modelos probabilísticos implementados pelo ToPS. Como referência, frequentemente imaginaremos que o sistema é um preditor gênico, no qual os vários submodelos do GHMM correspondem às várias partes componentes do gene. Adotamos em geral o MYOP e suas configurações como modelo de preditor gênico, mas tentamos manter a generalidade do configurador, dado que o próprio ToPS é uma ferramenta bastante geral, no contexto de modelagem estatística de sequências de caracteres. Assumimos que o profissional que usará o configurador compreende ao menos em parte os modelos estatísticos envolvidos. Nossa intenção foi tornar mais simples a ele realizar tarefas como as seguintes:

- visualizar e compreender como está sendo modelada a estrutura do gene completo, incluindo qual submodelo está sendo usado para cada elemento gênico;
- observar e possivelmente modificar quais parâmetros de configuração estão sendo usados em cada submodelo e em cada modelo de duração;
- perceber rapidamente quais parâmetros são compartilhados por mais de um submodelo;
- verificar quais algoritmos de treinamento estão disponíveis para cada submodelo, de modo a poder testar o efeito de mudanças de algoritmo, ou dos parâmetros que controlam o algoritmo, sobre a acurácia do modelo como um todo;
- introduzir modificações na estrutura do gene como um todo, de modo a contemplar elementos gênicos previamente não modelados;
- incluir novos submodelos que copiem parcial ou totalmente modelos previamente existentes.

Dados estes objetivos, foi implementada uma interface gráfica com uma série de elementos que tornam intuitiva a realização de tais tarefas por parte do usuário. Descrevemos a seguir quais decisões foram tomadas na criação desta interface.

Primeiro, associamos cada um dos tipos diferentes de submodelo estatístico suportado pelo ToPS a uma forma geométrica específica. Concretamente, foram usados polígonos de fácil reconhecimento. Na parte central do polígono aparece o nome dado ao submodelo. O posicionamento do cursor sobre o mesmo mostra transitoriamente qual o tipo de modelo estatístico em questão. Ao redor de cada polígono inserido surge automaticamente uma área

circular que representa a duração do submodelo. Cada submodelo pode ser selecionado com um clique simples do usuário, após o que muda a cor de sua borda e torna-se possível arrastá-lo, excluí-lo ou criar arestas que tenham origem ou destino no mesmo (logo abaixo tratamos destas arestas). Também foi implementada seleção múltipla de modelos.

Segundo, as transições entre submodelos do GHMM com probabilidade não nula no GHMM são representadas por meio de arestas direcionais entre as figuras geométricas mencionadas. Algoritmos empíricos foram usados para tornar automaticamente natural em termos visuais o posicionamento dos componentes das arestas: um segmento de reta e um triângulo, que são dispostos automaticamente em função apenas do posicionamento dos dois polígonos que correspondem às extremidades da aresta. Ou seja, para cada aresta, o usuário inicialmente especifica apenas modelo de origem e modelo de destino. Tomou-se cuidado especial para tornar elegante a representação do caso em que há arestas simultaneamente em ambos os sentidos. Também recebeu um tratamento especial o caso dos laços (arestas começadas e terminadas no mesmo submodelo), para os quais tornamos possível ao usuário reposicionar a aresta ao redor do submodelo, de modo que a mesma interfira menos com outros elementos gráficos próximos. Ou seja, abriu-se nesta situação uma exceção ao uso do posicionamento automático da aresta, ainda que no momento da criação o posicionamento seja automático.

Para testar nosso sistema, tomamos emprestado dos arquivos de exemplos que acompanham o ToPS um modelo de gene bastante simples. Trata-se de um modelo de gene procariótico, sem íntrons. Sua representação na tela principal do nosso sistema é vista na Figura 4.1. O mesmo tipo de representação em nosso sistema, para um gene eucariótico simplificado, bem mais complexo, aparece na Figura 4.2. Note-se que o modelo de gene usado no MYOP é ainda mais complexo, por exemplo incluindo submodelos para a fita complementar do DNA, o que imediatamente implica uma duplicação do número de submodelos do GHMM.

Os vários elementos do GHMM podem, uma vez selecionados, ser editados. Para tanto, basta ao usuário dar um duplo clique na área da representação gráfica. As telas para edição de submodelo, modelo de duração e aresta são vistas, nesta ordem, nas Figuras 4.3, 4.4 e 4.5.

