INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
UNIVERSIDADE DE SÃO PAULO

Bacharelado em Ciência da Computação

Montagem de regiões gênicas

Pedro Ivo Gomes de Faria
pedro.faria@usp.br

Supervisor: Prof. Dr. Alan Mitchell Durham
durham@ime.usp.br

São Paulo - SP

Primeiro semestre de 2013

A todos aqueles que acreditaram em mim e me deram o apoio que me permitiu chegar até aqui.

Agradecimentos

Primeiramente, agradeço ao professor Dr. Alan Durham pela orientação dada durante este trabalho e durante a iniciação científica (IC). Agradeço também à professora Dra. Glaucia Souza pela disponibilização dos dados, e à pós-doutoranda Roberta Campos e ao mestrando Abdalla Almeida pela ajuda na obtenção e no pré-processamento desses dados.

Também agradeço aos poucos (mas presentes) colegas da graduação que me apoiaram durante o curso, seja com a paciência para ouvir minhas lamúrias ou me ajudando nas disciplinas que tivemos a oportunidade de cursar juntos.

Agradeço a todos os professores do Instituto de Matemática e Estatística (IME), do Instituto de Biociências (IB), do Instituto de Química (IQ) e da Escola Politécnica (Poli) com os quais tive a oportunidade de cursar disciplinas. De alguma forma, todos tiveram alguma influência na minha formação, tanto acadêmica (ter uma formação interdisciplinar foi vital para o desenvolvimento deste trabalho) quanto pessoal (muitos tiveram atitudes que considero exemplares, embora - infelizmente - eu também tenha tido contato com alguns antiexemplos). Agradeço também aos funcionários dessas unidades, pela paciência e ajuda nas questões burocráticas[1] que surgiram como consequência dessa "saga" interdisciplinar.

Aproveito para agradecer os colegas e professores da escola e do cursinho, pois desde o início recebi reconhecimento, apoio e incentivo deles para continuar com minha dedicação aos estudos. Sem isso, provavelmente eu não teria conseguido nem entrar na Universidade de São Paulo (USP).

Finalmente, agradeço ao Estado de São Paulo por manter uma universidade pública, gratuita e de qualidade como a USP, sem a qual eu não teria a oportunidade de cursar o ensino superior.

[1] infelizmente, tais questões foram mais presentes do que eu gostaria...

“Seja a mudança que você quer ver no mundo.”

MAHATMA GANDHI

“Todas as vitórias ocultam uma abdicação.”

SIMONE DE BEAUVOIR

Resumo

A montagem de sequências refere-se ao alinhamento e fusão de fragmentos (os fragmentos fundidos denominam-se *contigs*) vindos de uma molécula de DNA maior para poder reconstruir a sequência original. Isto é necessário pois a tecnologia atual de sequenciamento de DNA não consegue lidar com cromossomos inteiros, mas apenas com pequenos fragmentos (chamados de *reads*) de tamanho entre 20 e 1000 pares de bases [1]. Além da grande quantidade de dados gerada pelos ditos sequenciadores da "próxima geração" (*next generation sequencing* ou NGS) [2], outros problemas incluem a presença de erros nos *reads* e a existência de sequências quase idênticas (conhecidas como repetições), que podem dificultar a montagem (gerando *contigs* que não existem na molécula original, chamados de quimeras) [3].

Para tentar evitar as dificuldades causadas pelas repetições, a ferramenta desenvolvida tentará apenas obter os genes (e suas regiões adjacentes) de interesse do usuário (mais precisamente, tentará montar apenas os *reads* que tenham um mínimo de similaridade com as sequências de interesse). Idealmente, iremos obter também os elementos cis-regulatórios (regiões do DNA que regulam a expressão de genes localizados na mesma molécula [4]) dos genes em questão.

Palavras-chave: montagem; DNA; gene; Perl; *pipeline*; alinhamento; sequenciamento.

Sumário

I Parte Objetiva **10**

1 Introdução **10**

1.1 Motivação 10

1.2 Objetivos 11

1.3 Organização da monografia 11

2 Mecanismos genéticos básicos **12**

2.1 Estrutura do DNA 12

2.2 Duplicação do DNA 13

2.3 Transcrição do RNA 15

2.3.1 *Splicing* do pré-RNAm 17

2.4 Tradução do RNA 18

3 Sequenciamento de genomas **22**

3.1 Estratégias 22

3.1.1 Sequenciamento *shotgun* 22

3.1.2 Sequenciamento BAC a BAC (*shotgun* hierárquico) 23

3.2 Pirossequenciamento 24

3.2.1 Passo 1: preparação das amostras de DNA (duração: 4 a 5h) 25

3.2.2 Passo 2: PCR em emulsão (emPCR) (duração: 8h) 25

3.2.3 Passo 3: sequenciamento (duração: 7,5h) 26

4 Alinhamento de sequências **28**

4.1 Definição 28

4.2 Medidas (identidade e cobertura) 29

4.3 Tipos de alinhamentos 30

4.3.1 Alinhamento global 30
4.3.2 Alinhamento local 30
4.3.3 Alinhamento semiglobal 31
4.4 Alinhamento heurístico 31

5 Montagem de sequências **32**
5.1 Definição 32
5.2 Complicações tecnológicas 33
5.2.1 Erros 33
5.2.2 Orientação desconhecida 35
5.2.3 Repetições 36
5.2.4 Falta de cobertura 38
5.3 Modelagem 39
5.4 Complicações teóricas 39

6 Implementação **42**
6.1 O *pipeline* de mascaramento 42
6.2 O *pipeline* de montagem 42
6.2.1 Leitura dos parâmetros 43
6.2.2 Leitura dos arquivos com os *reads* e com as sequências de consulta 43
6.2.3 Divisão do arquivo com os *reads* 43
6.2.4 Alinhamento das sequências de consulta nos *reads* 44
6.2.5 Seleção das sequências de consulta correspondentes a *reads* 44
6.2.6 Seleção dos *reads* correspondentes a sequências de consulta 45
6.2.7 Seleção dos *reads* não mapeados durante o alinhamento 45
6.2.8 Montagem inicial das regiões gênicas 45
6.2.9 Extensão final das regiões gênicas 45
6.3 O *pipeline* de validação 46

7 Resultados **48**

8 Conclusão **49**

Glossário **50**

Referências **61**

II Parte Subjetiva 72

9 Desafios e frustrações **73**

9.1 Em relação ao curso 73

9.2 Em relação ao TCC 75

10 Disciplinas relevantes e conceitos utilizados **76**

10.1 Cursadas no IME 76

10.2 Cursadas em outras unidades 77

11 Planos para continuação na área **78**

Lista de Figuras

1 Organização e localização do DNA 10

2 Estrutura de um desoxirribonucleotídeo 12

3 Estrutura química da molécula de DNA 13

4 Duplicação semiconservativa do DNA 14

5 Química da síntese de DNA 14

6 Diferenças entre DNA e RNA 15

7 A conformação de uma molécula de RNA 16

8 Transcrição do DNA pela RNA-polimerase 17

9 Transcrição e *splicing* do pré-RNAm eucariótico 18

10 Estrutura genérica de um aminoácido 18

11 Os 20 aminoácidos que compõem as proteínas 19

12 O código genético 20

13 Formação da ligação peptídica 20

14 As três fases da tradução do RNAm 21

15 As fases do sequenciamento *shotgun* 22

16 BACs e *mate pairs* 23

17 *Tiling path* 24

18 As fases do sequenciamento *shotgun* hierárquico 24

19 As fases do passo 1 do pirossequenciamento 25

20 As fases do passo 2 do pirossequenciamento 26

21 Início do passo 3 do pirossequenciamento 26

22 Fase intermediária do passo 3 do pirossequenciamento 27

23 Fase final do passo 3 do pirossequenciamento: pirograma 28

24 Exemplo de alinhamento 29

25 *Phasing* e *pre-phasing* 33

26 Tipos de sobreposições entre fragmentos 36
27 Colapso de repetições seguidas (em tandem) 37
28 Excisão de regiões flanqueadas por repetições 37
29 Rearranjo de regiões flanqueadas por repetições 37
30 Inversão causada por repetições invertidas 38
31 Cobertura do genoma e o processo de montagem 38
32 Problemas causados por repetições na modelagem via *superstring* comum mais curta 40

Parte I

Parte Objetiva

1 Introdução

1.1 Motivação

Cada célula de um organismo vivo contém um conjunto de cromossomos, que são formados principalmente de DNA. Esse conjunto de toda a informação hereditária do organismo (o genoma) representa as instruções que controlam a sua duplicação e o seu funcionamento. O sequenciamento automático de DNA deu origem à genômica, que consiste no estudo analítico e comparativo de genomas diferentes. O problema é que os genomas possuem tamanhos variando de milhões (em bactérias) a bilhões (em humanos e na maioria dos animais e plantas) de nucleotídeos, sendo que a maioria dos métodos atuais de sequenciamento são precisos o suficiente para determinar sequências contínuas de não mais que 900 nucleotídeos em média [2, 5]. Mesmo o método capaz de determinar as sequências mais longas até o momento (chamado de SMRT, do inglês *single molecule real time sequencing*) não consegue (em média) determinar sequências contínuas de mais de 2900 nucleotídeos[2][6].

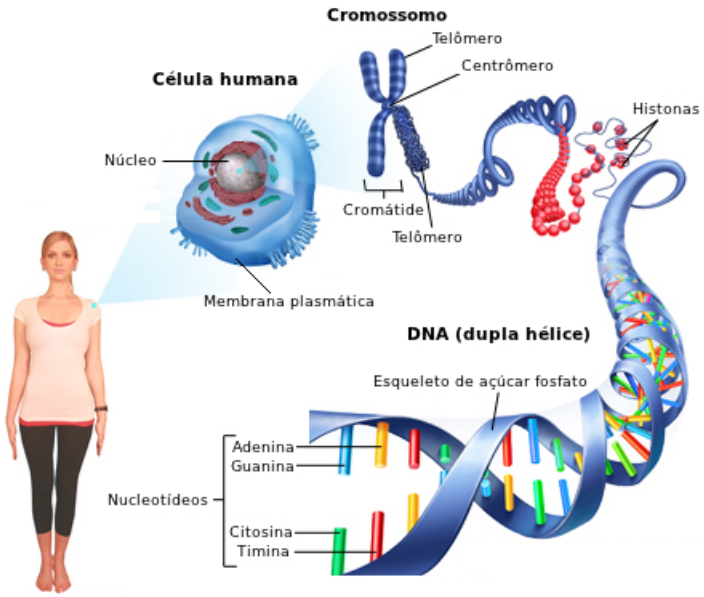

Figura 1: Organização do DNA em cromossomos (onde o DNA está associado a histonas) e sua localização na célula (que é o núcleo, no caso de eucariontes). Fonte: [8].

[2] o tamanho máximo é de 15000 nucleotídeos[7].

Tal como um grande quebra-cabeça, os *reads* (fragmentos) de DNA produzidos pelo sequenciador devem ser montados para a obtenção de uma representação completa do genoma. Porém, os *reads* contém erros (oriundos das limitações da tecnologia de sequenciamento ou de falhas humanas[3]), o que dificulta a tarefa. Mesmo na ausência de erros, o DNA possui características que complicam consideravelmente o processo de montagem, tais como as repetições. O genoma humano, por exemplo, possui repetições que aparecem mais de 100000 vezes cada uma[4]. Assim como as peças correspondentes ao "céu" no quebra-cabeça de uma paisagem, os *reads* que correspondem a repetições são difíceis de posicionar corretamente[5], o que resulta em lacunas na sequência montada [5].

1.2 Objetivos

O objetivo principal deste trabalho é obter um programa que monte as regiões gênicas de interesse do usuário, estendendo-as o máximo possível (de forma confiável, ou seja, sem gerar quimeras) para obter os elementos cis-regulatórios dos genes em questão. Como objetivo secundário, visamos integrar e consolidar conceitos obtidos tanto nas disciplinas do Bacharelado em Ciência da Computação (BCC) quanto nas disciplinas relativas à área de biológicas (cursadas em outros institutos, principalmente no IB e no IQ).

1.3 Organização da monografia

A monografia é dividida em duas partes: a objetiva e a subjetiva.

A parte objetiva possui 7 seções (sem considerar esta introdução), descritas a seguir. Na seção 2 é feita uma apresentação de conceitos básicos de biologia molecular[6] que permeiam todo o trabalho. Na seção 3 é feita uma descrição das principais abordagens para o sequenciamento de DNA, além de uma exposição da tecnologia utilizada para gerar os dados analisados. Na seção 4 é explicado o conceito de alinhamento de sequências e sua relação com o problema da montagem. Na seção 5 é definido o problema da montagem e suas dificuldades. Na seção 6 são descritos os *pipelines* desenvolvidos, incluindo funcionalidades e alguns detalhes de implementação. Na seção 7 são analisados os resultados obtidos para a montagem de sequências de DNA do cultivar R570 de cana-de-açúcar (híbrido entre *S. officinarum* e *S. spontaneum*). Finalmente, a seção 8 apresenta as considerações finais em relação ao trabalho.

[3] cometidas durante a execução do protocolo de sequenciamento.

[4] como os elementos Alu, que aparecem mais de um milhão de vezes[9].

[5] a maior parte do genoma humano não sequenciado corresponde a regiões repetitivas, concentradas em telômeros e centrômeros[10].

[6] de forma simplificada (omitindo alguns detalhes), explicando o que for necessário para o entendimento do que foi desenvolvido.

A parte subjetiva relata as experiências vividas durante a graduação e durante o desenvolvimento do trabalho, além de relacionar os conceitos estudados que foram mais relevantes para a execução do mesmo.

2 Mecanismos genéticos básicos

2.1 Estrutura do DNA

A molécula de DNA é uma fita dupla, sendo que cada monômero em uma das fitas (ou seja, cada nucleotídeo) consiste de duas partes: um açúcar (desoxirribose) com um grupo fosfato ligado a ele e uma base, que pode ser adenina (A), timina (T), citosina (C) ou guanina (G). Cada açúcar está ligado ao próximo por meio do grupo fosfato[7], criando uma cadeia composta por um esqueleto repetitivo de açúcar e fosfato, com séries de bases projetando-se dela. As bases de uma fita ligam-se com as bases da outra fita de forma complementar[8], sendo que A liga-se com T e C liga-se com G. Essas duas fitas torcidas entre si formam uma dupla-hélice[11].

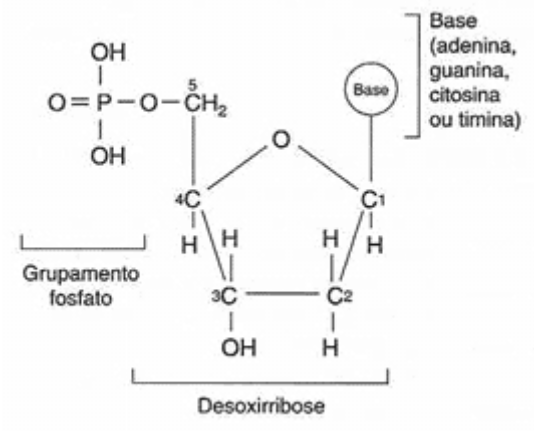

Figura 2: Estrutura de um desoxirribonucleotídeo, mostrando também a numeração dos átomos de carbono. Fonte: [12].

A forma com que os nucleotídeos estão ligados faz com que os terminais de cada fita sejam diferentes (definindo uma orientação a cada uma delas): o terminal com um grupo fosfato livre (i.e., que não participa da ligação entre nucleotídeos) é denominado terminal $5'$ (lê-se "cinco linha"), enquanto o terminal com uma hidroxila livre é denominado terminal $3'$ (lê-se "três linha")[9]. Os

[7] através de ligações fosfodiéster entre o fosfato (-PO_4^{2-}) de um nucleotídeo e a hidroxila (-OH) do outro.

[8] através de ligações de hidrogênio.

[9] pois esses grupos estão ligados aos átomos de carbono $5'$ e $3'$ da desoxirribose, respectivamente (seguindo a numeração da figura 2).

membros de cada par de bases somente se encaixam na dupla hélice se as duas fitas forem antiparalelas: se uma delas está na orientação $5' \rightarrow 3'$, então a fita complementar está na orientação $3' \rightarrow 5'$[11].

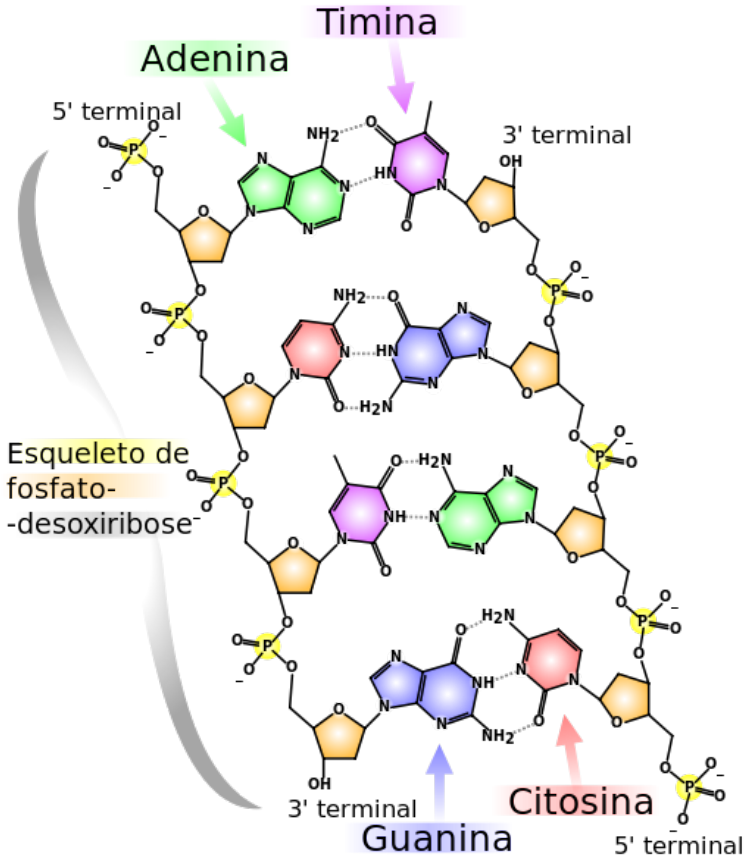

Figura 3: Estrutura química da molécula de DNA, mostrando a complementaridade das bases (ligações de hidrogênio aparecem em pontilhado), as ligações entre os nucleotídeos e o antiparalelismo entre as fitas. Fonte: [13].

2.2 Duplicação do DNA

A cada divisão celular, a célula deve copiar seu genoma e passá-lo para as duas células filhas. Como cada fita de DNA contém uma sequência de nucleotídeos complementar à fita associada, cada fita pode atuar como um molde para a síntese de uma nova fita complementar. Isso é possível pois as ligações entre os pares de bases são fracas quando comparadas às ligações açúcar-fosfato, permitindo que as duas fitas de DNA sejam separadas sem que ocorram danos aos seus esqueletos [11]. Como cada uma das novas moléculas de DNA possui uma fita da molécula original, a duplicação do DNA é dita semiconservativa[14].

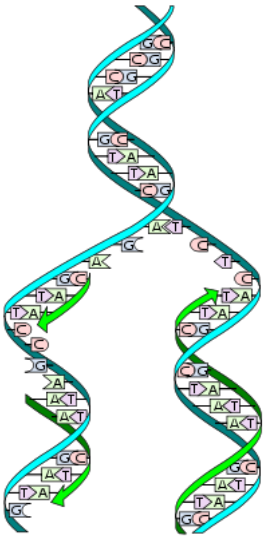

Figura 4: Duplicação semiconservativa do DNA. A dupla-hélice da molécula original (em azul) é desenrolada e cada uma das fitas serve como molde para a síntese de novas fitas complementares (em verde). Fonte: [15].

A polimerização de DNA é catalisada pela enzima DNA-polimerase. Os nucleotídeos livres que servem como substrato para essa enzima são os trifosfatos de desoxirribonucleosídeo (dATP, dTTP, dCTP e dGTP), e sua polimerização requer um molde de DNA de fita simples[11]. Para cada nucleotídeo incorporado à fita em formação é liberado um íon pirofosfato (PP_i, cuja fórmula é $P_2O_7^{4-}$), que é posteriormente hidrolisado em dois íons fosfato [16].