A janela de edição de submodelo, além de permitir a entrada de um nome para o submodelo e a visualização e edição de sua especificação, disponibiliza opções de cópia, por valor ou por referência, das especificações de outros modelos do mesmo tipo previamente editados no GHMM, e a opção de obter a especificação por treinamento. A cor do modelo, branca no momento de sua inserção, pode ser diretamente escolhida, ou é possível solicitar uma ou mais vezes a geração automática de cores. O algoritmo de geração de cores soma módulo 256 aos valores RGB da cor previamente gerada três números primos entre si e a 256, de modo que cores não se repetem antes que todas sejam geradas. Em caso de cópia, a cor também é copiada. No caso de cópia por referência, modificações posteriores da cor do modelo referido se refletirão na representação dos modelos definidos por referência ao mesmo. Modelos definidos por referência não podem ser referidos por outros modelos.

A janela de edição de aresta é extremamente simples, permitindo apenas edição da probabilidade da transição correspondente no GHMM. A única sutileza é a possibilidade de selecionar e editar de uma só vez várias arestas. Arestas selecionadas também podem ser removidas do GHMM.

Na janela de edição de duração, há as seguintes opções: duração fixa, duração geométrica com uso de laço e duração dada por um modelo estatístico que gere um número inteiro positivo, sempre à semelhança do ToPS. No caso de duração dada por um modelo estatístico, há novamente as opções de obtenção da especificação por meio de treinamento e de cópia por valor e cópia por referência da especificação do modelo de duração de algum outro submodelo