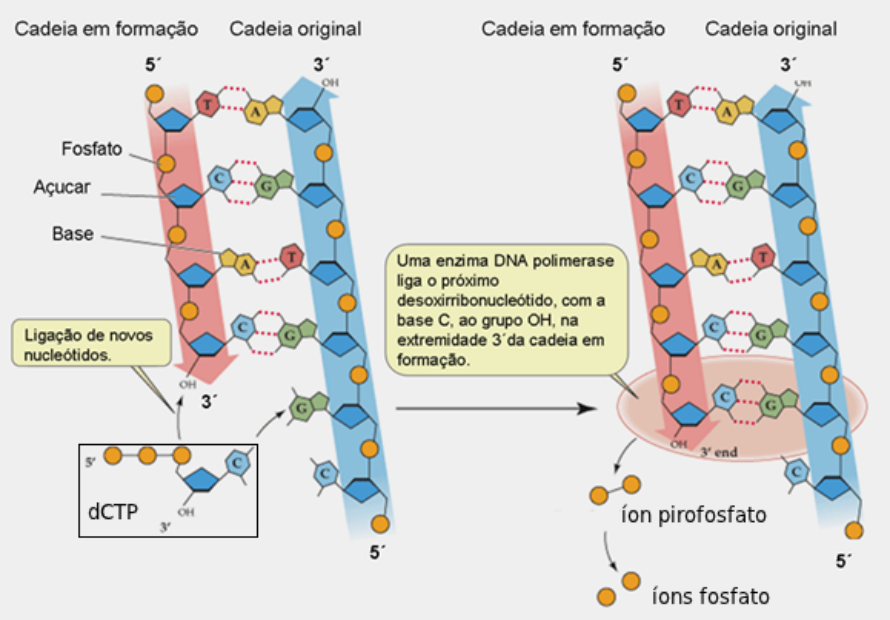

Figura 5: Química da síntese de DNA. A incorporação do desoxirribonucleosídeo trifosfato (dCTP, nesse caso) é sempre feita no terminal $3'$ da nova fita, e é guiada pelo pareamento entre as bases (C e G, nesse caso). Fonte: [17].

Porém, as DNA polimerases possuem uma limitação: elas apenas conseguem estender uma fita de DNA já existente que esteja pareada com a fita molde (ou seja, ela não consegue começar a síntese de uma nova fita). Para começar a síntese, um fragmento curto de DNA ou RNA (chamado de iniciador ou *primer*) deve ser criado e pareado com a fita de DNA molde. Após essa etapa, a DNA polimerase sintetiza uma nova fita de DNA estendendo o terminal $3'$ do iniciador[15].

2.3 Transcrição do RNA

O DNA, além de cumprir sua função como armazenador de informação, também deve ser capaz de expressá-la, guiando a síntese de outras moléculas na célula. O início desse processo é denominado transcrição, no qual segmentos da sequência de DNA (os genes) são usados como moldes para guiar a síntese de polímero de ácido ribonucleico, ou RNA. No RNA, o esqueleto é formado por um açúcar ligeiramente diferente daquele do DNA (ribose em vez de desoxirribose), e uma das quatro bases também é diferente (uracila (U) no lugar de timina (T)). Apesar disso, as três outras bases (A, C e G) são as mesmas, e os 4 tipos de bases do RNA (A, U, C e G) pareiam com os 4 tipos de bases complementares no DNA (T, A, G e C, respectivamente) [11].

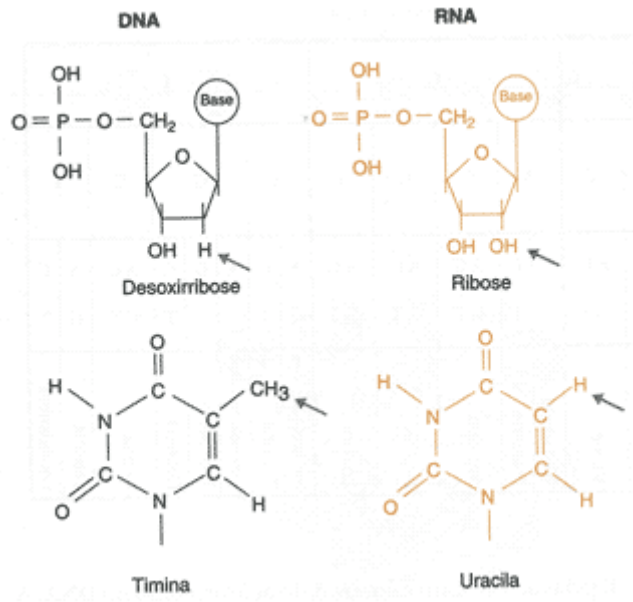

Figura 6: Diferenças químicas (indicadas por setas) entre DNA (coluna esquerda) e RNA (coluna direita). Fonte: [18].

Apesar das pequenas diferenças químicas, o DNA e o RNA diferem drasticamente em termos de estrutura. Enquanto o DNA sempre ocorre nas células sob a forma de uma hélice de fita dupla, o RNA se apresenta como fita simples. Assim, as cadeias de RNA podem se dobrar de diversas

formas (adotando estruturas tridimensionais complexas), o que permite que algumas moléculas de RNA desempenhem funções estruturais e catalíticas[11].

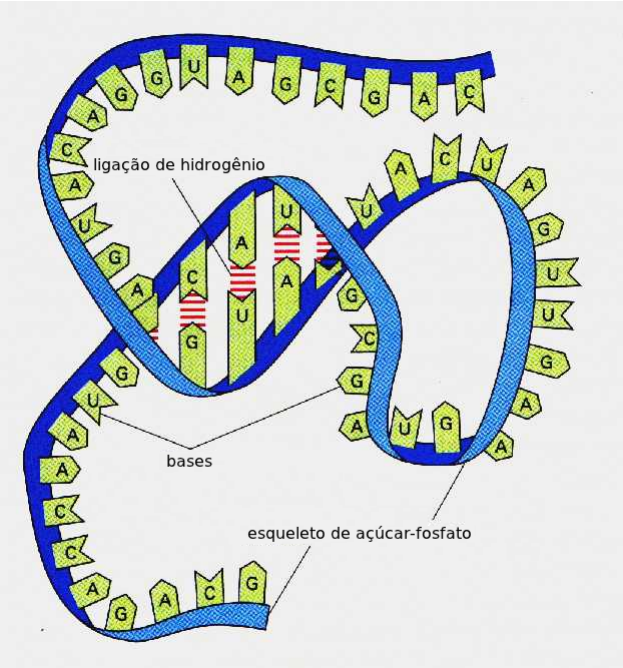

Figura 7: A conformação de uma molécula de RNA. O pareamento de nucleotídeos entre diferentes regiões da mesma fita de RNA faz com que a molécula adquira uma conformação distinta. Fonte: [19].

Todo o RNA de uma célula é produzido pela transcrição de DNA, num processo semelhante ao da duplicação de DNA. A transcrição começa com a abertura e a desespiralização de uma pequena porção da dupla-hélice de DNA, o que expõe as bases em cada fita. Apenas uma das duas fitas (a fita molde) age como um molde para a síntese de uma molécula de RNA[10]. Tal como na duplicação de DNA, a sequência de nucleotídeos do RNA é determinada pelo pareamento de bases entre os trifosfatos de ribonucleosídeo (ATP, UTP, CTP e GTP) a serem incorporados e a fita de DNA molde. Quando um pareamento adequado é estabelecido, o ribonucleotídeo a ser incorporado é covalentemente ligado à cadeia de RNA em formação[11], por meio de uma reação catalisada pelas enzimas RNA-polimerases. Porém, ao contrário das DNA polimerases, as RNA polimerases podem começar a síntese de uma nova cadeia de RNA sem um iniciador[11].

[10] a outra fita é denominada fita codificante, pois sua sequência de bases é idêntica à sequência do RNA a ser sintetizado (exceto pelas timinas, que não existem no RNA).

[11] e uma molécula de pirofosfato é liberada, assim como na duplicação de DNA.

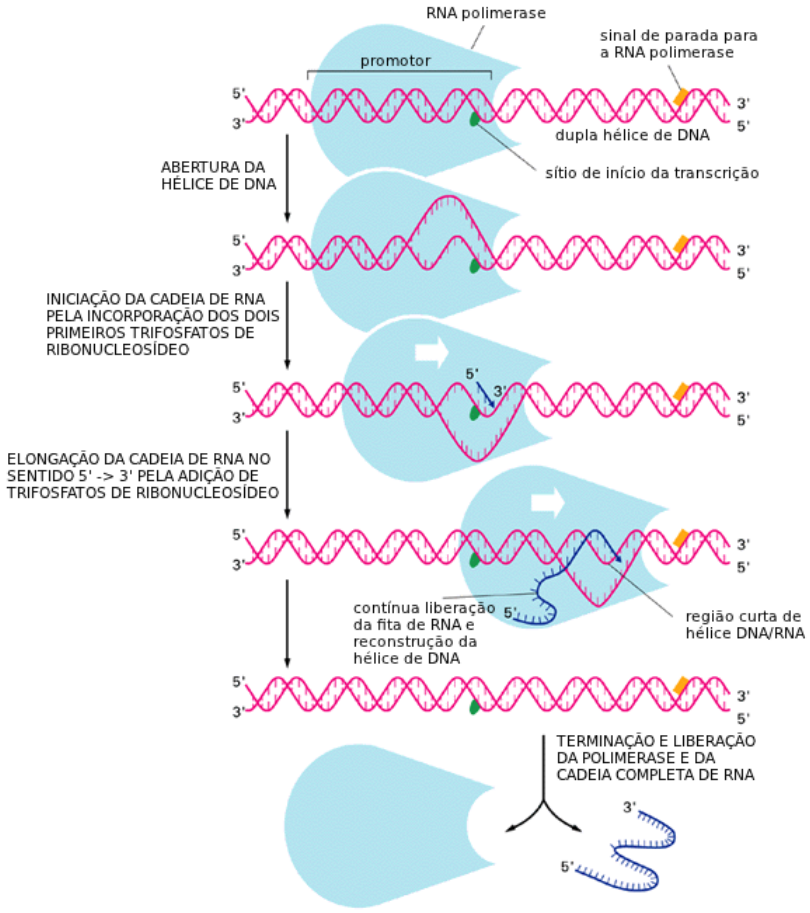

Figura 8: Transcrição do DNA pela RNA-polimerase. O promotor é a região de DNA à qual a RNA-polimerase se liga para iniciar a transcrição. Fonte: [20].

2.3.1 *Splicing* do pré-RNAm

A maioria dos genes presentes no DNA das células especifica a sequência de aminoácidos de proteínas; as moléculas de RNA que são transcritas a partir deles são denominadas moléculas de RNA mensageiro (RNAm)[12]. Em eucariontes, entretanto, nem toda a molécula de RNA codifica aminoácidos: os genes eucarióticos são encontrados sob a forma de pequenas sequências codificantes (éxons) intercaladas por sequências longas (íntrons) não codificantes[11].

Tanto as sequências de íntrons quanto de éxons são transcritas em RNA, sendo que os íntrons são removidos do RNAm recentemente sintetizado (pré-RNAm) por um processo denominado *splicing* de RNA. A maquinaria celular responsável (o spliceossomo) reconhece as divisões entre os íntrons e éxons (chamadas de sítios de *splice*) através de sequências de consenso, que são similares entre os íntrons e fornecem indícios para a célula do local onde deve ocorrer o *splicing*. Na maioria dos íntrons, tais sequências (no pré-RNAm) são GU (sítio doador, que fica na extremidade $5'$ do

[12]contudo, o produto final de uma minoria de genes é o próprio RNA.

íntron) e AG (sítio aceptor, que fica na extremidade $3'$ do íntron)[21].

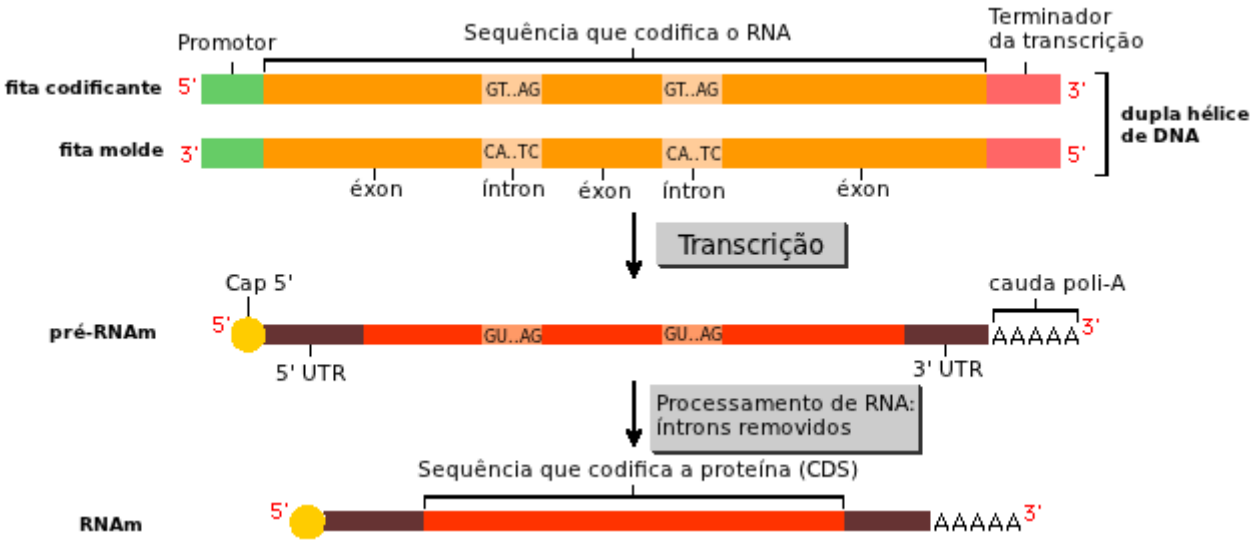

Figura 9: Transcrição e *splicing* do pré-RNAm eucariótico, mostrando as divisões entre éxons e íntrons. O *cap* $5'$ e a cauda poli-A são modificações adicionais que conferem maior estabilidade ao RNAm, e as UTRs são sequências não traduzidas em proteína. Fonte: modificado de [22].

2.4 Tradução do RNA

As moléculas de proteína, assim como as de DNA e RNA, são cadeias poliméricas longas não ramificadas, cujos monômeros (de 20 tipos diferentes) são os aminoácidos. Cada aminoácido possui uma estrutura básica (por meio da qual se liga a outros aminoácidos) e uma cadeia lateral (variável) que atribui a cada um uma característica distinta. Cada uma das moléculas de proteína, ou polipeptídeos, dobra-se para adquirir uma forma tridimensional precisa, com sítios reativos em sua superfície. Dessa forma, elas desempenham várias funções na célula, como catálise de reações químicas (enzimas), manutenção de estruturas, geração de movimentos, percepção de sinais e assim por diante[11].

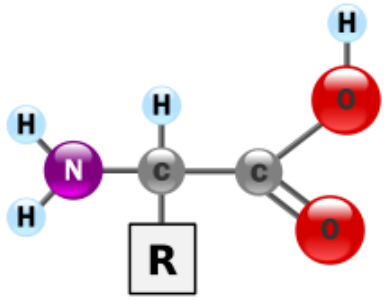

Figura 10: Estrutura genérica de um aminoácido, mostrando o grupo amino ($-NH_2$), o grupo carboxila (-COOH) e a cadeia lateral (-R, diferente para cada aminoácido). Fonte: [23].

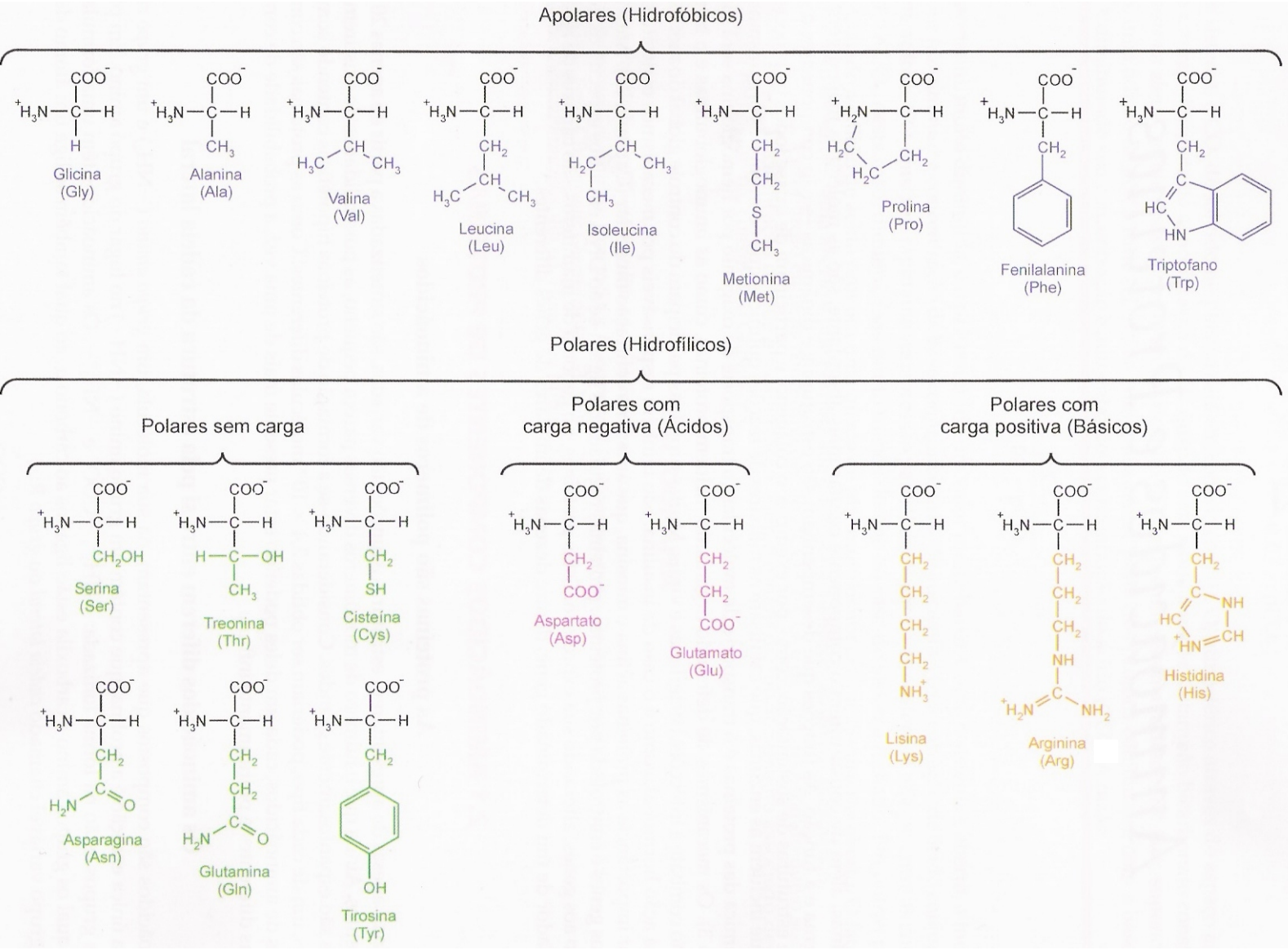

Figura 11: Os 20 aminoácidos que compõem as proteínas, classificados segundo as características de suas cadeias laterais. Fonte: [24, 25].

A informação contida na parte codificante (CDS) do RNAm é lida em trincas de nucleotídeos (códons), com cada trinca especificando um único aminoácido na proteína correspondente. Como existem 61 ($= 4 \times 4 \times 4 - 3$) códons codificantes possíveis[13] e apenas 20 aminoácidos, vários códons codificam o mesmo aminoácido[14][11].

[13] 3 códons (UAA, UAG e UGA) não especificam aminoácidos, e sim o final da tradução.

[14] e por isso se diz que o código genético (associação entre códons e aminoácidos) é degenerado.

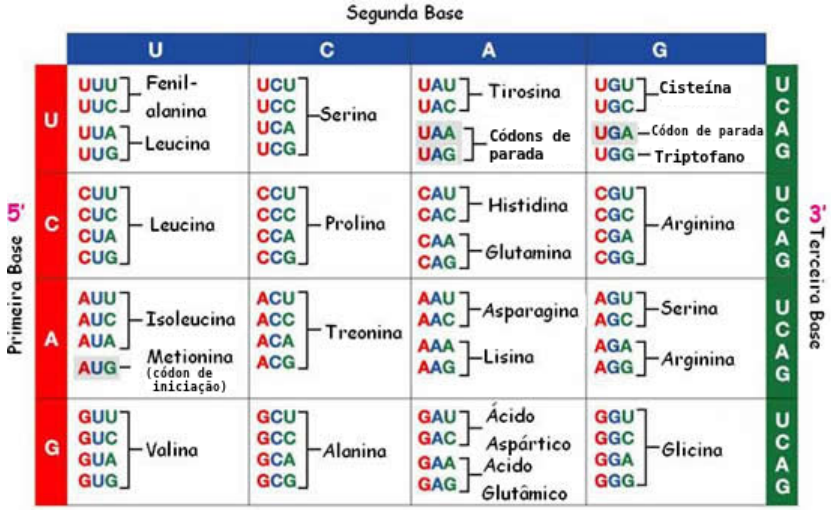

Figura 12: O código genético. O corpo da tabela mostra as associações entre os códons e os aminoácidos. Fonte: [26].

O código é lido por RNAs denominados RNA transportadores (RNAts). Cada tipo de RNAt liga-se a uma extremidade de um aminoácido específico, possuindo (em outra extremidade) uma sequência de três nucleotídeos (o anticódon) que o permite reconhecer (por pareamento de bases) um códon[15] no RNAm [27].

Para a síntese proteica, os anticódons dos RNAts (carregados com seus respectivos aminoácidos) emparelham-se com seus códons, os aminoácidos são utilizados para alongar a cadeia nascente de proteína e os RNAts descarregados são liberados. Esse conjunto de processos (que se inicia com o reconhecimento do códon de iniciação no RNAm e termina com o reconhecimento de um dos três códons de parada) é realizado pelo ribossomo, que é um complexo formado por diversas moléculas de RNA (RNAs ribossomais ou RNArs) e mais de 50 proteínas diferentes. A reação fundamental para a síntese de proteínas é a formação de uma ligação peptídica entre o grupo carboxila na extremidade da cadeia polipeptídica em crescimento e um grupo amino livre do novo aminoácido [11].

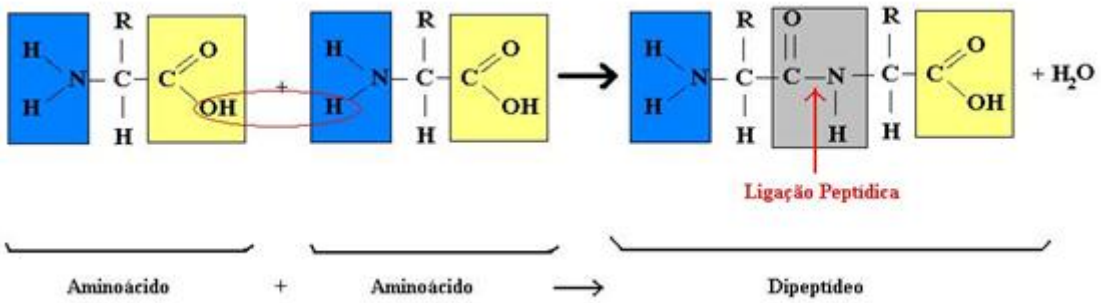

Figura 13: Formação da ligação peptídica, catalisada pelo ribossomo. O grupo amino aparece em azul e o grupo carboxila em amarelo. Fonte: [28].

[15] ou um grupo de códons.

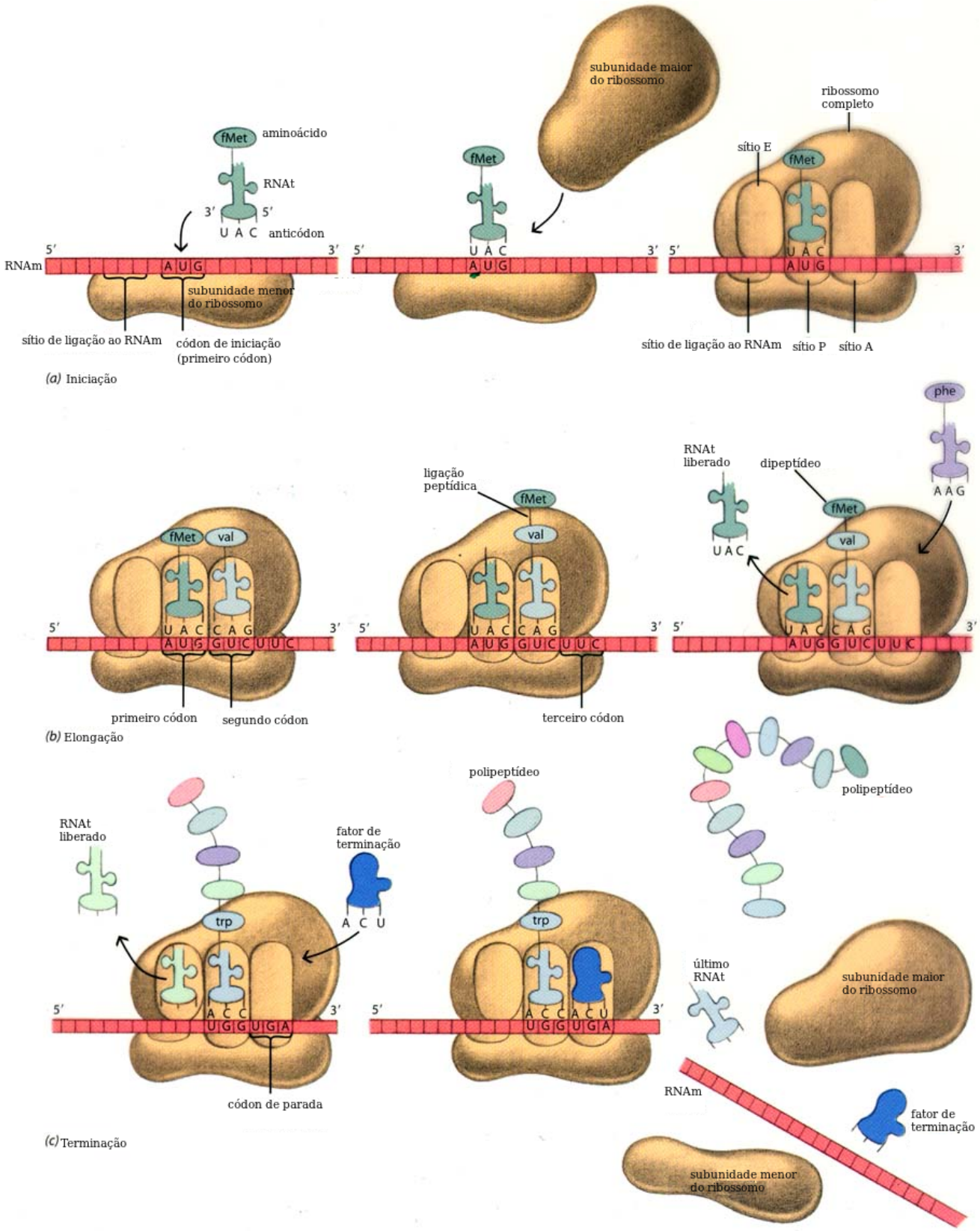

Figura 14: As três fases da tradução do RNAm: iniciação, elongação e terminação. O RNAm é lido no sentido $5' \rightarrow 3'$, e a proteína é gerada no sentido N-terminal $\rightarrow$ C-terminal (ou seja, o primeiro aminoácido possui o grupo amino livre e o último possui o grupo carboxila livre). Fonte: [29].

3 Sequenciamento de genomas

3.1 Estratégias

Existem dois modos de se sequenciar um genoma. O método BAC a BAC (ou *shotgun* hierárquico), o primeiro a ser usado nos estudos do genoma humano, é lento mas preciso. Também conhecido como método baseado em mapeamento, ele evoluiu a partir de procedimentos desenvolvidos nas décadas de 1980 e 1990, e continua a ser aperfeiçoado. A outra técnica (conhecida como sequenciamento *shotgun*) é muito mais rápida (permitindo que os pesquisadores realizem a tarefa em meses), mas menos precisa. Ela foi desenvolvida por J. Craig Venter em 1996, no Instituto para a Pesquisa Genômica (TIGR)[30].

3.1.1 Sequenciamento *shotgun*

Esse é um método usado para sequenciar fitas longas de DNA, assim chamado pela analogia com o padrão de tiro quase aleatório de uma espingarda (*shotgun*, em inglês). Várias cópias do DNA são clivadas aleatoriamente em vários fragmentos pequenos, que são então sequenciados para obter *reads*. Em seguida, programas de computador montam as sequências utilizando as sobreposições entre os terminais dos *reads* [31]. O método *shotgun* é mais rápido e mais barato, mas mais propício a erros por ter que lidar com um número muito grande de fragmentos [32].

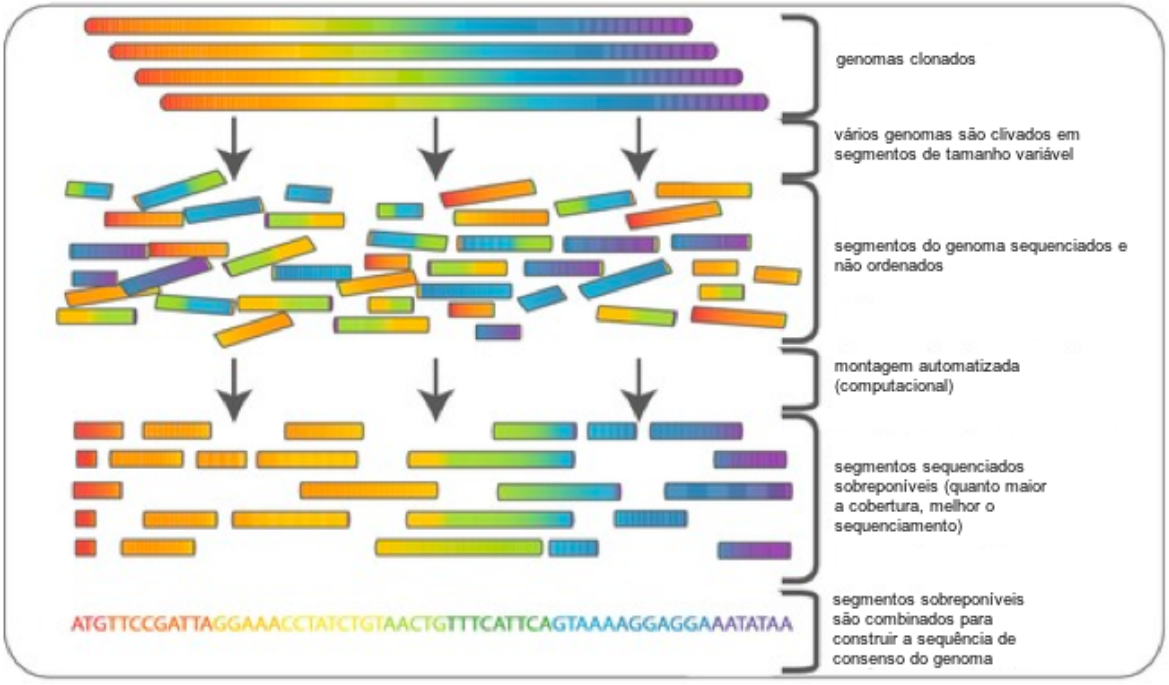

Figura 15: As fases do sequenciamento *shotgun*. As cópias do genoma são clivadas aleatoriamente em fragmentos pequenos (apropriados para o sequenciamento) e então montados. Fonte: [31].

3.1.2 Sequenciamento BAC a BAC (*shotgun* hierárquico)

O método BAC a BAC é um método bem estabelecido de sequenciamento, mas tende a ser muito lento [33]. Em primeiro lugar, um mapa físico de baixa resolução do genoma é feito antes do sequenciamento[31]. Isso requer dividir os cromossomos em grandes pedaços e descobrir qual a ordem deles no genoma [30].

Em seguida, várias cópias do genoma são cortadas aleatoriamente em fragmentos de 50-200 kb (insertos), que são inseridos em BACs e transferidos para bactérias[30, 31]. A coleção completa dos BACs contendo o genoma é dita uma biblioteca de BACs, pois cada BAC é como se fosse um livro que pode ser acessado e copiado [30].

Na maioria dos projetos, ambos os terminais de cada inserto são então sequenciados, definindo um par de *reads* para cada BAC (chamado de *mate pair*). Esses pares podem ser usados tanto durante o processo de montagem de cada BAC como após dele, para ordenar os *contigs* resultantes da montagem dos BACs[5].

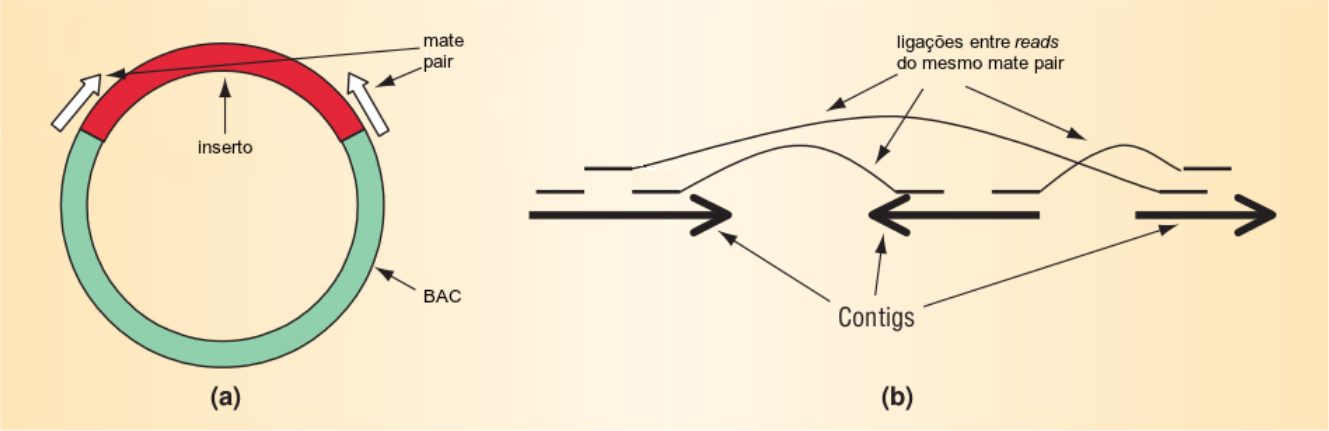

Figura 16: (a) Os insertos do BAC são sequenciados de ambos os lados, gerando *mate pairs*. (b) Os *mate pairs* podem ser usados para ordenar e orientar os *contigs* no genoma que está sendo montado. Fonte: [5].

Como múltiplas cópias do genoma foram clivadas aleatoriamente, os insertos possuem terminais diferentes e, com cobertura suficiente, é teoricamente possível achar (utilizando o mapa físico construído inicialmente) um conjunto de *contigs* de BACs (chamado de *tiling path*) que cubra todo o genoma. Em seguida, cada um dos BACs que forma o *tiling path* pode ser clivado aleatoriamente e sequenciado (ou seja, é feito um sequenciamento *shotgun* para cada BAC) [31]. A principal vantagem do método é a precisão, já que a localização cromossômica de cada BAC é conhecida e o número de *reads* que precisa ser montado é menor [32].

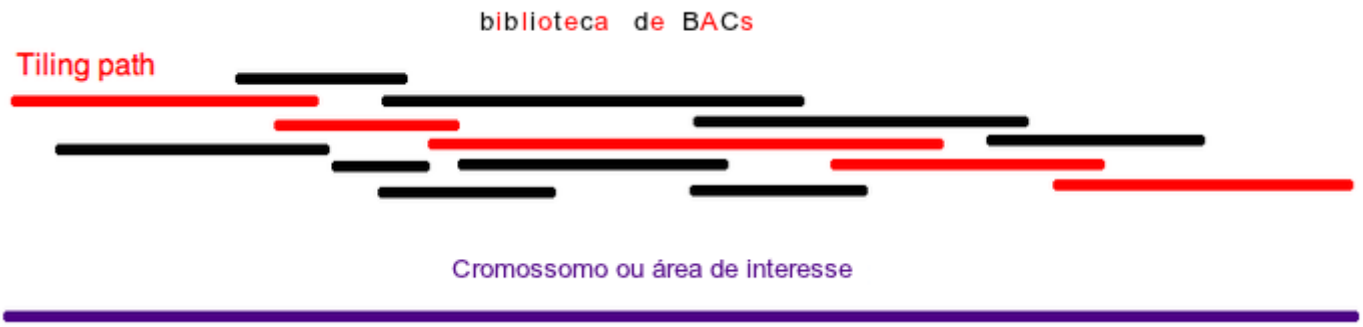

Figura 17: Um conjunto de BACs que cobre toda a área genômica de interesse constitui um *tiling path*. Fonte: [31].

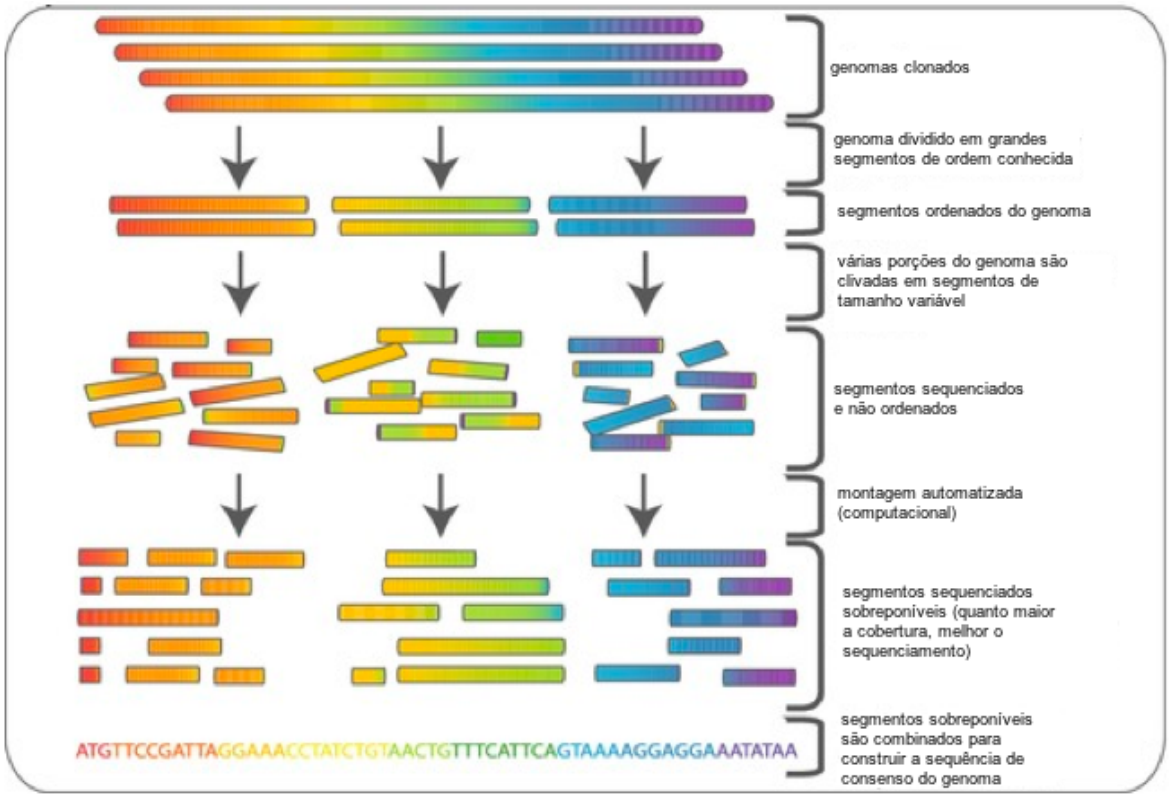

Figura 18: As fases do sequenciamento *shotgun* hierárquico. O genoma é clivado em grandes segmentos e, após a ordem entre eles ser deduzida, esse segmentos são clivados novamente em tamanhos apropriados para o sequenciamento. Fonte: [31].

3.2 Pirossequenciamento

A seguir será descrita a tecnologia utilizada pelo sequenciador Roche/454 (*Genome Sequencer* FLX™), que foi utilizado para gerar os *reads* que foram utilizados nesse trabalho.

3.2.1 Passo 1: preparação das amostras de DNA (duração: 4 a 5h)

O primeiro passo é a fragmentação do DNA genômico em fragmentos de 400 a 600 pb (nebulização), seguida do polimento (isto é, fazer com que ambas as pontas sejam terminais cegos)[34, 35]. Em seguida, são anexados dois tipos (A e B) de adaptadores (pequenas moléculas de DNA cujas sequências são conhecidas, que são complementares aos iniciadores presentes entre os reagentes) aos terminais dos fragmentos[34–36]. Finalmente, os fragmentos de fita dupla são separados em fitas simples, criando uma biblioteca de DNA molde de fita simples (*single-stranded template DNA* (sstDNA) *library*) [34].

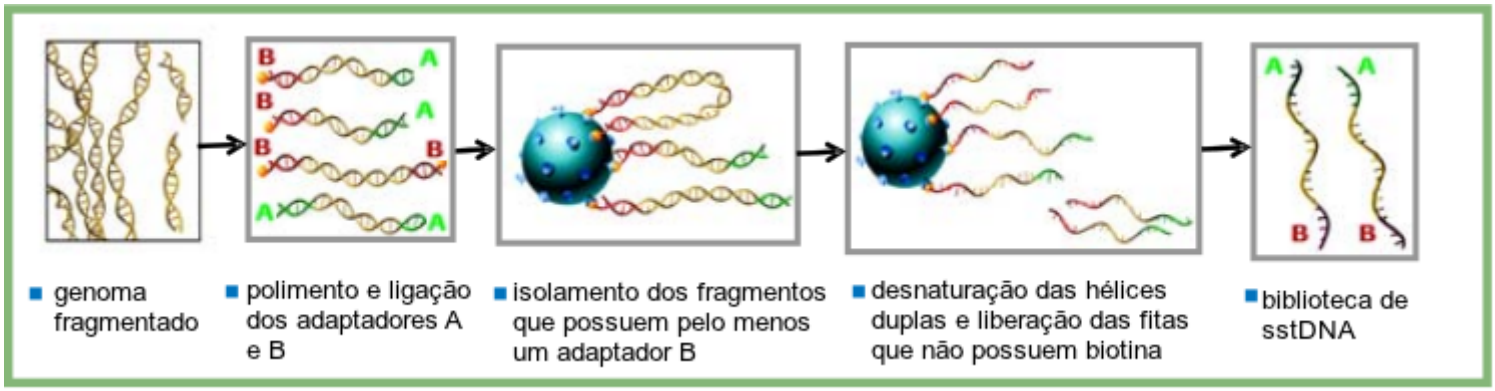

Figura 19: As fases do passo 1. Uma das fitas do adaptador B contém uma molécula (biotina, representada por uma esfera amarela) reconhecida pelo *bead* de captura (em azul). Se o fragmento capturado tiver o adaptador A na outra ponta, uma das fitas será liberada após a desnaturação. As fitas liberadas (que contêm os adaptadores A e B) compõem a biblioteca de sstDNA utilizada no sequenciamento [35]. Fonte: [34].