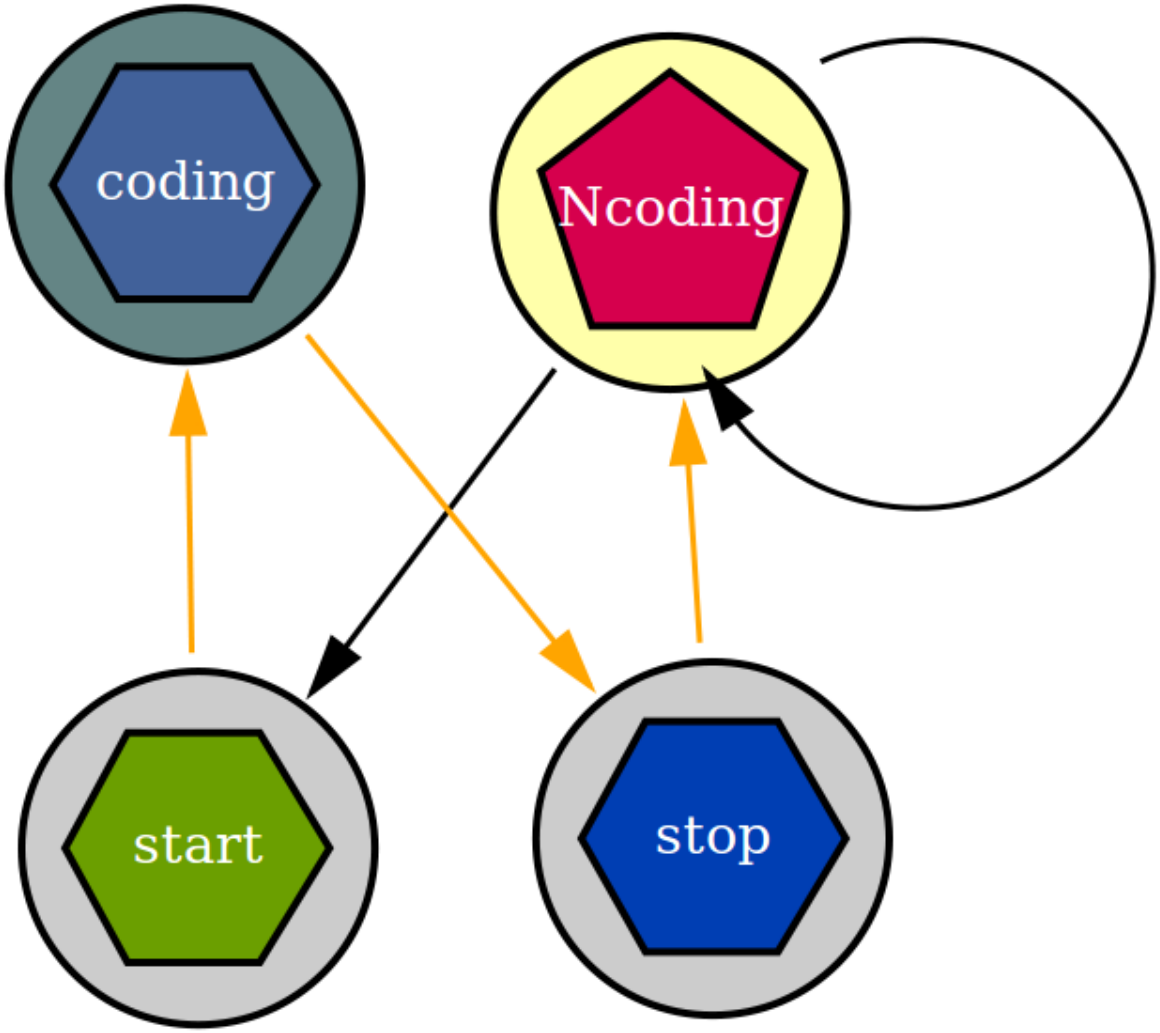

Figura 4.1: ***Preditor gênico minimalista representado como GHMM no sistema.*** *Hexágonos e o pentágono representam, respectivamente, cadeias de Markov inomogêneas e um modelo discreto independente e identicamente distribuído. O círculo amarelo pálido indica que o modelo de duração é geométrico, dado em função da probabilidade de auto-transição. Círculos de cor cinza indicam um modelo com duração fixa, dada por uma constante inteira. Coding = região codificante, NCoding = região não codificante, start (stop) = sítio de início (parada) da tradução. Notar as três arestas selecionadas em laranja, bem como o laço existente no submodelo para a região não codificante.*