3.2.2 Passo 2: PCR em emulsão (emPCR) (duração: 8h)

Primeiramente, uma mistura aquosa (contendo os fragmentos da biblioteca de sstDNA, *beads* de captura e os reagentes para a PCR) são injetados em pequenos contêineres de plástico contendo um óleo sintético. Após agitação, o resultado é uma emulsão água em óleo, com as gotículas de água envolvendo os *beads*. Na maioria dos casos, cada gotícula de água terá apenas um *bead* e um único fragmento da biblioteca de sstDNA. Em seguida é iniciada uma reação conhecida como PCR, que faz com que cada fragmento de cada gotícula seja amplificado em milhões de cópias que ficam imobilizadas nos *beads*. Ao término da reação, os *beads* são isolados do óleo (rompimento da emulsão) e limpos. Os que não contêm DNA são eliminados, e os que possuem mais de um tipo de fragmento são descartados durante o processamento do sinal gerado na fase de sequenciamento[34].

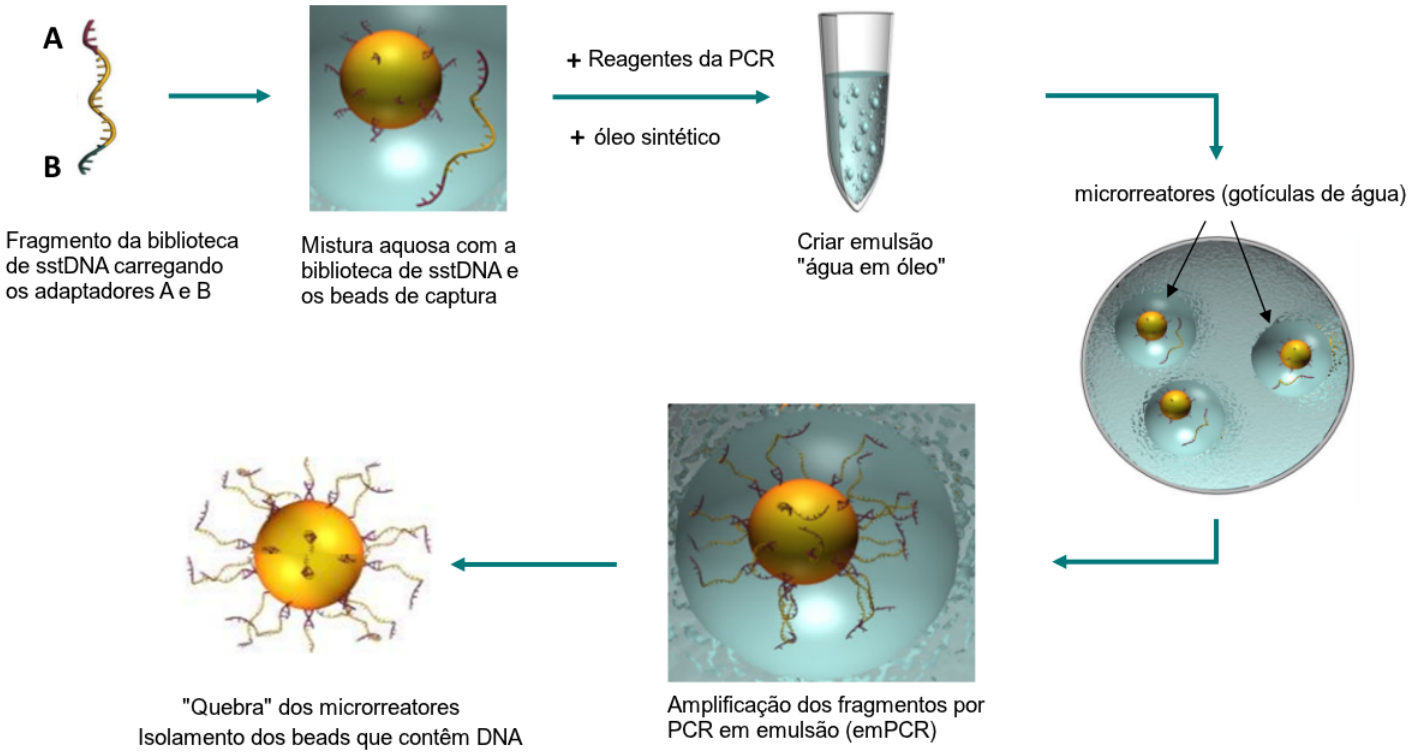

Figura 20: As fases do passo 2. O resultado é a produção de milhões de cópias do mesmo fragmento imobilizadas no *bead* (enriquecimento da amostra) [36]. Fonte: [37].

3.2.3 Passo 3: sequenciamento (duração: 7,5h)

A abordagem utilizada é o "sequenciamento por síntese", na qual a sequência de uma molécula de DNA de fita simples é deduzida a partir da detecção dos nucleotídeos incorporados na síntese da fita complementar [38]. Os *beads* de captura de DNA resultantes do passo 2 são colocados numa placa de sequenciamento (*PicoTiterPlate*™), que possui 1,6 milhões de poços. O diâmetro dos poços é projetado para que cada um deles possua apenas um *bead*. Em seguida são adicionados os *beads* enzimáticos (que possuem as enzimas - ATP sulfurilase e luciferase - utilizadas nas reações que detectam a incorporação de nucleotídeos) e a mistura de incubação dos *beads* (contendo DNA polimerase)[34, 36].

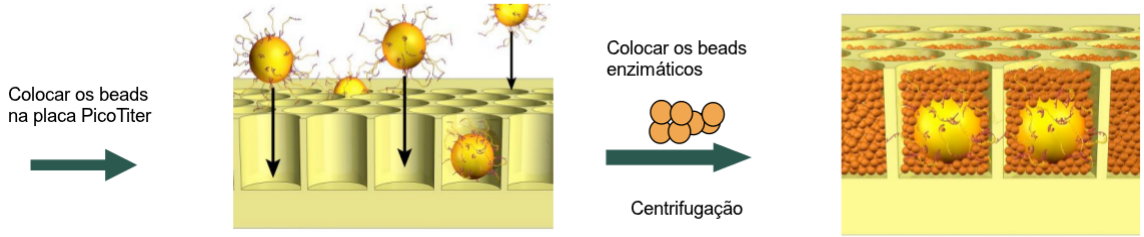

Figura 21: Início do passo 3. Cada um dos poços da placa possui apenas um *bead* com milhões de cópias do mesmo sstDNA e vários *beads* enzimáticos. Fonte: [37].

Em seguida, o sistema fluídico do sequenciador deposita os substratos da DNA polimerase (dNTPs, correspondentes às bases T, A, C e G) na placa, sequencialmente e na mesma ordem (de forma cíclica), de modo que apenas um tipo de dNTP esteja presente na placa por vez (o que permite descobrir qual deles foi incorporado). Quando um desses nucleotídeos é incorporado às fitas de DNA, as enzimas dos *beads* enzimáticos convertem o pirofosfato (PP_i) liberado[16] em luz, numa reação quimiluminescente semelhante à dos vaga-lumes. A intensidade da luz determina se um mesmo tipo de dNTP foi incorporado mais de uma vez na mesma rodada, e os *beads* têm suas cópias do DNA sequenciadas em paralelo [34, 36, 38].

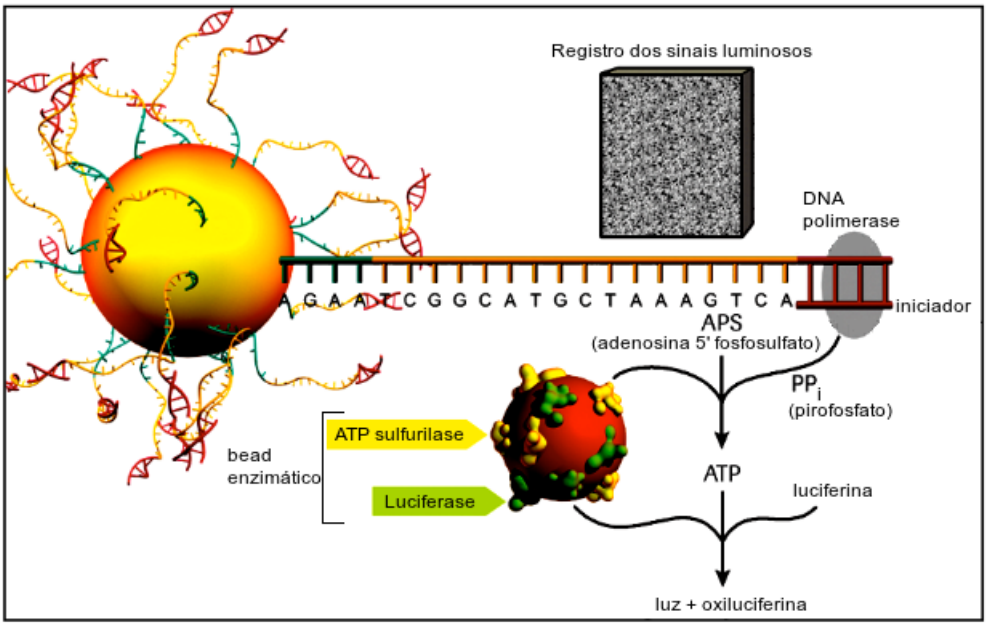

Figura 22: Sequência de reações que faz com que a incorporação de nucleotídeos seja detectável (através do sinal luminoso gerado no final). O excesso de dNTPs e ATPs é degradado por outra enzima, a apirase (não mostrada) [38, 39]. Fonte: [37].

O sinal luminoso produzido é detectado por uma câmera CCD (*charge-coupled device*), que usa um pequeno pedaço retangular de silício (o CCD) para receber luz (em vez de um filme). A intensidade da luz gerado durante o fluxo de um único tipo de nucleotídeo varia de modo proporcional ao número de nucleotídeos complementares ao fragmento de DNA de fita simples sendo analisado (e.g., se existirem 3 A's seguidos, então a intensidade será 3 vezes maior do que se existisse um único A). Os sinais criados no processo de sequenciamento são analisados para gerar milhões de bases sequenciadas por hora. As imagens são processadas para obter um gráfico de barras (que registra a intensidade de luz para cada tipo de nucleotídeo) denominado pirograma, que permite obter a sequência correspondente ao fragmento original de sstDNA (é gerado um pirograma para cada poço). No fim, é válida a relação "1 fragmento sstDNA : 1 *bead* : 1 poço : 1 pirograma : 1 *read*"[34]. Em média, os *reads* obtidos possuem tamanho de 700 pares de bases (pb) [2].

[16] conforme explicado na seção 2.2.

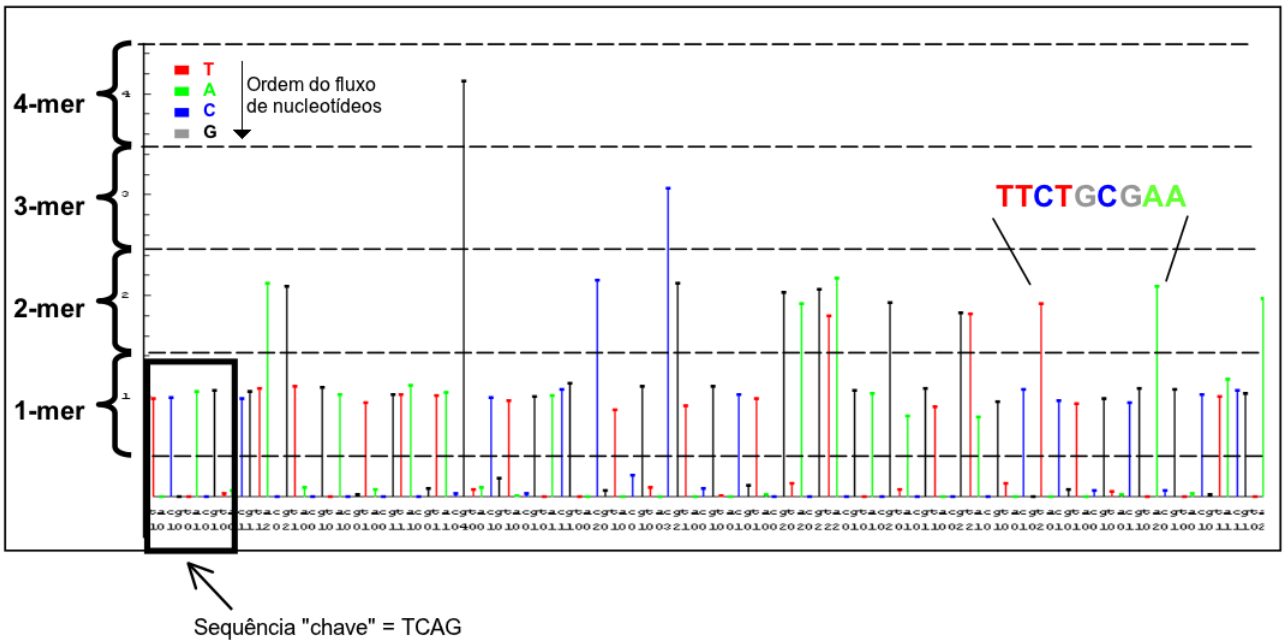

Figura 23: Um pirograma. O eixo horizontal indica qual foi o nucleotídeo incorporado, e o vertical (dividido em regiões chamadas k-mers) a intensidade da luz detectada. As linhas tracejadas indicam os limiares de classificação: se a intensidade detectada correspondente ao nucleotídeo X estiver na região n-mer, assume-se que existem n nucleotídeos X seguidos na sequência. A sequência "chave" (TCAG) está presente nos adaptadores (A e B, cujas sequências são conhecidas), e é utilizada para calibrar o sinal [34, 40, 41]. Fonte: [34].

4 Alinhamento de sequências

4.1 Definição

O alinhamento de sequências consiste em comparar duas (alinhamento par a par) ou mais (alinhamento múltiplo) sequências (de nucleotídeos ou aminoácidos) pela procura de caracteres que aparecem na mesma ordem. O alinhamento consiste em escrever as sequências em duas linhas distintas, colocando os pares de caracteres alinhados em colunas (lacunas (*gaps*) - indicadas por "–" - também podem ser inseridas). Num alinhamento ótimo, os caracteres não idênticos e as lacunas são posicionados de forma a fazer com que mais colunas possuam caracteres idênticos. Sequências que podem ser facilmente alinhadas dessa forma (com várias colunas de caracteres idênticos) são ditas *similares* [42].

Cada tipo de coluna (duas lacunas, uma lacuna, dois caracteres idênticos, dois caracteres distintos, etc.) recebe uma determinada *pontuação*, estabelecida *a priori* (tipicamente positiva para colunas idênticas e negativa para outros tipos). A pontuação do alinhamento é definida pela soma

da pontuação de cada coluna, e um *alinhamento ótimo*[17] entre duas sequências é o que possui pontuação máxima[43].

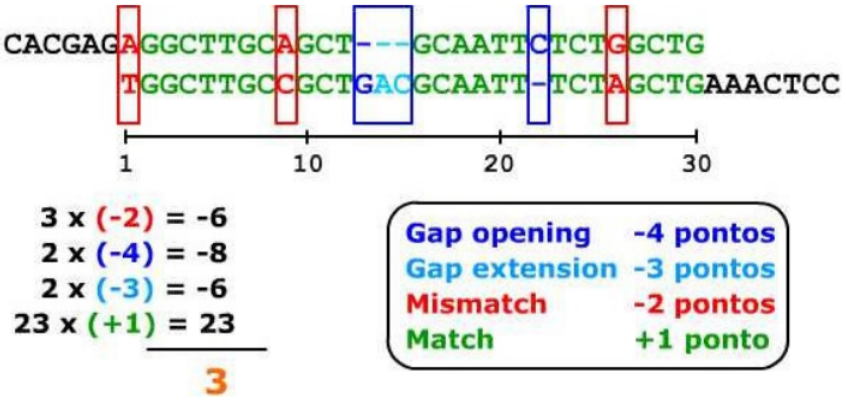

Figura 24: Exemplo de um alinhamento semiglobal com pontuação igual a 3 (em laranja). Os diferentes tipos de coluna estão indicados pelas cores: verde para caracteres idênticos (*matches*), vermelho para caracteres distintos (*mismatches*), azul escuro para abertura de lacunas (*gap opening*) e azul claro para extensão de lacunas (*gap extension*). Fonte: [44].

Existem muitas aplicações do alinhamento de sequências, como identificação de genes e proteínas desconhecidas, comparação da ordem relativa entre os genes em genomas proximamente relacionados (sintenia) e montagem de sequências (para achar sobreposições entre as sequências, o que permite formar os *contigs*)[44, 45]. De modo geral, esse processo serve para identificar regiões de similaridade, que podem ser consequências de relações funcionais, estruturais ou evolutivas entre as sequências [45].

4.2 Medidas (identidade e cobertura)

Além da pontuação, duas medidas que podem ser utilizadas para avaliar a qualidade de um alinhamento são a identidade (do alinhamento todo) e a cobertura (de cada sequência). A *identidade* do alinhamento é a porcentagem de colunas idênticas[46], enquanto a *cobertura* de uma sequência é a porcentagem de caracteres presentes na região alinhada.

Como exemplo[18], consideremos as sequências $s = \texttt{QUERIDAROSAVERMELHA}$ ($|s| = 19$), $t = \texttt{QUEROUMAMOROSOVERME}$ ($|t| = 19$) e o alinhamento local $a = \left\{\begin{smallmatrix}\text{ROSAVERME}\\ \text{ROSOVERME}\end{smallmatrix}\right\}$. Temos que cobertura$(s, a)$ = $|\texttt{ROSAVERME}|$ / $|s| = 9/19 \approx 47\%$, cobertura(t, a) = $|\texttt{ROSOVERME}|$ / $|t| = 9/19 \approx 47\%$ e identidade$(a) = 8/9 \approx 89\%$.

[17] pode haver mais de um.
[18] retirado de [47].

4.3 Tipos de alinhamentos

A seguir serão explicados alguns dos tipos de alinhamentos existentes e suas aplicações. Os exemplos foram retirados de [47].

4.3.1 Alinhamento global

O alinhamento global é o que compara duas sequências ao longo de toda a sua extensão, de modo a incluir o maior número possível de colunas idênticas[42, 44]. Como exemplo, aqui está um alinhamento global entre as sequências `QUERIDAROSAVERMELHA` e `QUEROUMAMOROSOVERME` (barras verticais indicam colunas idênticas):

```
QUERIDA---ROSAVERMELHA
||||       ||| |||||
QUEROUMAMOROSOVERME---
```

O algoritmo que encontra esse tipo de alinhamento é o de Needleman-Wunsch, e ele é comumente utilizado para identificar genes ou proteínas com funções similares (ambas as sequências são tratadas como potencialmente equivalentes) [48].

4.3.2 Alinhamento local

O alinhamento local acontece quando a comparação entre as sequências não é feita ao longo de toda sua extensão, mas entre suas subsequências[43, 44]. O alinhamento para no final de regiões altamente similares, e encontrá-las possui uma prioridade maior do que maximizar o número de colunas idênticas (ou semelhantes) [42]. Como exemplo, aqui estão *dois* alinhamentos locais entre as sequências `QUERIDAROSAVERMELHA` e `QUEROUMAMOROSOVERME`:

```
QUER         ROSAVERME
||||    e    ||| |||||
QUER         ROSOVERME
```

O algoritmo que encontra esse tipo de alinhamento é o de Smith-Waterman, e ele é comumente utilizado para detectar padrões de nucleotídeos ou aminoácidos (domínios proteicos) conservados[42, 48].

4.3.3 Alinhamento semiglobal

Numa comparação semiglobal, as lacunas terminais (à esquerda do primeiro caractere ou à direita do último caractere de uma das sequências) são ignoradas (ou seja, colunas com lacunas desse tipo possuem pontuação nula)[43]. Como exemplo, aqui está um alinhamento semiglobal (em que todas as lacunas são terminais) entre as sequências `ROSAVERMELHA` e `AMOROSOVERME`:

```
---ROSAVERMELHA
   ||| |||||
AMOROSOVERME---
```

O algoritmo que encontra esse tipo de alinhamento é uma modificação do algoritmo de Smith-Waterman, e ele é comumente utilizado na montagem de sequências (para encontrar as sobreposições entre os *reads*)[45, 48].