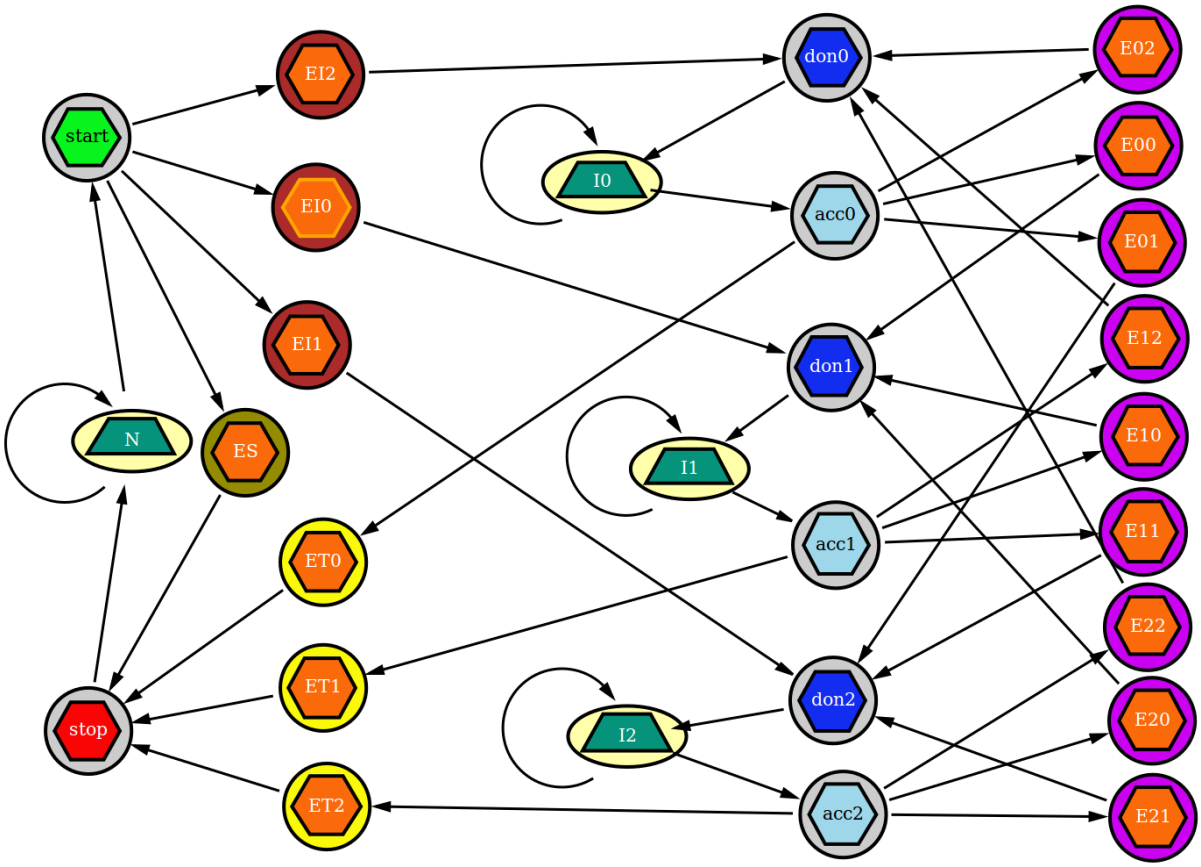

Figura 4.2: ***Preditor simplificado para genes eucarióticos representado como GHMM no sistema.*** *Hexágonos e trapézios representam, respectivamente, cadeias de Markov inomogêneas e cadeias de Markov de alcance variável. Círculos/elipses de cor amarelo pálido e cinza representam duração geométrica e fixa, respectivamente. Polígonos e outros círculos/elipses de mesma coloração indicam compartilhamento de, respectivamente, submodelos de sequência e modelos de duração. É o caso, por exemplo, dos submodelos para éxons (ES, EIk, ETk, Ejk, j, k = 0, 1 ou 2, em laranja), íntrons (Ik, em verde escuro), sítios doadores (donk, em azul escuro) e aceptores de íntrons (acck, em azul claro), ou dos modelos de duração para éxons internos (Ejk, em magenta), iniciais (Eik, em bordô) ou terminais (Etk, em amarelo intenso). N = região não codificante, start (stop) = sítio de início (parada) da tradução, ES = éxon único. Notar que o submodelo para o éxon E10 aparece selecionado, conforme indica a mudança da cor do polígono de preto para laranja. O grafo para este preditor foi extraído de Kashiwabara (2012).*

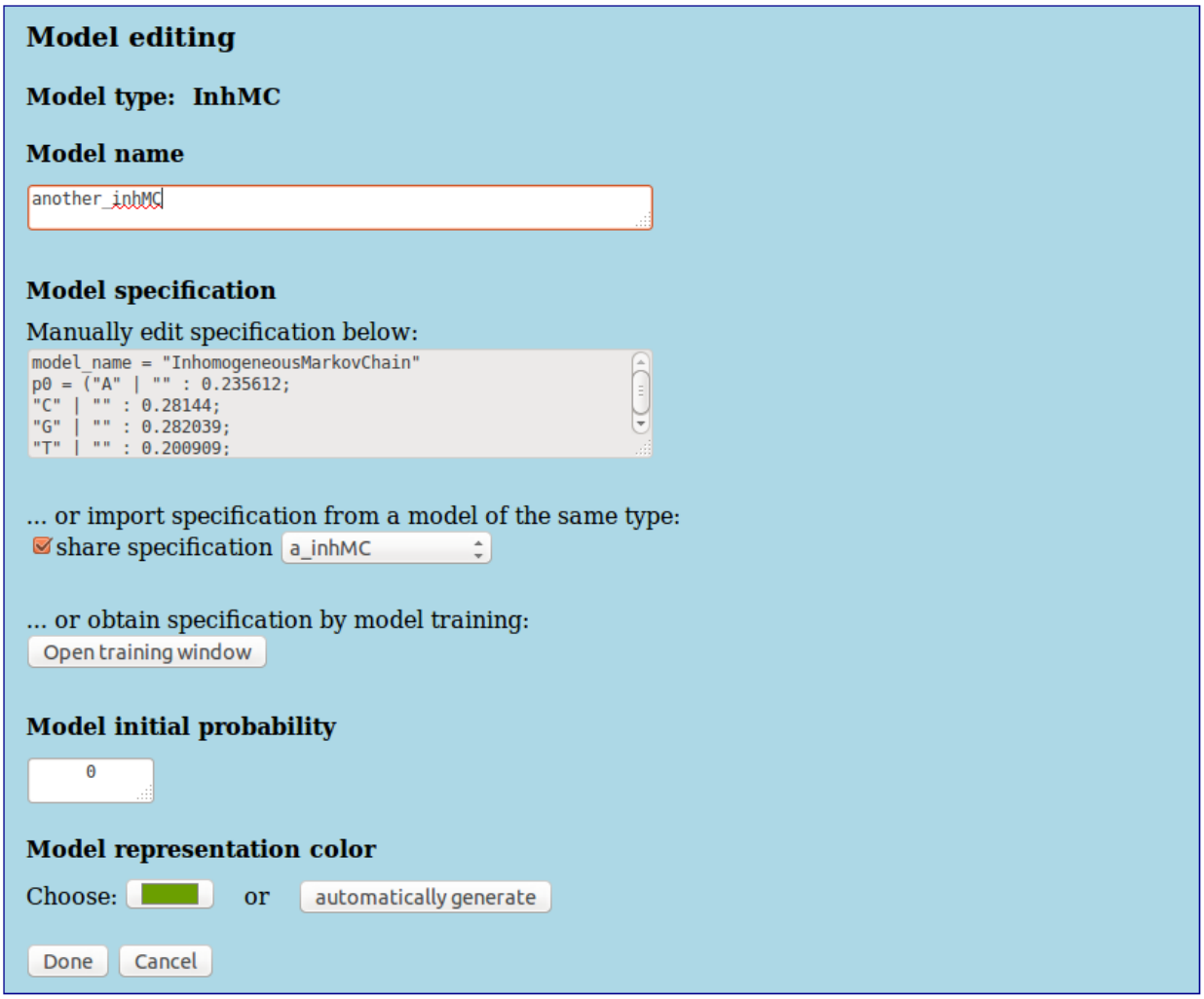

Figura 4.3: *Janela de edição de submodelo.*

Figura 4.4: *Janela de edição de duração.*

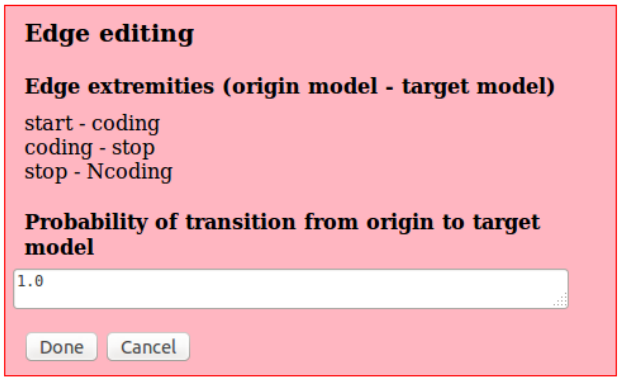

Figura 4.5: ***Janela de edição de aresta.*** *Note-se que as arestas sendo editadas correspondem exatamente às arestas que aparecem selecionadas na figura 4.1.*

do GHMM. Neste caso não há a restrição de que o submodelo seja do mesmo tipo de modelo estatístico. O comportamento das cores dos modelos de duração é bastante semelhante aos das cores dos submodelos, exceto pelo algoritmo de geração de cores levar à geração de tons mais acinzentados para os modelos de duração, e pelo fato de nos casos de duração fixa e geométrica a cor ser sempre cinza claro e amarelo pálido. A Figura 4.2 ilustra em suas cores o compartilhamento de modelos de duração num modelo gênico para eucariotos. Igualmente, nas opções padrão do MYOP muitas durações diferentes estão descritas por alguns poucos modelos estatísticos.

A obtenção da especificação de um submodelo do GHMM ou de um modelo de duração por meio de treinamento requer uma chamada ao servidor e isso é feito por janelas próprias. A janela de treinamento de submodelo é mostrada na Figura 4.6. Note-se que o tipo de modelo estatístico foi previamente escolhido. Nesta tela, a escolha do algoritmo de treinamento (dentre os disponíveis no ToPS para aquele tipo de modelo) leva a janela a automaticamente oferecer ao usuário a escolha dos parâmetros de treinamento necessários e opcionais. Estes parâmetros são definidos num