4.4 Alinhamento heurístico

Os algoritmos de Smith-Waterman e Needleman-Wunsch possuem uma garantia de conseguir encontrar o alinhamento ótimo (para um dado esquema de pontuação) entre um par de sequências, mas são ineficientes para sequências longas (ambos são algoritmos de programação dinâmica que possuem consumo de tempo e espaço $\mathcal{O}(mn)$, sendo m e n os tamanhos das sequências)[42, 49]. Por isso existem os algoritmos ditos *heurísticos*, que não necessariamente encontram o alinhamento ótimo mas são mais eficientes[46].

Uma heurística possível (presente em algoritmos como o BLAST[50] e o BLAT[51][19], por exemplo) é a das "palavras" ou "k-tuplas". Ela começa procurando por pares de subsequências de tamanho k (tipicamente, $k = 3$ para sequências de aminoácidos e $k = 11$ para sequências de nucleotídeos) que sejam altamente similares (chamados de "palavras" ou "k-tuplas") e então os incorporam em um alinhamento utilizando programação dinâmica. Os métodos derivados dessa heurística são rápidos o suficiente para buscas em (grandes) bancos de dados por sequências que melhor se alinhem com uma dada sequência de interesse[42, 46, 48].

[19] utilizado para fazer os alinhamentos no *pipeline* desenvolvido neste trabalho.

5 Montagem de sequências

5.1 Definição

A montagem de sequências refere-se ao alinhamento e fusão de fragmentos (os fragmentos fundidos denominam-se *contigs*) vindos de uma molécula de DNA maior para poder reconstruir a sequência original. Isto é necessário pois a tecnologia atual de sequenciamento de DNA não consegue lidar com cromossomos inteiros, mas apenas com pequenos fragmentos (chamados de *reads*) de tamanho entre 20 e 1000 pares de bases [1]. A montagem de um genoma é análoga ao processo de picotar várias cópias idênticas de um livro (cujas palavras e a ordem entre elas sejam completamente desconhecidas) e tentar reconstruir uma das cópias desse livro a partir dos fragmentos[44].

No problema biológico, sabemos o tamanho da sequência a ser montada (a *sequência alvo*) com uma margem de erro de aproximadamente 10%, além da sequência de bases e dos terminais ($5'$ e $3'$) de cada fragmento. O que não sabemos é a posição e a orientação ($5' \rightarrow 3'$ ou $3' \rightarrow 5'$) dos fragmentos na sequência alvo [20][43].

Como exemplo[21] do "caso ideal"[22], suponhamos que a sequência alvo tenha aproximadamente 10 bases e que a entrada seja dada pelos seguintes fragmentos:

```
5′ ACCGT  3′
5′ CGTGC  3′
5′ TTAC   3′
5′ TACCGT 3′
```

Um modo possível de montá-los é através da sequência de consenso de um alinhamento múltiplo (envolve mais de duas sequências) semiglobal (ignora lacunas terminais), como o mostrado a seguir:

```
5′ --ACCGT-- 3′
5′ ----CGTGC 3′
5′ TTAC----- 3′
5′ -TACCGT-- 3′
-------------------
5′ TTACCGTGC 3′
```

[20] sobre estrutura do DNA, veja a seção 2.1.
[21] retirado de [43].
[22] o caso real possui várias complicações, descritas em 5.2.

Os espaços terminais são ignorados pois supostamente representam partes da molécula não cobertos por cada fragmento, sendo que as únicas informações que guiam a montagem (além do tamanho da sequência alvo) são as sobreposições (*overlaps*) entre o prefixo (parte inicial) de um fragmento e o sufixo (parte final) de outro (quanto maior for a sobreposição entre um par de fragmentos, maior será a probabilidade de que tenham vindo da mesma região da sequência alvo). O alinhamento múltiplo formado pelos fragmentos é chamado de *layout*, enquanto a sequência abaixo da linha horizontal é o *consenso* [23] [43].

A sequência de consenso é a aproximação resultante da sequência alvo, e é obtida por "maioria de votos" (isto é, cada base do consenso é a que aparece o maior número de vezes na coluna correspondente do *layout*). Neste exemplo "ideal", todas as colunas são unânimes (aparece apenas uma base em cada coluna), o consenso possui um número de bases (9) próximo ao número conhecido (10) e cada fragmento é uma *substring* do consenso. Isso dificilmente ocorre na prática, devido a uma série de complicações[43].

5.2 Complicações tecnológicas

5.2.1 Erros

Os tipos mais simples de erros ocorrem no processo de *base calling* (chamados de erros de *base call*), e consistem de substituições, inserções e remoções de bases nos fragmentos[43]. A taxa desse tipo de erro varia de 0 a 5%, sendo que eles se concentram na extremidade $3'$ do fragmento (devido a fenômenos como *phasing* e *pre-phasing*)[43, 52]. No caso do sequenciador Roche/454, os erros ocorrem principalmente na limiarização do sinal (feita para determinar quantas bases foram incorporadas em cada ciclo de sequenciamento)[24] [52].

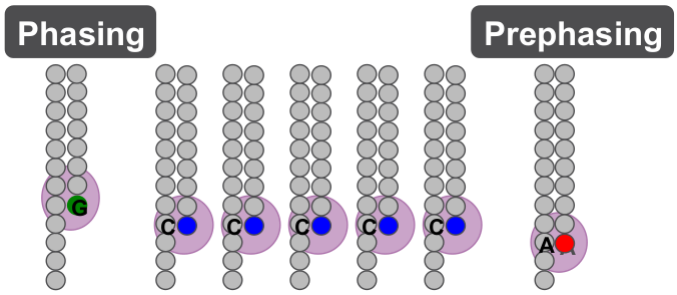

Figura 25: *Phasing* e *pre-phasing*. O *phasing* ocorre quando uma das fitas de um *bead* não incorpora uma base em um dos ciclos de sequenciamento e fica "atrasada" em relação às outras fitas do mesmo *bead*. O *pre-phasing* ocorre quando uma das fitas incorpora muitas bases num mesmo ciclo e fica "adiantada" em relação às outras fitas do mesmo *bead*. Tais fenômenos são comuns a todas as tecnologias de sequenciamento baseadas em amplificação. Fonte: [55].

[23] por isso, essa abordagem é conhecida como *overlap-layout-consensus* (OLC) [53, 54].
[24] sobre o assunto, veja a seção 3.2.3 (principalmente a figura 23).

Como mostrado pelos exemplos a seguir, a obtenção do consenso correto (na presença de erros) ainda é possível via "maioria de votos" e introdução de espaços (−) no alinhamento, mas isso requer programas preparados para lidar com a situação (o que normalmente envolve algoritmos menos eficientes) [43].

Sequência original:	Entrada:	Resposta:
5′ TTACCGTGC 3′	5′ ACCGT 3′	5′ --ACCGT-- 3′
	5′ CGTGC 3′	5′ ----CGTGC 3′
	5′ TTAC 3′	5′ TTAC----- 3′
	5′ TGCCGT 3′	5′ -TGCCGT-- 3′
		5′ TTACCGTGC 3′

Tabela 1: Nesse caso, o erro foi uma substituição de um A por um G (em vermelho) na segunda posição do último fragmento. Fonte: [43].

Sequência original:	Entrada:	Resposta:
5′ TTACCGTGC 3′	5′ ACCGT 3′	5′ --ACC-GT-- 3′
	5′ CAGTGC 3′	5′ ----CAGTGC 3′
	5′ TTAC 3′	5′ TTAC------ 3′
	5′ TACCGT 3′	5′ -TACC-GT-- 3′
		5′ TTACC-GTGC 3′

Tabela 2: Nesse caso, o erro foi uma inserção de um A (em vermelho) na segunda posição do segundo fragmento. Retirando o espaço "−" do consenso, obtemos a sequência correta. Fonte: [43].

Sequência original:	Entrada:	Resposta:
5′ TTACCGTGC 3′	5′ ACCGT 3′	5′ --ACCGT-- 3′
	5′ CGTGC 3′	5′ ----CGTGC 3′
	5′ TTAC 3′	5′ TTAC----- 3′
	5′ TACGT 3′	5′ -TAC-GT-- 3′
		5′ TTACCGTGC 3′

Tabela 3: Nesse caso, o erro foi uma remoção da terceira base (C) do último fragmento, que estava entre as bases A e C (em vermelho). Fonte: [43].

Além dos erros de *base call*, outros fatores que podem atrapalhar a montagem são a presença de fragmentos quiméricos (quimeras) ou contaminação por fragmentos de DNA do vetor ou do

hospedeiro, que precisam ser reconhecidos e removidos antes da montagem[25]. Os fragmentos quiméricos surgem a partir de dois fragmentos corretos de partes distintas da molécula, que se unem para formar um único fragmento. A contaminação ocorre quando a purificação dos fragmentos de DNA de interesse (insertos) não é perfeita e, com isso, parte do vetor (por exemplo, um BAC) também é sequenciada[26][43]. O exemplo a seguir mostra a presença de um fragmento quimérico:

Sequência original:	Entrada:	Resposta:
5′ TTACCGTGC 3′	5′ ACCGT 3′	5′ --ACCGT-- 3′
	5′ CGTGC 3′	5′ ----CGTGC 3′
	5′ TTAC 3′	5′ TTAC----- 3′
	5′ TACCGT 3′	5′ -TACCGT-- 3′
	5′ TTATGC 3′	5′ TTACCGTGC 3′
		5′ TTA---TGC 3′

Tabela 4: Nesse caso, o último fragmento é quimérico, com diferentes regiões da molécula original indicadas por cores diferentes (azul e vermelho). O consenso correto é obtido pois a quimera não foi utilizada na montagem. Na última coluna, um alinhamento entre a quimera e o consenso evidencia as diferentes origens da quimera. Fonte: [43].

5.2.2 Orientação desconhecida

Cada um dos fragmentos pode vir de qualquer uma das fitas da molécula de DNA, e geralmente não sabemos de qual fita cada fragmento veio (apenas sabemos que os *reads* estão na orientação $5' \rightarrow 3'$). Isso cria uma explosão combinatória, pois se temos n fragmentos então existem 2^n (pois cada fragmento pode ser usado na sua versão original ou como o complemento reverso[27]) configurações de orientações, sendo que apenas 2 são corretas (uma configuração para uma das fitas e a outra para a fita complementar). Tentar todas as possibilidades não é o método utilizado por programas de montagem, mas isso permite entender melhor a complexidade adicional devido às orientações[43].

[25] no caso da contaminação, isso é feito comparando as sequências do fragmento com as sequências - já conhecidas - do vetor ou do hospedeiro[43].

[26] sobre BACs, veja a seção 3.1.2 (principalmente a figura 16).

[27] se temos um fragmento de uma das fitas, para obter o fragmento correspondente à fita complementar devemos complementá-lo e depois invertê-lo (para obedecer o padrão de escrevê-lo na orientação $5' \rightarrow 3'$).

Sequência original (ambas as fitas):	Entrada:		Saída:		
5′ CACGTAGTAC 3′	5′ CACGT 3′	→	5′	CACGT-----	3′
3′ GTGCATCATG 5′	5′ ACGT 3′	→	5′	-ACGT-----	3′
	5′ ACTACG 3′	←	5′	--CGTAGT--	3′
	5′ GTACT 3′	←	5′	-----AGTAC	3′
			5′	CACGTAGTAC	3′

Tabela 5: Exemplo de montagem com orientações desconhecidas. Na entrada, as cores indicam a origem dos fragmentos na sequência original. Na saída, a seta para a direita (→) indica que o fragmento foi utilizado na sua versão original, enquanto a seta para a esquerda (←) indica que foi usado o complemento reverso. Fonte: [43].

5.2.3 Repetições

Regiões repetitivas (ou repetições) são sequências que aparecem duas ou mais vezes na molécula alvo. Os tipos de repetição que mais dificultam a montagem são as repetições longas (não totalmente contidas em um único fragmento), sendo que os problemas ocorrem mesmo que as cópias[28] da repetição não sejam idênticas (pois pequenas diferenças entre duas regiões podem ser interpretadas pelo montador como erros de *base call*) [43]. Tais problemas ocorrem pois a existência de repetições pode invalidar a hipótese de que fragmentos com sobreposição entre si vieram da mesma região genômica, como mostra a figura a seguir:

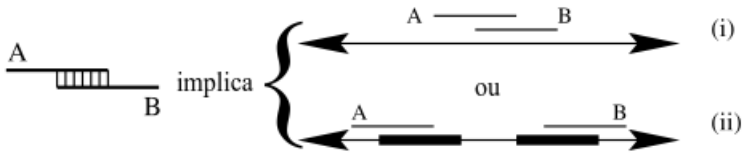

Figura 26: Sobreposição verdadeira (i) e devido a repetições (ii). O objetivo principal é tentar achar (de forma conservadora) as sobreposições verdadeiras e evitar as que são devido a repetições, especialmente no início da montagem. Fonte: modificado de [56].

Se um fragmento estiver totalmente contido em uma repetição, ele pode (no *layout*) fazer parte de qualquer uma das cópias da repetição, o que é especialmente problemático no caso em que as cópias da repetição não são exatamente iguais (pois o consenso será enfraquecido se esse tipo de fragmento for posicionado na cópia errada). Além disso, as repetições podem ser posicionadas de modo a tornar a montagem um processo ambíguo (isto é, dois ou mais *layouts* são compatíveis

[28] nesta seção o termo "cópia" é melhor entendido como uma "versão" (sendo que as versões são semelhantes entre si), e não como uma "reprodução idêntica".

com o conjunto de fragmentos e com o tamanho aproximado da sequência alvo)[29][43]. Exemplos de montagens incorretas geradas pela presença de repetições são apresentados a seguir.

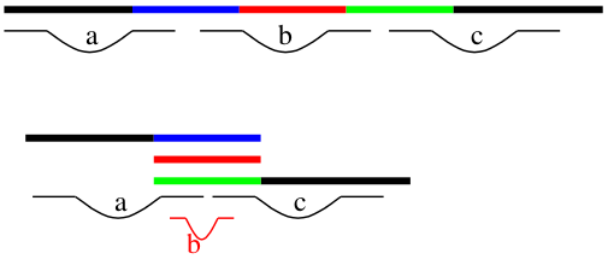

Figura 27: Colapso de repetições seguidas (em tandem). As regiões em azul, vermelho e verde são três cópias de uma mesma repetição. A sequência superior é a sequência alvo, e as inferiores representam um *layout* incorreto de montagem (em que apenas uma cópia da repetição será representada no consenso). Fonte: [57].

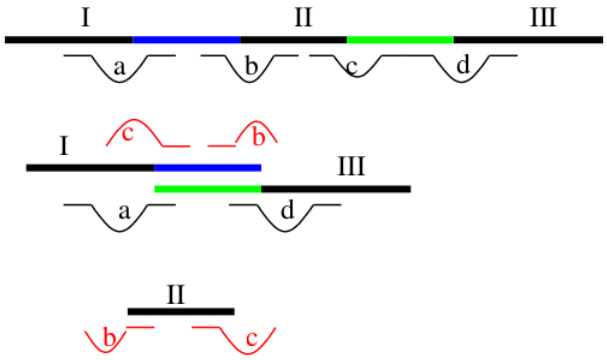

Figura 28: Excisão de regiões flanqueadas por repetições. As regiões em azul e verde são duas cópias de uma mesma repetição. A sequência superior é a sequência alvo, e as inferiores representam um *layout* incorreto de montagem (em que a região II não aparece entre as regiões I e III). Fonte: [57].

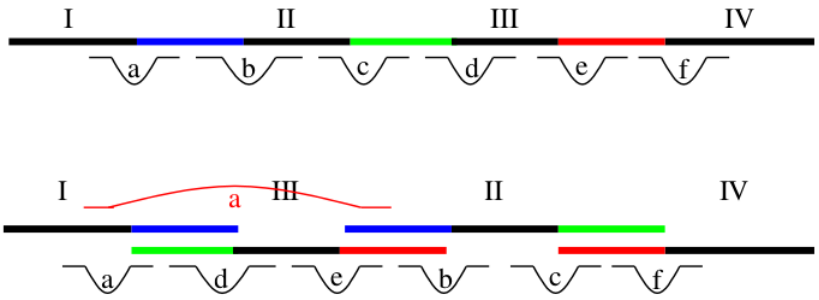

Figura 29: Rearranjo de regiões flanqueadas por repetições. As regiões em azul, vermelho e verde são três cópias de uma mesma repetição. A sequência superior é a sequência alvo, e as inferiores representam um *layout* incorreto de montagem (em que as regiões I, II, III e IV não aparecem na ordem correta). Fonte: [57].

[29] veja a figura 29 para um exemplo desse tipo.

Até aqui, foram discutidos os erros causados por repetições diretas (quando cópias da repetição estão na mesma fita de DNA), mas repetições invertidas (quando cópias da repetição estão em fitas diferentes de DNA) também causam erros. A propensão a erros é ainda maior no segundo caso, pois apenas duas cópias de uma repetição invertida podem gerar ambiguidade na montagem, como mostra a figura a seguir: [43, 58].

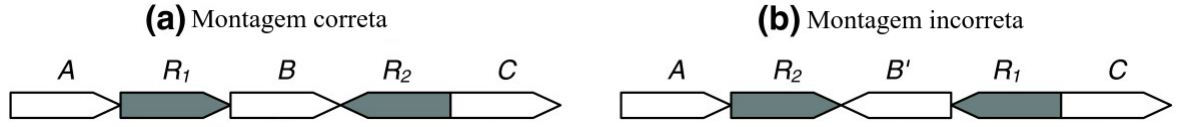

Figura 30: Inversão causada por repetições invertidas. As repetições invertidas R_1 e R_2 podem gerar tanto as montagens em (a) (correta, em que B aparece entre R_1 e R_2) quanto em (b) (incorreta, em que o fragmento B invertido - denominado B′ - aparece entre R_1 e R_2). Fonte: [58].

5.2.4 Falta de cobertura

A cobertura de uma posição do genoma é o número de fragmentos que contêm essa posição. Como não sabemos quais são as posições dos fragmentos na sequência alvo, costuma-se considerar a *cobertura média*, dada por $(N \times T)/G$ (G é o tamanho do genoma, N é o número de *reads* e T é o tamanho médio dos *reads*). Se a cobertura for nula para uma ou mais regiões do genoma (ditas lacunas (*gaps*)), então não é possível formar uma única sequência de consenso para toda a molécula, e sim uma para cada região contígua que foi possível reconstruir a partir dos *reads* (*contig*) [43].

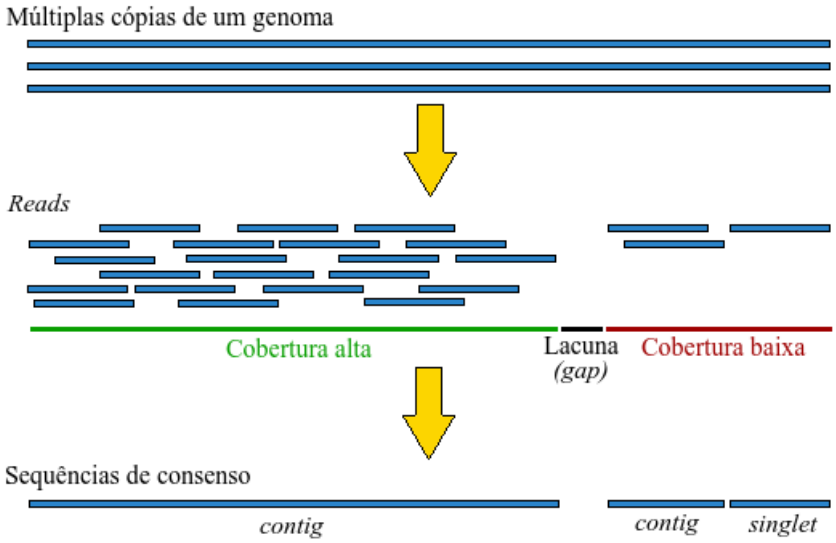

Figura 31: Cobertura do genoma e o processo de montagem. Regiões do genoma que possuem muita, pouca ou nenhuma sobreposição com os *reads* são ditas regiões de cobertura alta, cobertura baixa ou lacunas (*gaps*), respectivamente. Fragmentos que possuam sobreposição com outros formam os *contigs*, enquanto fragmentos sem sobreposição com outros formam os *singlets*. Fonte: modificado de [59].

A falta de cobertura ocorre pois a amostragem dos fragmentos é um processo aleatório. Quanto maior for a cobertura, menores serão as lacunas (*gaps*) obtidas e melhor será a estimativa da sequência alvo a partir do consenso (já que o consenso é obtido via "maioria de votos"). Para tanto, recomenda-se amostrar fragmentos para obter obter uma cobertura mínima de 8x[30] (ou seja, cada posição do genoma aparece 8 vezes no conjunto de fragmentos, em média)[43].