arquivo de configuração do sistema, o qual simplesmente define um objeto em formato JSON contendo o nome e o tipo dos parâmetros, para cada modelo estatístico e cada algoritmo de treinamento. Além do formato JSON ser bastante legível a humanos, isso permitirá adaptar facilmente o sistema configurador a futuras incorporações de modelos e algoritmos ao ToPS, que permanece em desenvolvimento e refatoração no presente.

Além da possibilidade de inserir modelos estatísticos e arestas e dispor estes elementos na área gráfica, a tela principal oferece mais algumas funcionalidades, ilustradas na Figura 4.7. A mais importante destas é a submissão de sequências ao servidor para decodificação de acordo com o GHMM. Neste caso, o cliente deve submeter ao servidor toda a especificação relevante do GHMM.

Quanto à estrutura do programa cliente, optamos por deixar os dados correspondentes aos arquivos de especificação de modelos e configurações de treinamento no próprio DOM (Document Object Model) da página, dado que estes dados apresentam dimensões modestas. Arquivos de treinamento ou sequências a serem decodificadas, por outro lado, podem apresentar tamanhos facilmente problemáticos para este tipo de armazenamento, de modo que são sempre submetidas imediatamente ao servidor, não sendo armazenadas. O uso de elementos gráficos do tipo SVG (Scalable Vector Graphics) se mostrou relativamente fácil e apresenta a inegável vantagem de não ocorrerem problemas relacionados à degradação da qualidade dos elementos gráficos quando eles são reduzidos.

A decisão de desenvolver o sistema cliente em linguagem Javascript teve certas consequências. Ela foi escolhida tendo em mente ser suportada por todos os navegadores. Dada esta vantagem inicial, com vistas a tornar mais simples o desenvolvimento, optamos por dar suporte apenas aos navegadores mais recentes. A consequência principal da escolha da linguagem, porém, tem relação com suas múltiplas restrições de segurança. A configuração final de um GHMM no MYOP é descrita por dezenas de arquivos, ao passo que Javascript, usada por um cliente no navegador, restringe o acesso em disco a um arquivo selecionado pelo usuário por meio de uma ação. Se por um lado um único arquivo é suficiente para armazenar todas as configurações de todos os modelos, por outro essa restrição da linguagem torna impossível que a configuração seja feita sem a chamada a um servidor, que então criará localmente todos os arquivos de configuração necessários para que o ToPS possa ser invocado por uma linha de comando. De fato, também a chamada ao sistema operacional para executar a linha de comando é impossível a partir do navegador, novamente tornando o servidor necessário. Estes procedimentos são também necessários na fase de treinamento dos modelos estatísticos, quando os parâmetros a serem escolhidos são outros, mas igualmente

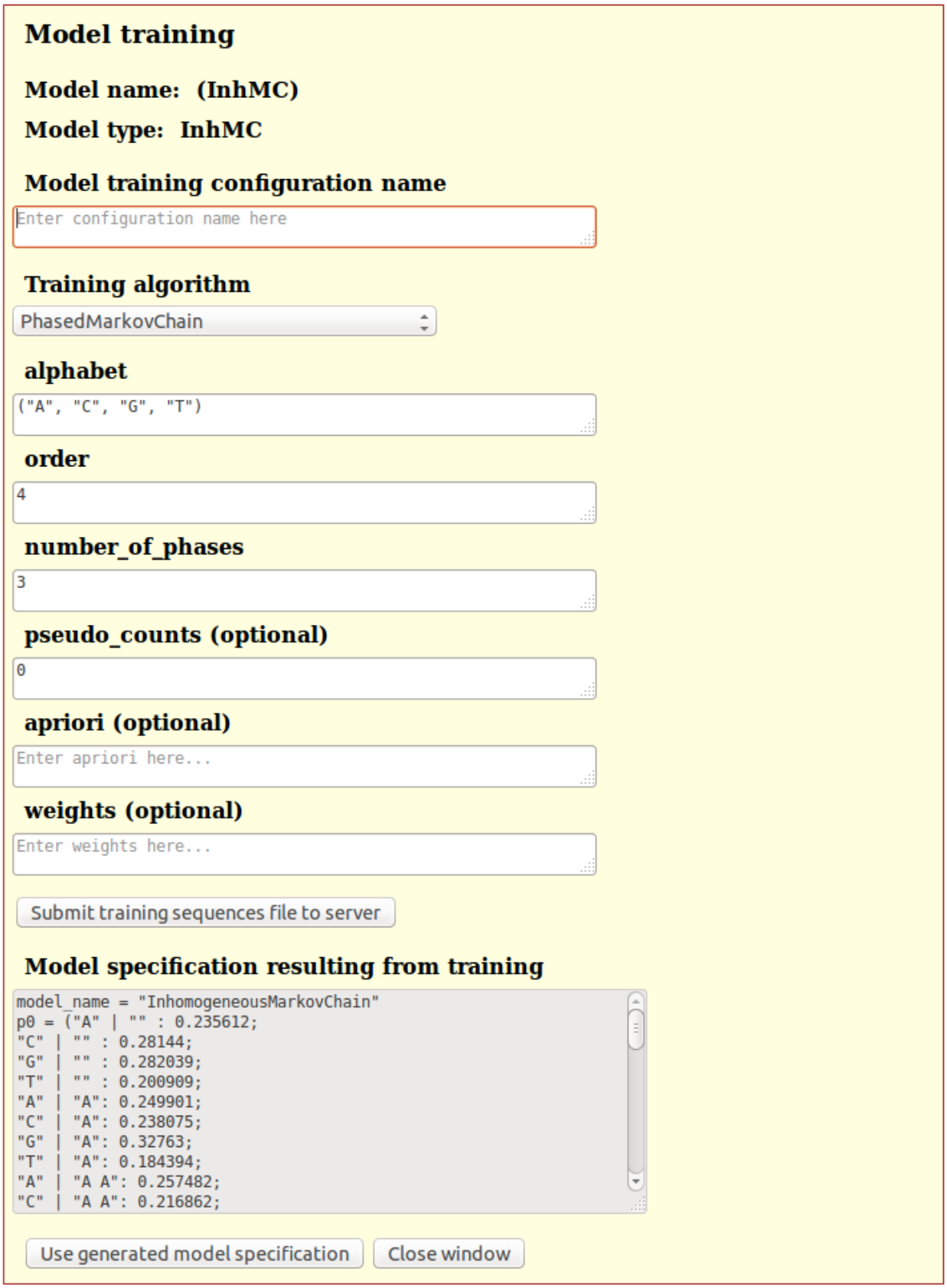

Figura 4.6: ***Janela de treinamento de modelo.*** *Abaixo do botão de submissão das sequências de treinamento ao servidor existe uma janela de texto não editável onde é exibida a especificação de um modelo gerada pelo ToPS e retornada pelo servidor ao cliente.*

torna-se necessário invocar o ToPS por linha de comando.

O sistema servidor, por sua vez, foi desenvolvido usando as tecnologias Express e Node.js. Isso já o habilita a processar de modo independente várias requisições simultâneas. Cada requisição leva à criação de um subdiretório temporário no servidor. No caso de uma requisição de decodificação de sequência, é criado neste subdiretório um arquivo ghmm.model contendo os dados entrados para o GHMM no formato esperado pelo ToPS, bem como um arquivo .model para cada um dos submodelos. Por fim, o ToPS é invocado e em caso de sucesso o cliente recebe para baixar um arquivo contendo o resultado da decodificação feita pelo ToPS. As requisições de treinamento de submodelo e treinamento de duração exigem apenas que seja colocado no diretório temporário um arquivo de configuração do treinamento e o arquivo de sequências de treinamento, sendo o retorno dos dados ao servidor dirigido para uma área de texto na janela de treinamento correspondente à especificação de modelo gerada pelo treinamento.