5.3 Modelagem

Com a hipótese de que cada fragmento obtido deve fazer parte (ou seja, ser uma *substring*) da sequência alvo e utilizando a Lei da Parsimônia [31], o problema da montagem de sequências passou a ser modelado pelo problema da *superstring* comum mais curta (*shortest common superstring*, abreviada por SCS)[61], definido formalmente (na sua versão de otimização) a seguir[62–64]:

- instância: um alfabeto finito Σ (= {A,T,C,G}) e um conjunto finito de *strings* $\mathcal{F} \subset \Sigma^*$[32];
- solução viável: uma *string* $w \in \Sigma^*$ tal que cada *string* $x \in \mathcal{F}$ seja uma *substring* de w (i.e., $\forall x \in \mathcal{F}, \exists w_0, w_1 \in \Sigma^* : w = w_0 x w_1$);
- objetivo: minimizar o tamanho de w ($|w|$).

Em outras palavras, a solução do problema é uma sequência w de menor tamanho possível tal que todos os fragmentos (pertencentes ao conjunto $\mathcal{F}$) sejam *substrings* de w. [43, 61, 65].

5.4 Complicações teóricas

A modelagem anterior do problema da montagem possui várias limitações. Ela supõe que não há fragmentos quiméricos, contaminados ou com erros; e que a orientação de cada fragmento é conhecida (o que raramente ocorre na prática). Mesmo que essas suposições fossem verdadeiras, essa modelagem ainda seria problemática na presença de repetições, como mostrado na figura a seguir. Apesar disso, as técnicas usadas para resolver o problema da *superstring* comum mais curta possuem aplicações em outros modelos do problema da montagem, além do problema em questão ter importância teórica[43, 61].

[30] lê-se "oito vezes".

[31] ou Navalha de Occam, princípio segundo o qual a hipótese preferível para qualquer fenômeno é a que possuir o menor número de suposições[60].

[32] Σ^* é o conjunto de todas as *strings* que podem ser formadas usando as letras do alfabeto Σ.

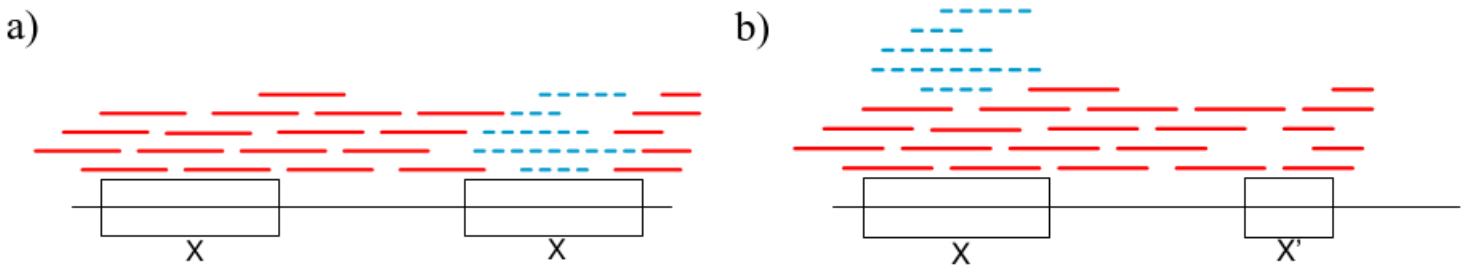

Figura 32: Problemas causados por repetições na modelagem via *superstring* comum mais curta. A sequência alvo (figura a) possui duas cópias de uma repetição longa (X), sendo que os fragmentos tracejados (em azul) estão totalmente contidos na segunda cópia da repetição. Na *superstring* comum mais curta (figura b), os fragmentos totalmente contidos em repetições seriam colapsados para uma única cópia da repetição, fazendo com que as outras cópias ficassem mais curtas (X') ou inexistentes. Fonte: [65].

Além das dificuldades decorrentes das diferenças entre o problema real e o modelo[33], o próprio modelo possui limitações intrínsecas, pois o problema da *superstring* comum mais curta é NP-difícil (ou seja, não existem algoritmos "eficientes"[34] para resolvê-lo, a menos que P = NP) [43, 65–67].

Logo, como não se espera que existam algoritmos exatos (isto é, que encontrem a *superstring* comum mais curta) "eficientes" para o problema, isso motiva o desenvolvimento de algoritmos de aproximação. Sejam A um algoritmo, I uma instância do problema (dada pelo conjunto $\mathcal{F}$ descrito na modelagem, que possui os fragmentos), $A(I)$ a solução devolvida por A (uma *superstring* comum a todos os fragmentos de R), $\langle I \rangle$ o tamanho da instância (dada pela soma dos comprimentos dos fragmentos), $val(A(I))$ o valor da solução devolvida (que corresponde ao comprimento da *superstring* comum encontrada por A) e $opt(I)$ o valor da solução ótima (dada pelo comprimento da *superstring* comum mais curta). Dizemos que A é uma *α-aproximação* para um problema de minimização[35] (como o da *superstring* comum mais curta) se, para toda instância I, o consumo de tempo de A for polinomial em $\langle I \rangle$ e se $val(A(I)) \leq \alpha \; opt(I)$. O fator α é um número que pode depender de I, chamado de *razão de aproximação* [68].

Outra dificuldade do problema é que ele é APX-completo [62–64, 69], o que significa que ele está em APX (o conjunto de problemas de otimização que possui uma razão de aproximação constante[36]) e é APX-difícil (não possui um esquema de aproximação em tempo polinomial, a menos que P = NP). Isso é considerada uma má notícia, já que um esquema de aproximação em tempo polinomial (PTAS, do inglês *polynomial-time approximation scheme*) é o tipo mais útil de algoritmo de aproximação[70].

[33] citando George E. P. Box: "essencialmente, todos os modelos estão errados, mas alguns são úteis".
[34] isto é, com consumo de tempo polinomial no tamanho da entrada, que é a soma dos tamanhos dos fragmentos.
[35] nesse caso, $\alpha \geq 1$ (o algoritmo é exato se $\alpha = 1$).
[36] ou seja, a razão de aproximação independe de I.

Para problemas de minimização, um esquema de aproximação em tempo polinomial (PTAS) é um algoritmo A_ϵ (onde $\epsilon > 0$ é um parâmetro fornecido como entrada) que é uma $(1+\epsilon)$-aproximação para cada $\epsilon > 0$[71, 72]. Em outras palavras, um PTAS devolve uma solução tão próxima quanto se queira da solução ótima (quanto menor for ϵ, melhor a solução), consumindo tempo polinomial em $\langle I \rangle$ (mas não necessariamente polinomial em $1/\epsilon$) [71, 72].

Um algoritmo guloso simples forma a base dos melhores algoritmos de aproximação atuais para o problema: ele repetidamente une duas *strings* com sobreposição máxima até que reste apenas uma. Esse algoritmo é mostrado a seguir [43, 73]:

Algoritmo: O algoritmo guloso

Entrada: um conjunto $\mathcal{F}$ de n *strings* que é livre de *substrings* ▶ i.e., não existem duas *strings* distintas a e b em $\mathcal{F}$ tais que a é *substring* de b

Saída: uma *superstring* do conjunto $\mathcal{F}$

1: **função** GULOSO($\mathcal{F}$)
2: **enquanto** $|\mathcal{F}| > 1$ **faça**
3: escolha $a, b \in \mathcal{F}$ tais que $a \neq b$ e o comprimento de $\langle a, b \rangle$ seja máximo ▶ $\langle a, b \rangle$ denota o maior sufixo de a que também é um prefixo de b
4: $c \leftarrow$ CONCATENA($a, b - \langle a, b \rangle$) ▶ c é a *string* obtida pela concatenação de a com o maior sufixo de b que não faz parte de $\langle a, b \rangle$; note que c é a *superstring* comum mais curta de a e b
5: $\mathcal{F} \leftarrow (\mathcal{F} \cup \{c\}) \setminus \{a, b\}$ ▶ remova a e b de $\mathcal{F}$ e insira c em $\mathcal{F}$
6: **fim enquanto**
7: **devolva** $f \in \mathcal{F}$ ▶ nesse ponto $\mathcal{F}$ possui apenas um elemento, que é uma *superstring* de $\mathcal{F}$ pois a linha 4 garante que sempre obtemos uma *superstring* das *strings* unidas
8: **fim função**

Até agora a melhor razão de aproximação provada para esse algoritmo é de 3,5 [74, 75], mas existe uma conjectura de que o algoritmo é uma 2-aproximação[69, 76]. O caso que motiva essa conjectura (supostamente o pior caso do algoritmo) ocorre para $\mathcal{F} = \{c(ab)^k, (ba)^k, (ab)^k c\}$, para o qual a resposta do algoritmo seria $c(ab)^k c(ba)^k$ (de tamanho $4k+2$), sendo que a *superstring* comum mais curta de $\mathcal{F}$ é $c(ab)^{k+1}c$ (de tamanho $2k+4$)[69, 75]. Observe que $4k+2$ é quase o dobro de $2k+4$ para k suficientemente grande[37], o que leva à conjectura.

Além disso, o melhor algoritmo de aproximação para o problema até o momento (segundo [73–75, 77]) é uma 2,5-aproximação desenvolvida por Z. Sweedyk[78].

[37] formalmente, $\lim_{k \to +\infty} \frac{4k+2}{2k+4} = \lim_{k \to +\infty} \frac{k(4+\frac{2}{k})}{k(2+\frac{4}{k})} = \frac{4}{2} = 2$.

6 Implementação

A parte prática desse trabalho consistiu na implementação de três *pipelines* em Perl: um para o mascaramento das sequências, um para a montagem de regiões gênicas e outro para a validação das montagens obtidas. Nesta seção serão descritos os principais passos de cada um.

6.1 O *pipeline* de mascaramento

O *pipeline* de mascaramento (arquivo `pipeline_mascaramento.pl`) foi construído utilizando o EGene[79], e consiste das seguintes etapas:

1. seleção de todos os arquivos em formato FASTA (identificados pela terminação `.fasta`) do diretório atual (cada arquivo FASTA possui os *reads* de um único BAC);
2. para cada arquivo FASTA selecionado, são executados os seguintes passos dentro do *pipeline* rodado pelo EGene:
   (a) mascaramento das sequências do arquivo usando o programa `cross_match`[80, 81] e o banco de sequências contaminantes UniVec[82];
   (b) mascaramento das sequências obtidas no passo anterior, usando o programa `cross_match` e a sequência do BAC pBeloBAC11[83] como banco de sequências contaminantes[38];
   (c) eliminação das bases contaminantes (identificadas anteriormente) que estejam nos terminais dos *reads* (processo conhecido como *trimming*), usando o componente `trimming.pl` do EGene;
   (d) armazenamento das sequências resultantes do passo anterior num arquivo FASTA com a extensão `.fasta.masked`;

6.2 O *pipeline* de montagem

Após o mascaramento, as sequências obtidas podem ser utilizadas para a montagem. Obrigatoriamente, o *pipeline* (arquivo `pipeline_montagem.pl`) recebe os seguintes parâmetros[39]:

- um arquivo FASTA com os *reads* a serem montados;

[38] pois essa é a sequência do vetor presente no conjunto de dados que foi analisado.
[39] os principais parâmetros opcionais serão mencionados durante a explicação das etapas do *pipeline*.

- um arquivo FASTA com as sequências oriundas de regiões gênicas (chamadas genericamente de *sequências de consulta* ou *queries*, daqui em diante), que podem ser proteínas, ESTs ou DNAcs completos ("*full length*").

Por padrão, a saída (que consiste no conjunto de *contigs* que supostamente contêm as regiões gênicas que originaram as sequências de consulta fornecidas) está em `output_pipeline/output_genseed/final_contigs.fasta`. O objetivo de tentar fazer a montagem a partir das regiões gênicas está em tentar evitar a montagem de repetições (já que a montagem começa a partir de um ponto que sabemos[40] estar presente), e assim evitar os problemas descritos na seção 5.2.3.

O fluxo de execução do *pipeline* está dividido nas seguintes etapas:

6.2.1 Leitura dos parâmetros

Nessa etapa ocorre a obtenção dos parâmetros passados ao *pipeline*, dentro da função `le_parametros()`.

6.2.2 Leitura dos arquivos com os *reads* e com as sequências de consulta

Nessa etapa os arquivos com os *reads* e com as sequências de consulta são lidos pela função `popula_hash()`, responsável por indexar os arquivos FASTA usando *hashes* da linguagem Perl. As chaves dos *hashes* são os identificadores das sequências (presentes nos cabeçalhos dos arquivos FASTA), enquanto os valores são as posições do arquivo em que as sequências começam. Tal abordagem consegue economizar memória (pois não são as próprias sequências que são armazenadas como valores dos *hashes*) sem perder tanta eficiência no acesso, o que permite lidar com uma grande quantidade de *reads*.

6.2.3 Divisão do arquivo com os *reads*

Nessa etapa (efetuada pela função `divide_arquivo_com_os_reads()`) o arquivo com os *read* é dividido igualmente em n outros arquivos, sendo n um parâmetro opcional (o valor padrão é $n = 1$) que indica qual o número de núcleos de processamento (*cores*) que serão utilizados. O objetivo desta etapa é fazer balanceamento de carga (dividir a carga de total de processamento entre os *cores*)[84], o que permite obter melhor desempenho na paralelização do alinhamento (que será feita na etapa seguinte). O arquivo com as sequências de consulta também poderia ser dividido, mas optou-se por dividir o arquivo com os *reads* pois o número de *reads* é tipicamente maior que o número de sequências de consulta.

[40] ou no mínimo esperamos

6.2.4 Alinhamento das sequências de consulta nos *reads*

Nessa etapa (efetuada pela função `roda_e_processa_saida_blat()`) as sequências de consulta são alinhadas em cada um dos conjuntos de *reads* definidos anteriormente, de forma paralela (usando as funções `fork()` e `exec()` da linguagem Perl). O programa usado para fazer os alinhamentos é o BLAT[51], que faz alinhamentos locais[41] de forma heurística[42].

Em seguida, apenas os alinhamentos que obedecem certos critérios são analisados para determinar quais foram as sequências de consulta que melhor se alinharam (quanto maior a pontuação, melhor o alinhamento[43]) com cada um dos *reads*. Tais critérios são os seguintes:

- a identidade do alinhamento deve ser maior ou igual a um mínimo pré-determinado[44];
- pelo menos uma das seguintes condições deve ser satisfeita:
  - a cobertura da sequência de consulta deve ser maior ou igual a um mínimo pré-determinado[44];
  - a cobertura do *read* deve ser maior ou igual a um mínimo pré-determinado[44];
  - se uma das sequências não pôde ser "totalmente" alinhada na outra, então a região do *read* que está na borda do alinhamento deve possuir um sítio de *splice*[45]; essa detecção é feita pela `função possui_splice_site_bordas()`[46].

As estruturas construídas permitem obter as seguintes informações (que serão utilizadas pelas funções posteriores):

- dada um sequência de consulta, é possível descobrir quais os *reads* que se alinharam com ela de modo a satisfazer os critérios anteriores (pelo *hash* `%reads_mapeados_para`);
- quais foram os *reads* que conseguiram se alinhar com alguma sequência de consulta satisfazendo os critérios anteriores (pelo vetor `@nomes_reads_mapeados`);

6.2.5 Seleção das sequências de consulta correspondentes a *reads*

Nessa etapa (efetuada pela função `gera_arquivo_queries_selecionadas()`), as sequências de consulta para as quais foram mapeadas *reads* (que são as chaves do hash `%reads_mapeados_para`)

[41] no caso de eucariontes é necessário considerar alinhamentos locais, pois as sequências de consulta não irão se alinhar de modo contínuo nos *reads* devido à existência de íntrons.

[42] sobre alinhamentos heurísticos, veja a seção 4.4.

[43] sobre pontuação de um alinhamento, veja a seção 4.1.

[44] esse valor mínimo é um parâmetro opcional do *pipeline*, cujo valor padrão é 90% .

[45] identificado através das sequências de consenso dos sítios de *splice*, que são GT no terminal $5'$ do íntron e AG no terminal $3'$ do íntron[85].

[46] implementação baseada no programa `blat2hints.pl`[86].

são escritas num arquivo FASTA (cujo nome termina com `.selecionadas`) para que possam ser examinadas posteriormente (embora esse arquivo não seja utilizado nas etapas seguintes do *pipeline* de montagem). Com isso, as sequências de consulta selecionadas podem ser utilizadas para alinhamento nos *contigs* gerados pelo *pipeline* de montagem, o que permite verificar o quanto os *contigs* conseguiram reconstruir de cada região gênica (isso é feito pelo *pipeline* de validação).

6.2.6 Seleção dos *reads* correspondentes a sequências de consulta

Nessa etapa (efetuada pela função `gera_arquivos_sementes_nao_montadas()`), é gerado um arquivo FASTA para cada conjunto de *reads* (chamado de "*dataset*" no código-fonte) que foram mapeados para uma mesma sequência de consulta. Idealmente, cada um desses conjuntos irá constituir uma "semente" (ainda não montada) da última etapa do *pipeline*.

6.2.7 Seleção dos *reads* não mapeados durante o alinhamento

Nessa etapa (efetuada pela função `gera_arquivo_reads_nao_mapeados()`), é gerado um arquivo FASTA que contém todos os *reads* que não foram mapeados para alguma sequência de consulta. Esse arquivo constitui o "banco de sequências" que será usado na última etapa do *pipeline*.

6.2.8 Montagem inicial das regiões gênicas

Nessa etapa (efetuada pela função `roda_montador()`), cada um dos conjuntos de *reads* mapeados para a mesma sequência de consulta (gerados em 6.2.6) é montado separadamente (de forma paralela, assim como os alinhamentos descrito em 6.2.4) utilizando um dos seguintes montadores: Phrap[80], MIRA[87] ou Newbler[88][47]. Cada um dos *contigs* que forem montados será efetivamente uma "semente" da última etapa do *pipeline*.

6.2.9 Extensão final das regiões gênicas

Nessa etapa (efetuada pela função `roda_genseed()`), é utilizado o programa GenSeed[89] com as "sementes" obtidas em 6.2.8 e o banco obtido em 6.2.7.

O GenSeed é baseado na seleção iterativa e montagem de sequências que tenham sobreposição com uma sequência inicial, chamada de "sequência semente". O software faz uma busca de similaridade da sequência semente num banco de *reads* (não montados) e seleciona quais podem

[47] o padrão é usar o Newbler, que é feito com o propósito de montar *reads* oriundos do sequenciador Roche/454.

estender os terminais da semente. Os *reads* selecionados são então montados junto com a semente, resultando numa sequência de consenso maior. O processo então recomeça usando esse consenso como nova semente e os *reads* ainda não utilizados para a montagem como novo banco, até que não seja possível continuar a extensão da(s) semente(s)[48][89].

6.3 O *pipeline* de validação

Após a montagem, os *contigs* gerados pelo *pipeline* de montagem podem ser comparados com as sequências corretas[49] (os "*contigs* confiáveis", para verificar se a montagem não gerou quimeras) e também com algum outro conjunto de "*contigs* de comparação" (gerados por algum outro método de montagem, para verificar se os *contigs* do *pipeline* possuem algum diferencial em relação ao outro método). Obrigatoriamente, o *pipeline* (arquivo `pipeline_validacao.pl`) recebe os seguintes parâmetros:

- um arquivo FASTA com os "*contigs* de interesse"[50] (no caso, os *contigs* gerados pelo *pipeline* de montagem);
- um arquivo FASTA com as sequências de consulta (*queries*) usadas para montar os *contigs* que desejam ser analisados[51];
- um arquivo FASTA com os "*contigs* confiáveis" (sequências montadas de forma supostamente correta, que serão usadas para verificar a qualidade das montagens);
- um arquivo FASTA com os "*contigs* de comparação" (montados por algum outro método);

A saída principal do *pipeline* é uma tabela em formato TSV (*Tab-separated values*)[90], que pode ser visualizada em qualquer editor de planilhas[52]. Também é gerado um histograma (em formato PNG[91]) dos tamanhos dos *contigs* que desejam ser analisados.

Cada linha da tabela é referente a um par de *contigs* (um *contig* de interesse e outro de comparação) que melhor se alinharam um no outro. As colunas (na mesma ordem em que aparecem na tabela) estão descritas a seguir[53]:

- colunas que indicam qual o par de *contigs* analisado em cada linha da tabela:

[48] ou até que seja ultrapassado um número pré-determinado de iterações.
[49] ou, melhor dizendo, com sequências que se pensam estar corretas, caso estejam disponíveis.
[50] que são os *contigs* que se deseja analisar.
[51] esse é o arquivo gerado em 6.2.5
[52] basta indicar ao editor de planilhas que o separador entre os campos é uma tabulação (`\t`).
[53] assim como no *pipeline* de montagem, todos os alinhamentos são feitos usando o BLAT[51].

contigName nome (identificador do arquivo FASTA) do *contig* de interesse.

compName nome (identificador do arquivo FASTA) do *contig* de comparação.