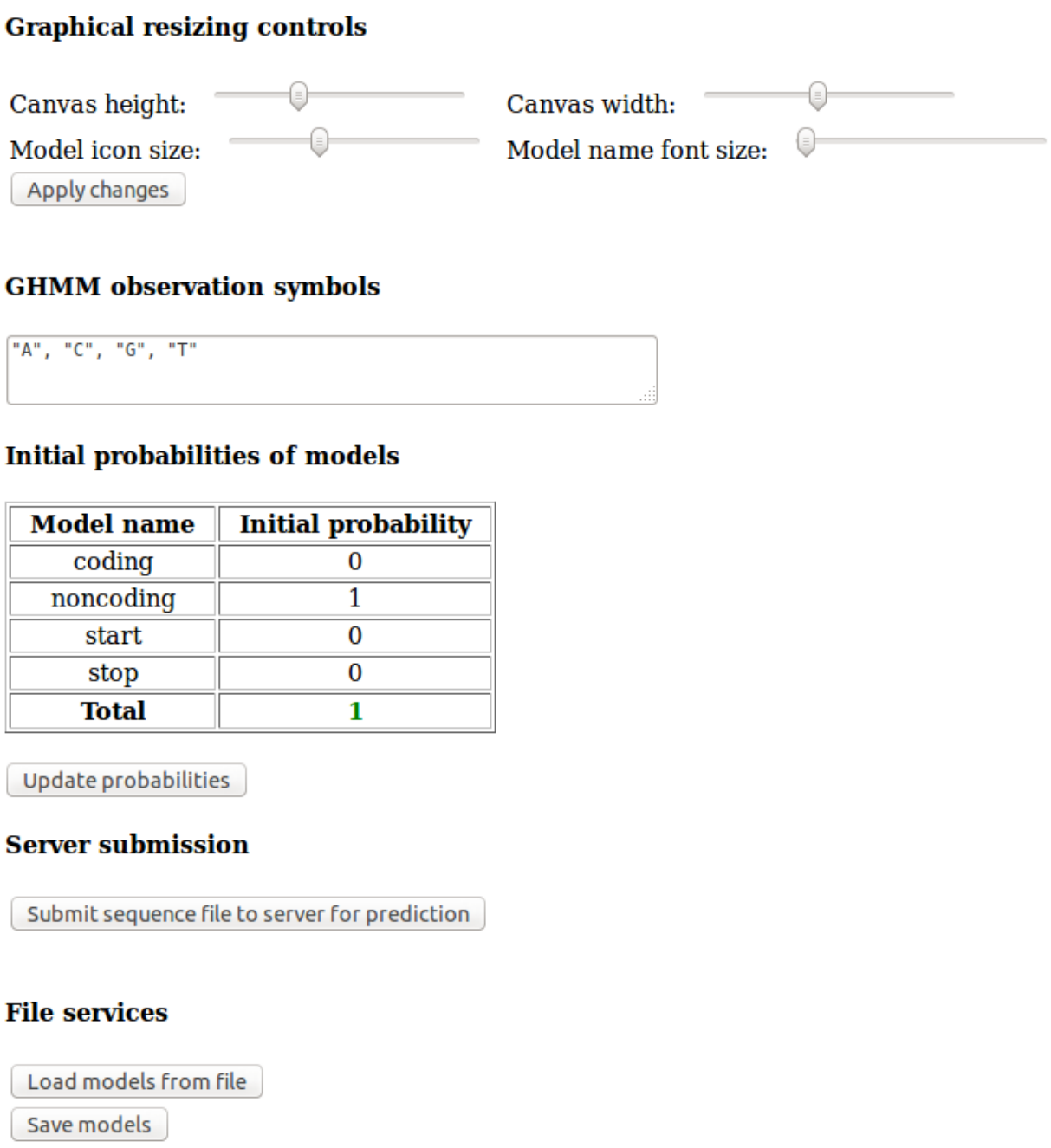

Figura 4.7: ***Vista parcial da janela principal do sistema.*** *Como se vê, algumas funcionalidades disponibilizadas incluem alterações nas dimensões do espaço de representação, dos elementos representados e da fonte usada para os nomes, verificação e edição da probabilidade inicial de submodelos, submissão de sequências ao servidor para decodificação de acordo com o GHMM, gravação e recuperação do GHMM em disco. Note-se que a tabela de probabilidades iniciais fornece automaticamente o cálculo da soma das probabilidades e alerta o usuário caso ela não valha 1.*

Capítulo 5

Conclusões e perspectivas futuras

Temos um sistema cliente-servidor que já apresenta uma interface bastante amigável para a edição das configurações do ToPS. Ele em breve deverá facilitar de fato o uso do ToPS, em toda a sua flexibilidade de configurações. O sistema cliente contempla todos os elementos básicos para uma fácil especificação do GHMM, seus submodelos e seus modelos de duração. No momento, já são disponibilizadas pelo sistema servidor as funções de decodificação de sequências, treinamento de submodelo e treinamento de modelo de duração. A função de simulação de modelo por certo pode ser trivialmente acrescentada ao sistema atual.

O sistema deverá ter seu desenvolvimento continuado pelos membros do grupo. Uma funcionalidade que talvez seja interessante obter seja o treinamento de cada modelo estatístico por um arquivo de sequências de treinamento gerado no momento pelo sistema MYOP, de modo automático para todos os submodelos do GHMM. Isso não é permitido num sistema cliente baseado em navegador, no qual restrições de segurança fazem com que cada arquivo a ser acessado deva ser indicado pelo usuário. O navegador certamente não pode explorar diretórios em busca dos arquivos gerados automaticamente pelo MYOP. Porém, há sistemas como o Electron, que oferecem as funcionalidades normais de um navegador e a possibilidade de relaxar as restrições de segurança citadas. O grupo está estudando a viabilidade de usar o Electron para tal finalidade.

Referências Bibliográficas

Kashiwabara(2012) AY Kashiwabara. *MYOP/ToPS/SGEval: um ambiente computacional para estudo sistemático de predição de genes*. Tese de Doutorado, Departamento de Ciência da Computação, Universidade de São Paulo, Brasil. Citado na pág. 10, 14

Kashiwabara *et al.***(2013)** AY Kashiwabara, I Bonadio, V Onuchic, F Amado, R Mathias e AM Durham. Tops: a framework to manipulate probabilistic models of sequence data. *PLoS Comput Biol*, 9(10):e1003234. Citado na pág. 10

Majoros(2007) WH Majoros. *Methods for Computational Gene Prediction*. Cambridge University Press. Citado na pág. 3

Yoon(2009) BJ Yoon. Hidden markov models and their applications in biological sequence analysis. *Curr Genomics*, 10(6):402. Citado na pág. 1