- colunas referentes ao alinhamento das sequências de consulta nos *contigs* de interesse:

  pipe_queryName nome (identificador do arquivo FASTA) da sequência de consulta.

  pipe_queryStart posição do início do alinhamento na sequência de consulta.

  pipe_queryEnd posição do fim do alinhamento na sequência de consulta.

  pipe_querySize tamanho da sequência de consulta.

  pipe_contigStart posição do início do alinhamento no contig de interesse.

  pipe_contigEnd posição do fim do alinhamento no contig de interesse.

  pipe_contigSize tamanho do *contig* de interesse.

  pipe_5_size tamanho da região do *contig* de interesse que está a $5'$ do início da sequência de consulta [54].

  pipe_3_size tamanho da região do *contig* de interesse que está a $3'$ do fim da sequência de consulta[55].

  pipe_cobertura_query cobertura da sequência de consulta no alinhamento[56].

  pipe_id_query identidade do alinhamento[56].

- colunas referentes ao alinhamento dos *contigs* de interesse nos *contigs* confiáveis:

  pipe_cobertura_contig_no_correto cobertura do *contig* de interesse no alinhamento.

  pipe_id_contig_no_correto identidade do alinhamento.

- colunas referentes ao alinhamento das sequências de consulta nos *contigs* de comparação:

  comp_queryName nome (identificador do arquivo FASTA) da sequência de consulta.

  comp_queryStart posição do início do alinhamento na sequência de consulta.

  comp_queryEnd posição do fim do alinhamento na sequência de consulta.

  comp_querySize tamanho da sequência de consulta.

  comp_contigStart posição do início do alinhamento no contig de comparação.

  comp_contigEnd posição do fim do alinhamento no contig de comparação.

[54] valor igual a -1 indica que o início da sequência de consulta não foi mapeado no *contig*.
[55] valor igual a -1 indica que o fim da sequência de consulta não foi mapeado no *contig*.
[56] número no intervalo [0;1].

comp_contigSize tamanho do *contig* de comparação.

comp_5_size tamanho da região do *contig* de comparação que está a $5'$ do início da sequência de consulta [54].

comp_3_size tamanho da região do *contig* de comparação que está a $3'$ do fim da sequência de consulta[55].

comp_cobertura_query cobertura da sequência de consulta no alinhamento[56].

comp_id_query identidade do alinhamento[56].

- colunas referentes ao alinhamento dos *contigs* de comparação nos *contigs* confiáveis:

  comp_cobertura_comp_no_correto cobertura do *contig* de comparação no alinhamento.

  comp_id_comp_no_correto identidade do alinhamento.

- colunas referentes ao alinhamento dos *contigs* de interesse nos *contigs* de comparação:

  pipecomp_cobertura_contig_no_comp cobertura do *contig* de interesse no alinhamento.

  pipecomp_id_contig_no_comp identidade do alinhamento.

7 Resultados

Os três *pipelines* descritos anteriormente foram usados em *reads* de 6 BACs[57] do cultivar R570 de cana-de-açúcar (híbrido entre *S. officinarum* e *S. spontaneum*). As sequências completas dos insertos dos BACs (supostamente corretas) já estavam disponíveis, o que permitiu avaliar a qualidade dos *contig* montados pelo *pipeline* de montagem.

Foram usadas como sequências de consulta as proteínas de sorgo (*S. bicolor*) disponíveis em [92], pois sorgo é a planta de cultivo mais próxima evolutivamente da cana-de-açúcar (estima-se que a divergência evolutiva entre ambas tenha ocorrido há 5 milhões de anos)[93].

Como *contigs* de comparação, foram usados os *contigs* resultantes da montagem dos *reads* (após a fase de mascaramento) utilizando somente o montador Newbler.

Os resultados (tabelas em formato TSV) estão no arquivo `resultados.tar.gz`, sendo que uma das tabelas (referente ao BAC SHCRBa_218_D04) está parcialmente reproduzida a seguir [58]:

[57] por abuso de linguagem, daqui em diante será usado o termo "montagem de BACs", sendo que o mais apropriado seria "montagem dos insertos dos BACs" (veja seção 3.1.2).

[58] de forma simplificada, apenas mostrando o essencial para avaliar as montagens resultantes do *pipeline*

nome do contig	nome da proteína	tamanho da região $5'$ (pb)	tamanho da região $3'$ (pb)	identidade na proteína	cobertura no BAC	identidade no BAC
C_1	P_1	708	X	0,99	0,95	1
C_2	P_2	515	X	0,90	1	0,99
C_3	P_2	X	X	0,97	0,99	1
C_4	P_2	X	266	0,92	0,98	0,99
C_5	P_3	20	X	0,96	1	1
C_6	P_4	282	X	0,93	0,94	1
C_7	P_5	740	X	0,96	0,97	0,99
C_8	P_5	X	855	0,96	0,93	0,99
C_9	P_6	940	X	0,94	0,93	0,99
C_{10}	P_6	X	605	0,96	1	0,99
C_{11}	P_7	618	450	0,96	0,91	0,99

Tabela 6: Resultados das montagens para o BAC SHCRBa_218_D04. O símbolo X está no lugar do valor "-1" descrito na seção 6.3.

Observa-se que foi possível estender as regiões $5'$ das proteínas P_1 a P_7 e as regiões $3'$ das proteínas P_2, P_5, P_6 e P_7. Com isso, é possível que tais extensões contenham os elementos cis-regulatórios dos genes de cana-de-açúcar que sejam homólogos aos genes de sorgo em questão, principalmente o promotor (presente na região $5'$ da fita codificante, próximo ao gene que ele regula [59])[94].

Além disso, todos os *contigs* puderam ser mapeados com alta cobertura e identidade no BAC, o que indica que as montagens são confiáveis.

8 Conclusão

Para os 6 BACs que puderam ser montados e validados:

- aproximadamente 70% das regiões gênicas puderam ser estendidas em algum sentido (a $5'$ do início de tradução ou a $3'$ do fim da tradução);
- de modo geral, não houve ocorrência de quimeras (os *contigs* foram mapeados com aproximadamente 96% de cobertura e 99% de identidade na sequência supostamente correta do BAC).

[59] veja a figura 8.

Logo, o *pipeline* poderia ser utilizado como uma forma razoavelmente confiável (embora limitada[60]) de montar regiões gênicas, com alguma chance de conseguir estender a montagem até a região promotora dos genes selecionados pelo *pipeline* de montagem.

[60] tais limitações decorrem principalmente da presença de íntrons grandes, o que dificultou a união dos *contigs* que possuíam os éxons de cada proteína.

Glossário

açúcar pequeno carboidrato com uma unidade monomérica de fórmula geral $(CH_2O)_n$[11]. 12, 15, 50, 51

adaptador molécula de DNA de fita dupla curta e sintetizada quimicamente (cuja sequência é conhecida), utilizada para ligar os terminais de duas outras moléculas[95]. 25, 28, 50, 52

alelo uma das várias formas alternativas de um gene. Em uma célula diploide, cada gene terá dois alelos, cada um ocupando a mesma posição (locus) em cromossomos homólogos[11]. 50, 54

alinhamento em Bioinformática, um alinhamento de sequências é uma forma de organizar sequências de DNA, RNA ou proteína para identificar regiões similares que possam ser consequência de relações funcionais, estruturais ou evolutivas entre elas[96]. 4, 11, 28–33, 45–47, 50, 56

aminoácido molécula orgânica que contém tanto um grupo amino quanto um grupo carboxila; monômero utilizado na construção de proteínas[11]. 17–21, 28, 30, 31, 50, 52, 54, 56–60

amplificação é a criação de múltiplas cópias de uma molécula de DNA[97]. 33, 50, 57

anticódon sequência de três nucleotídeos em uma molécula de RNAt que é complementar ao códon de três nucleotídeos em uma molécula de RNAm[11]. 20, 50

antiparalelo descreve a orientação relativa das duas fitas em uma dupla-hélice de DNA ou em duas regiões pareadas de uma cadeia polipeptídica; a polaridade de uma fita é orientada na direção oposta da polaridade da outra[11]. 50

BAC cromossomo artificial de bactéria (***b**acterial **a**rtificial **c**hromosome*); vetor de clonagem que pode acomodar grandes fragmentos de DNA (de até 1 milhão de pares de bases)[11]. 22–24, 35, 42, 48–50, 60

base uma substância que pode reduzir o número de prótons (H^+) em solução, tanto por aceitar diretamente íons H^+ quanto por liberar íons H^-, os quais se combinam a H^+ e formam H_2O. As purinas (A,G) e pirimidinas (T,C,U) do DNA e do RNA são bases orgânicas nitrogenadas, e com frequência são referidas apenas como bases[11]. 12, 15, 16, 32, 33, 42, 50, 52, 56, 60

base calling conversão de dados "brutos" de um sequenciador (tipicamente imagens que captam sinais de fluorescência) nas sequências propriamente ditas (*reads*) e pontuações de qualidade (uma estimativa do grau de confiança do sequenciamento) associadas a cada base[55]. 33, 50, 51

BCC Bacharelado em Ciência da Computação. 11, 50, 73

biblioteca de DNA coleção de moléculas de DNA clonadas, representando o genoma inteiro (biblioteca genômica) ou cópias de DNA complementar (DNAc) a partir do RNAm produzido por uma célula (biblioteca de DNAc)[11]. 50, 60

cap $5'$ nucleotídeo alterado adicionado ao terminal $5'$ do pré-RNAm em eucariontes para aumentar a estabilidade do RNA durante a tradução[98]. 18, 50, 60

carboidrato termo geral para designar açúcares e compostos relacionados contendo carbono, hidrogênio e oxigênio, geralmente com a fórmula empírica $(CH_2O)_n$[11]. 50

catalisador é toda e qualquer substância que acelera uma reação química sem ser consumida durante o processo[99]. 50, 51

catálise é a mudança de velocidade de uma reação química devido à adição de uma substância (catalisador) que praticamente não se transforma ao final da reação[99]. 18, 50, 53

cauda poli-A longa sequência de nucleotídeos "A" que é adicionada à extremidade $3'$ da molécula de RNAm nascente em eucariontes, importante para a tradução e para a estabilidade do RNAm[11, 100]. 18, 50, 60

CDS sequência codificante (***c****oding* DNA ***s****equence*); porção do DNA ou do RNA de um gene, composta de éxons, que codifica uma proteína[101]. 19, 50

célula unidade estrutural e funcional básica de todos os organismos vivos conhecidos[102]. 10, 13, 15, 17, 18, 50–52, 54, 56, 58, 60

centrômero é a região mais condensada do cromossomo (normalmente no meio deste), que mantém as cromátides-irmãs unidas[103]. 11, 50

citoplasma é o espaço intracelular entre a membrana plasmática e o envoltório nuclear em seres eucariontes, enquanto nos procariontes corresponde à totalidade da área intracelular[104]. 50, 54

clone população de indivíduos idênticos (células ou organismos) formada por divisões repetidas (assexuadas) a partir de um ancestral comum. Também utilizado como verbo: "clonar um gene", significando produzir muitas cópias de um gene por meio do crescimento de um clone de células carreadoras (como *E. coli*), nas quais um gene foi introduzido e das quais ele pode ser recuperado, por técnicas de DNA recombinante[11]. 50, 52, 53, 60

cobertura é o número médio de vezes que uma posição qualquer do genoma foi sequenciada. Pode ser calculada como $(N \times T)/G$, sendo G o tamanho do genoma, N o número de *reads* e T o tamanho médio dos *reads*. Uma cobertura alta no sequenciamento *shotgun* é desejável, pois ela diminui erros na montagem e no *base calling*[105]. 23, 38, 50

códon sequência de três nucleotídeos em uma molécula de DNA ou RNAm que representa a instrução para a incorporação de um aminoácido específico em uma cadeia polipeptídica crescente[11]. 19, 20, 50, 60

complementar duas sequências de ácidos nucleicos são complementares se podem formar uma dupla-hélice com as bases perfeitamente pareadas[11]. 12–15, 26, 50, 53, 55–57, 60

consenso o mesmo que sequência de consenso[43]. 33, 46, 50

contaminação uma sequência contaminada é uma que não representa fielmente a informação genética da origem biológica de interesse, pois contém um ou mais segmentos de outras origens (como vetores, adaptadores ou iniciadores) [106]. 34, 50

contig um *contig* (da palavra **contíg**uo) é um conjunto de fragmentos de DNA sobreponíveis que representa uma sequência de consenso do DNA. Na montagem de sequências, refere-se a um conjunto de *reads* sobreponíveis que supostamente representa uma região contígua do DNA (quanto mais extensas forem as sobreposições, maior será a confiabilidade dessa suposição). No sequenciamento *shotgun* hierárquico, refere-se a um conjunto de clones sobreponíveis que forma um mapa físico do genoma, usado para guiar o sequenciamento e a montagem[107]. 4, 23, 29, 32, 38, 43, 45–50, 58

cromátide é cada um dos dois filamentos de DNA formados pela duplicação de um cromossomo. [108]. 50, 52

cromátides-irmãs são cromátides originadas a partir do mesmo cromossomo[11]. 50, 51

cromossomo estrutura composta por uma molécula de DNA muito longa e proteínas associadas, contendo toda ou parte da informação genética de um organismo[11]. 4, 10, 23, 32, 50–52, 55, 56, 59, 60

cromossomos homólogos cópia maternal e paternal de um cromossomo específico em uma célula diploide[11]. 50, 53

cultivar variedade cultivada (***culti****vated* ***var****iety*); é a designação dada a determinada forma de uma planta cultivada, correspondendo a um determinado genótipo e fenótipo que foi selecionado e recebeu um nome único e devidamente registado com base nas suas características produtivas, decorativas ou outras que o tornem interessante para cultivo. O cultivar deve apresentar em cultura, e manter durante o processo de propagação, um conjunto único de características que o distingam de maneira consistente de plantas semelhantes da mesma espécie[109]. 11, 48, 50

desnaturação em relação a ácidos nucleicos, significa a separação de uma fita dupla (de DNA ou RNA) em duas fitas simples, que ocorre quando as ligações de hidrogênio entre as fitas são quebradas (devido a temperaturas elevadas, por exemplo)[110]. 25, 50

diploide que contém um genoma duplo (dois conjuntos de cromossomos homólogos e, portanto, duas cópias de cada gene)[11]. 50, 52

DNA ácido desoxirribonucleico (**d**eoxyribo**n**ucleic **a**cid); polímero de nucleotídeos que contém a informação genética usada no desenvolvimento e funcionamento de todos os seres vivos[111]. 4, 10–18, 22, 25–27, 32, 50–60

DNA recombinante qualquer molécula de DNA formada pela ligação de segmentos de DNA de origens diferentes[11]. 50, 51

DNAc DNA complementar; molécula de DNA sintetizada como uma cópia de uma molécula de RNAm e, portanto, sem os íntrons que estão presentes no DNA genômico[11]. 43, 50, 51, 53

domínio proteico porção de uma proteína com uma estrutura terciária particular. As proteínas grandes são em geral compostas por vários domínios, cada um conectado ao próximo através de regiões flexíveis curtas da cadeia polipeptídica. Domínios homólogos são reconhecidos em várias proteínas diferentes[11]. 30, 50

duplicação processo pelo qual uma cópia de uma molécula de DNA é feita[11]. 13, 14, 16, 50, 52, 57, 59

elemento cis-regulatório região do DNA ou RNA que regula a expressão de genes localizados na mesma molécula de DNA[4]. 4, 11, 48, 50

emulsão é a mistura entre dois líquidos imiscíveis em que um deles (a fase dispersa) encontra-se na forma de finos glóbulos no seio do outro líquido (a fase contínua), formando uma mistura estável. Se o líquido "A" é a fase dispersa e o líquido "B" é a fase contínua, temos uma "emulsão A em B". As emulsões mais conhecidas consistem de água e óleo[112]. 25, 50

enzima proteína especializada na catálise de reações biológicas[113]. 14, 16, 18, 26, 27, 50, 59, 60

EP exercício-programa. 50

EST um marcador de sequência expressa (EST, do inglês ***e**xpressed **s**equence **t**ag*) é uma *substring* de uma sequência de DNAc. Podem ser usados para identificar transcritos de genes e para determinar sequências de genes. Um EST resulta do sequenciamento de uma porção de um DNAc clonado (por exemplo, sequenciando centenas de pares de bases de uma extremidade de um clone de DNAc tomado de uma biblioteca de DNAc). Como esses clones consistem

de DNA complementar ao RNAm, os ESTs representam porções de genes expressos[114]. 43, 50

estrutura terciária forma complexa tridimensional de uma cadeia polimérica enovelada, especialmente uma proteína ou molécula de RNA[11]. 50, 53

eucarionte organismo cujas células possuem um núcleo delimitado por um sistema de membranas (a membrana nuclear ou carioteca), nitidamente separado do citoplasma[115]. 10, 17, 44, 49–51, 59

éxon região expressa (***expressed region***); segmento de um gene eucariótico que será representado na molécula madura de RNA, geralmente adjacente a íntrons. Em genes que codificam proteínas, os éxons codificam os aminoácidos[11]. 17, 18, 49–51, 59

expressão gênica produção, por um gene, de um produto molecular observável (RNA ou proteína)[11]. 50

FASTA uma sequência em formato FASTA começa com uma descrição de uma única linha, seguida por linhas de dados em sequência. A linha de descrição se distingue a partir da sequência dos dados por um símbolo maior-que (“>”) na primeira coluna. A palavra que segue o símbolo “>” é o identificador da sequência, e o resto da linha é a descrição (ambos são opcionais). Não deve haver nenhum espaço entre o “>” e a primeira letra do identificador. Recomenda-se que todas as linhas do texto sejam mais curtas do que 80 caracteres. A sequência termina se uma outra linha de partida com um “>” aparece, o que indica o início de outra sequência. Um exemplo simples de uma sequência em formato FASTA:
`>seq1`
`KYRTWEEFTRAAEKLYQADPMKVRVVLKYRHCDGNLCIKVTDDVVCLLYRTDQAQDVKKIEKFHSQLMRLME`
`LKVTDNKECLKFKTDQAQEAKKMEKLNNIFFTLM` [116]. 42, 43, 45–47, 50, 75

fenótipo caráter observável em um célula ou organismo (incluindo aparência física e comportamento)[11]. 50, 52

gene sequência de nucleotídeos do DNA que pode ser transcrita em uma versão de RNA; segmento de DNA que carrega informação genética[117]. 4, 11, 15, 17, 29, 30, 48–51, 53–55

genoma informação genética total que pertence a uma célula ou a um organismo; em particular, a informação mantida no DNA[11]. 10, 11, 13, 22–24, 29, 32, 38, 50–57

genômica estudo das sequências de DNA e das propriedades dos genomas totais[11]. 10, 50

genótipo constituição genética de uma célula individual ou de um organismo. Combinação particular de alelos observada em um indivíduo específico[11]. 50, 52

grupo amino -NH_2; grupo funcional fracamente básico derivado da amônia (NH_3) no qual um ou mais átomos de hidrogênio são substituídos por outro átomo. Em soluções aquosas, ele pode receber um próton (H^+) e carregar uma carga positiva (-NH_3^+)[11]. 18, 20, 21, 50, 56

grupo carboxila -COOH; átomo de carbono ligado a um átomo de oxigênio por ligação dupla (-C=O) e a um grupo hidroxila (-C-OH). Moléculas contendo um grupo carboxila são ácidos fracos (ácidos carboxílicos)[11]. 18, 20, 21, 50, 56

hidrólise clivagem de uma ligação covalente com concomitante adição de água; fórmula geral AB + H_2O $\rightarrow$ AOH + BH[11]. 50

histona membro de um grupo abundante de pequenas proteínas, que formam a região central dos nucleossomos, ao redor dos quais o DNA se enrola nos cromossomos eucarióticos[11]. 10, 50

homologia relação entre genes, proteínas ou estruturas que possuem uma origem evolutiva comum[11]. 50

homólogo um de dois ou mais genes que possuem um mesmo gene ancestral[11]. 48, 50, 53

IB Instituto de Biociências. 2, 11, 50, 73

IC iniciação científica. 2, 50, 75

IME Instituto de Matemática e Estatística. 2, 50, 75

iniciador oligonucleotídeo que forma pares com uma fita molde de DNA ou RNA e promove a síntese de uma nova fita complementar por uma polimerase[11]. 15, 16, 25, 50, 52, 57, 58

inserto fragmento de DNA que é inserido em outro (o vetor) para que possa ser duplicado. No caso de projetos de sequenciamento, o inserto é a parte que queremos sequenciar (ou seja, a parte desconhecida). Normalmente, a sequência de DNA completa do vetor é conhecida[118]. 23, 35, 48, 50

íntron região intragênica (***intr**agenic regi**on***); região não codificante de um gene eucariótico que é transcrita na molécula de RNA, mas que é removida por *splicing* do RNA[11]. 17, 18, 44, 49, 50, 53, 54, 59

íon um átomo que tenha ganhado ou perdido elétrons, adquirindo carga; por exemplo, Na^+ e Cl^-[11]. 14, 50

IQ Instituto de Química. 2, 11, 50, 73

lacuna (*gap*) no contexto de sequenciamento, refere-se a uma região do genoma não capturada (coberta) por nenhum *read*[57]. No contexto de alinhamento de sequências, refere-se ao uso de caracteres "–" para indicar uma inserção ou deleção (*indel*) de um monômero de uma das sequências em relação à outra[46]. 28, 38, 39, 50

ligação covalente ligação química estável entre dois átomos, produzida pelo compartilhamento de um ou mais pares de elétrons[11]. 50, 55

ligação de hidrogênio ligação não covalente na qual um átomo de hidrogênio eletropositivo é parcialmente compartilhado por dois átomos eletronegativos[11]. 12, 13, 50, 52, 57

ligação fosfodiéster ligação química covalente formada quando dois grupos hidroxil formam ligações éster com o mesmo grupo fosfato, como entre nucleotídeos adjacentes no RNA e no DNA[11]. 12, 50

ligação não covalente ligação química na qual os elétrons não são compartilhados. Ligações não covalentes são relativamente fracas, mas podem ser somadas, gerando interações fortes e altamente específicas entre moléculas[11]. 50

ligação peptídica ligação química entre o grupo carboxila de um aminoácido e o grupo amino de um segundo aminoácido. As ligações peptídicas unem aminoácidos em proteínas[11]. 20, 50, 58

ligação química afinidade química entre dois átomos que os mantêm unidos[11]. 50, 55, 56

mapa físico mapa genético que posiciona fragmentos de DNA em cromossomos, mostrando a distância entre eles em pares de bases[119, 120]. 23, 50, 52

mascaramento é o processo de comparar um conjunto de *reads* de interesse com um banco de sequências indesejadas (contaminantes ou repetitivas) de forma a identificar quais sequências do banco estão presentes nos *reads*. As bases dos *reads* que correspondam a sequências indesejadas são normalmente substituídas por "Xs" ou "Ns"[121]. 42, 50, 60, 75

mate pair par de sequências curtas obtidas de ambos os terminais de um fragmento de DNA de interesse. Teoricamente, devem conter informação suficiente para mapear a sequência de forma única no genoma (e assim representar o fragmento de DNA completo)[31, 122]. 23, 50

membrana plasmática membrana biológica que separa o interior de todas as células do ambiente externo[123]. 50, 51

molde uma fita simples de DNA ou RNA, cuja sequência de nucleotídeos atua como um guia para a síntese de uma fita complementar[11]. 13–16, 50, 55, 59

monômero pequena molécula capaz de se ligar a outros monômeros, formando moléculas maiores denominadas polímero[124]. 12, 18, 50, 57

montagem alinhamento e fusão de fragmentos de DNA vindos de uma molécula maior, feito para poder reconstruir a sequência da molécula original[1]. 4, 11, 23, 29, 32, 33, 35, 42, 43, 45, 46, 48–52, 60

núcleo organela delimitada por membrana em uma célula eucariótica, contendo o DNA organizado em cromossomos[11]. 10, 50, 54, 59

nucleotídeo molécula que é a unidade estrutural do DNA e do RNA. É identificado por sua base nitrogenada, que pode ser adenina (A), timina (T), citosina (C), guanina (G) ou uracila (U)[125]. 10, 12–14, 16, 26–28, 30, 31, 50–60

otimização em matemática, refere-se ao estudo de problemas em que se busca minimizar ou maximizar uma função através da escolha sistemática dos valores de variáveis reais ou inteiras dentro de um conjunto viável[126]. 39, 50

P = NP o problema "P versus NP" é o principal problema aberto da ciência da computação. Informalmente, ele pergunta se todos os problemas cujas soluções podem ser verificadas "eficientemente" (i.e., em tempo polinomial) também podem ser resolvidas "eficientemente". A classe de problemas que podem ser resolvidos em tempo polinomial é a classe P, enquanto a classe de problemas para os quais a resposta pode ser verificada em tempo polinomial é a classe NP. Além de ser um problema importante em teoria da computação, sua solução teria implicações profundas para áreas como matemática, criptografia, pesquisa de algoritmos, inteligência artificial, teoria dos jogos, processamento multimídia e várias outras[127]. 40, 50

par de bases dois nucleotídeos em uma molécula de RNA ou DNA que estão emparelhados por ligações de hidrogênio (por exemplo, G com C e A com T ou U)[11]. 4, 13, 32, 50, 53, 56, 60

pb pares de bases. 27, 50

PCR reação em cadeia da polimerase (***p**olymerase **c**hain **r**eaction*); técnica para a amplificação de regiões específicas de DNA, utilizando oligonucleotídeos (iniciadores) específicos e múltiplos ciclos de síntese de DNA, com cada ciclo sendo seguido por um breve tratamento por calor para separar as fitas complementares[11]. 25, 50

pipeline em engenharia de software, é uma cadeia de elementos de processamento organizados de tal forma que a saída de cada elemento é a entrada do próximo[128]. 4, 11, 31, 42–46, 48–50, 75, 76

plasmídeo pequena molécula circular de DNA extracromossômico (ocorre geralmente em bactérias), com duplicação independente do genoma. Os plasmídeos modificados são amplamente utilizados como vetores para clonagem de DNA[11]. 50, 60

Poli Escola Politécnica. 2, 50, 73

polimerização união de moléculas de um dado composto (monômero) para formar um novo composto, designado por polímero[129]. 14, 50

polímero macromolécula formada pela repetição de pequenas e simples unidades químicas (monômeros), ligadas covalentemente[130]. 15, 50, 53, 56–58

polinomial Um algoritmo possui complexidade de tempo (ou espaço) polinomial se existe um polinômio p tal que para toda instância I do problema o seu consumo de tempo (ou espaço) é limitado superiormente por $p(\langle I \rangle)$ (onde $\langle I \rangle$ é o tamanho da instância). O conceito de algoritmo polinomial deve ser entendido como uma formalização da ideia de algoritmo eficiente. Se um problema é NP-difícil então é improvável que exista um algoritmo de consumo de tempo polinomial exato para o problema[68]. 40, 41, 50

polipeptídeo polímero linear composto por aminoácidos. As proteínas são grandes polipeptídeos, e os dois termos podem ser usados como sinônimos[11]. 18, 50

pré-RNAm molécula precursora do RNAm[11]. 17, 18, 50, 51, 59

prefixo um prefixo de uma *string* $T = t_1 \dots t_n$ é uma *string* $\widehat{T} = t_1 \dots t_m$, onde $m \leq n$. Em outras palavras, é uma *substring* de T que começa no primeiro caractere (t_1)[131]. 33, 41, 50

primer o mesmo que iniciador[132]. 15, 50, 60

procarionte micro-organismo unicelular cujas células não apresentam seu material genético delimitado por uma membrana[133]. 50, 51

promotor sequência de nucleotídeos no DNA à qual a RNA-polimerase se liga para iniciar a transcrição[11]. 17, 49, 50

proteína moléculas orgânicas mais abundantes e importantes nas células; polímero linear de aminoácidos ligados por ligações peptídicas em uma sequência específica[11, 134]. 17–21, 29, 30, 43, 48–58, 60

PTAS esquema de aproximação em tempo polinomial. 41, 50

quimera *contig* que não representa uma região contígua do DNA que originou os *reads* que o compõe. Também é usado para denotar um *read* resultante da união molecular de dois fragmentos de DNA vindos de diferentes partes da molécula[43]. 4, 11, 34, 46, 49, 50

read sequência de caracteres sobre o alfabeto {A,T,C,G}, que representa um fragmento de DNA. Por convenção, uma sequência nucleotídica é escrita sempre da extremidade $5'$ para a $3'$, e deve ser lida da esquerda para a direita e nas linhas sucessivas em direção ao fim (na extremidade inferior direita, como ocorre nos textos ocidentais)[1, 11]. 4, 11, 22–24, 27, 31, 32, 35, 38, 42–46, 48, 50–52, 55, 56, 58–60

repetição subsequência de nucleotídeos que aparece duas ou mais vezes na molécula de DNA a ser sequenciada[43]. 4, 11, 39, 43, 50

ribossomo partícula composta de RNArs e proteínas ribossomais que catalisa a síntese de proteína usando informações fornecidas pelo RNAm[11]. 20, 50, 58, 60

RNA ácido ribonucleico (***ribo****nucleic* ***a****cid*); polímero de nucleotídeos que desempenha vários papéis na célula, como síntese de proteínas e regulação gênica[135]. 15–18, 20, 50–60

RNAm RNA mensageiro; molécula de RNA que é traduzida em proteína pelos ribossomos[11]. 17–21, 50–53, 57–60

RNAr RNA ribossomal; qualquer uma entre várias moléculas de RNA específicas que formam parte da estrutura de um ribossomo e participam na síntese de proteínas[11]. 20, 50, 58

RNAt RNA transportador; conjunto de pequenas moléculas de RNA, usadas na síntese de proteínas como uma interface entre o RNAm e os aminoácidos. Cada tipo de molécula de RNAt é covalentemente ligada a um determinado aminoácido[11]. 20, 50

sequência de consenso forma mais frequente de uma sequência, que é reproduzida com pequenas alterações em um grupo relacionado de sequências de DNA, RNA ou proteína[11]. 17, 32, 33, 38, 44, 46, 50, 52

sequenciador instrumento científico usado para automatizar o processo de sequenciamento[136]. 4, 11, 24, 45, 50, 60

sequenciamento determinação da composição e da ordem dos nucleotídeos ou aminoácidos em um ácido nucleico ou molécula proteica, gerando *reads*[11]. 4, 10, 11, 22–25, 27, 32, 33, 50–52, 55, 59, 60

singlet *read* sem sobreposição com nenhum outro[57]. 38, 50

sítio de *splice* sítios de *splice* são as junções entre íntrons e éxons no pré-RNAm de eucariontes[137]. 17, 44, 50

spliceossomo estrutura com atividade catalítica responsável pela execução do *splicing*[138]. 17, 50

splicing processo pelo qual sequências de íntrons são removidas dos transcritos de RNA no núcleo durante a formação do RNAm e de outros RNAs[11]. 17, 18, 50, 59

sstDNA ***s**ingle-**st**randed **t**emplate DNA*; DNA de fita simples que será utilizado como molde para a sua duplicação[35]. 25–27, 50

string qualquer sequência finita de caracteres de algum alfabeto[139]. 39, 41, 50, 58, 59

subsequência é uma sequência que pode ser derivada a partir de outra pela remoção de alguns elementos, sem mudar a ordem dos demais. Por exemplo, a *string* `ATTA` é uma subsequência de `GATATA`. Formalmente, uma subsequência de uma *string* $T = t_1 t_2 \dots t_n$ é uma *string* $\hat{T} = t_{i_1} \dots t_{i_m}$ tal que $i_1 < \cdots < i_m$, onde $m \leq n$. Toda *substring* é uma subsequência. [131, 140]. 50, 59

substrato molécula sobre a qual uma enzima atua[11]. 14, 27, 50

substring uma *substring* (ou fator) de uma *string* $T = t_1 \dots t_n$ é uma *string* $\hat{T} = t_{1+i} \dots t_{m+i}$, onde $0 \leq i$ and $m + i \leq n$. Em outras palavras, uma *substring* é uma *string* que faz parte (de modo contínuo) de uma *string* maior. Se $\hat{T}$ é uma *substring* of T, então também é uma subsequência de T[131]. 33, 39, 41, 50, 53, 58, 59

sufixo um prefixo de uma *string* $T = t_1 \dots t_n$ é uma *string* $\hat{T} = t_{n-m+1} \dots t_n$, onde $m \leq n$. Em outras palavras, é uma *substring* de T que acaba no último caractere (t_n)[131]. 33, 41, 50

superstring uma *superstring* de uma *string* T é uma *string* $\hat{T}$ tal que T é *substring* de $\hat{T}$. Em outras palavras, uma *superstring* é uma *string* que contém (de modo contínuo) uma *string* menor. 39–41, 50

TCC trabalho de conclusão de curso. 50, 75

telômero região de sequências repetitivas localizada nos terminais dos cromossomos, que protegem esses terminais da deterioração ou de se fundir com cromossomos vizinhos. Compensa a tendência de um cromossomo de sofrer encurtamento a cada ciclo de duplicação. Do grego *telos* (fim) e *meros* (parte) [11, 141]. 11, 50

terminal cego é um terminal de uma molécula de DNA em que ambas as fitas terminam em um par de bases. Um exemplo de molécula em que ambos os terminais são cegos é
`5′-CTGATCTGACTGATGCGTATGCTAGT-3′`
`3′-GACTAGACTGACTACGCATACGATCA-5′` [142]. 25, 50

terminal coesivo é um terminal de uma molécula de DNA em que uma das fitas possui nucleotídeos não pareados. Dois exemplos de moléculas cujos terminais coesivos são compatíveis (e portanto podem formar uma única molécula) são
`5′-ATCTGACT  + GATGCGTATGCT-3′`
`3′-TAGACTGACTACG  CATACGA-5′` [142]. 50

tiling path conjunto mínimo de BACs que contém todo o cromossomo com o mínimo possível de sobreposição entre os BACs[143]. 23, 24, 50

tradução processo no qual a sequência de nucleotídeos em uma molécula de RNAm direciona a incorporação de aminoácidos em uma proteína. Ocorre no ribossomo[11]. 19, 21, 49–51

transcrição reprodução de uma fita de DNA em uma sequência de RNA complementar, pela enzima RNA-polimerase[11]. 15–18, 50, 58

trimming sequenciadores de DNA podem produzir *reads* de baixa qualidade, principalmente perto do local do *primer* de sequenciamento e próximo ao final de longas corridas de sequenciamento. As sequências de clones de bibliotecas de DNA frequentemente contêm sequências de vetores, caudas poli-A ou outras sequências contaminantes. A não ser que sejam removidas (após identificação via mascaramento) num processo denominado *trimming* (que remove as bases contaminantes presentes nas extremidades dos *reads*), essas sequências contaminantes irão distorcer a montagem e a análise das sequências de interesse[144]. 42, 50

USP Universidade de São Paulo. 2, 50, 75

UTR região não traduzida (***un**translated **r**egion*); região não codificante de uma molécula de RNAm. A UTR $5'$ se estende desde o *cap* $5'$ até o códon de início da síntese proteica. A UTR $3'$ se estende desde o códon de parada da síntese proteica até o início da cauda poli-A[11]. 18, 50

vetor em biologia celular, é o DNA de um agente (vírus, plasmídeo ou BAC) usado para transmissão de material genético a uma célula ou organismo[11]. 34, 35, 50, 52, 55, 57, 60

vetor de clonagem é uma molécula de DNA pequena, geralmente derivada de um vírus ou plasmídeo, usada para carregar o fragmento de DNA a ser clonado para dentro da célula recipiente, possibilitando que este fragmento seja duplicado[11]. 50

Referências

[1] Wikipedia. Sequence assembly. Disponível em <http://en.wikipedia.org/wiki/Sequence_assembly>. Acesso em: 27 fev. 2012.

[2] Wikipedia. DNA sequencing. Disponível em <http://en.wikipedia.org/wiki/DNA_sequencing#Next-generation_methods>. Acesso em: 7 fev. 2013.

[3] Wikipedia. Sequence assembly: genome assemblers. Disponível em <http://en.wikipedia.org/wiki/Sequence_assembly#Genome_assemblers>. Acesso em: 27 fev. 2012.

[4] Wikipedia. Cis-regulatory element. Disponível em <http://en.wikipedia.org/wiki/Cis-regulatory_element>. Acesso em: 27 fev. 2012.

[5] POP, M.; SALZBERG, S. L.; SHUNWAY, M. Genome Sequence Assembly: Algorithms and Issues. Computer, v.35, n. 7, jul 2000. Disponível em <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.87.9580&rep=rep1&type=pdf>. Acesso em: 13 jul. 2012.

[6] Wikipedia. Single molecule real time sequencing. Disponível em <http://en.wikipedia.org/wiki/Single_molecule_real_time_sequencing>. Acesso em: 7 fev. 2013.

[7] Wikipedia. Single molecule real time (SMRT) sequencing. Disponível em <http://en.wikipedia.org/wiki/DNA_sequencing#Single_molecule_real_time_.28SMRT.29_sequencing>. Acesso em: 7 fev. 2013.

[8] Virtual medical centre. DNA. Disponível em <http://www.virtualmedicalcentre.com/anatomy/dna-deoxyribonucleic-acid/37>. Acesso em: 14 jul. 2012.

[9] Wikipedia. Alu element. Disponível em <http://en.wikipedia.org/wiki/Alu_element>. Acesso em: 13 jul. 2012.

[10] MORAN, L.A. The Human Genome Sequence Is not Complete. Disponível em <http://sandwalk.blogspot.com.br/2009/05/

human-genome-sequence-is-not-complete.html>. Acesso em: 13 jul. 2012.

[11] ALBERTS, B. et al. Biologia molecular da célula. 5ª edição. Porto Alegre: Artmed, 2009. 1396 p.

[12] Nehmi. O DNA. Disponível em <http://www.nehmi-ip.com.br/print.php?id=140&serv=10&faq=23>. Acesso em: 14 jul. 2012.

[13] Wikipedia. Ácido desoxirribonucleico. Disponível em <http://pt.wikipedia.org/wiki/%C3%81cido_desoxirribonucleico>. Acesso em: 14 jul. 2012.

[14] Wikipedia. Semiconservative replication. Disponível em <http://en.wikipedia.org/wiki/Semiconservative_replication>. Acesso em: 14 jul. 2012.

[15] Wikipedia. DNA replication. Disponível em <http://en.wikipedia.org/wiki/DNA_replication>. Acesso em: 14 jul. 2012.

[16] WALTER, M. DNA: The Genetic Material. Disponível em <http://bioserv.fiu.edu/~walterm/GenBio2004/chapter11_DNA/dna.htm>. Acesso em: 14 jul. 2012.

[17] SANTOS, S. Tradução é Transformação. Disponível em <http://aeducadora.blogspot.com.br/2010/05/traducao-e-transformacao-de-um-codigo.html>. Acesso em: 14 jul. 2012.

[18] Nehmi. O RNA. Disponível em <http://www.nehmi-ip.com.br/print.php?id=144&serv=10&faq=23>. Acesso em: 15 jul. 2012.

[19] ANSARI, A. RNA structures. Disponível em <http://www.uic.edu/classes/phys/phys461/phys450/ANJUM04/>. Acesso em: 15 jul. 2012.

[20] InfoEscola. Transcrição. Disponível em <http://www.infoescola.com/genetica/transcricao/>. Acesso em: 15 jul. 2012.

[21] Wikipedia. RNA splicing. Disponível em <http://en.wikipedia.org/wiki/RNA_splicing>. Acesso em: 16 jul. 2012.

[22] BioCoach. Concept 9: mRNA in Eukaryotes. Disponível em <http://www.phschool.com/science/biology_place/biocoach/transcription/mrnaeuk.html>. Acesso em: 16 jul. 2012.

[23] Wikipedia. Amino acid. Disponível em <http://en.wikipedia.org/wiki/Amino_acid>. Acesso em: 17 jul. 2012.

[24] Química10. Tabela de aminoácidos. Disponível em <http://quimica10.com.br/10/?tag=aminoacidos>. Acesso em: 17 jul. 2012.

[25] MARZZOCO, A.; TORRES, B. B. Bioquímica básica. 3ª edição. Rio de Janeiro: Guanabara Koogan, 2007. 404 p.

[26] Só Biologia. O Código Genético. Disponível em <http://www.sobiologia.com.br/conteudos/Citologia2/AcNucleico6.php>. Acesso em: 17 jul. 2012.

[27] Wikipedia. Genetic code. Disponível em <http://en.wikipedia.org/wiki/Genetic_code>. Acesso em: 17 jul. 2012.

[28] SILVA, A. B. Proteínas. Disponível em <http://portaldoprofessor.mec.gov.br/fichaTecnicaAula.html?aula=1599>. Acesso em: 17 jul. 2012.

[29] The University of New Mexico. DNA VERSUS RNA. Disponível em <http://biology.unm.edu/ccouncil/Biology_124/Summaries/T&T.html>. Acesso em: 18 jul. 2012.

[30] Genome News Network. SEQUENCING THE GENOME. Disponível em <http://www.genomenewsnetwork.org/articles/06_00/sequence_primer.shtml>. Acesso em: 19 jul. 2012.

[31] Wikipedia. Shotgun sequencing. Disponível em <http://en.wikipedia.org/wiki/Shotgun_sequencing>. Acesso em: 19 jul. 2012.

[32] Davidson College. Sequencing Whole Genomes. Disponível em <http://www.bio.davidson.edu/courses/genomics/method/shotgun.html>. Acesso em: 19 jul. 2012.

[33] COILA, B. DNA Sequencing Using BAC and Shotgun Methods. Disponível em <http://suite101.com/article/dna-sequencing-using-bac-and-shotgun-methods-a167492>. Acesso em: 19 jul. 2012.

[34] 454 Life Sciences. How genome sequencing is done. Disponível em <http://www.454.com/downloads/news-events/how-genome-sequencing-is-done_FINAL.pdf>. Acesso em: 24 jul. 2012.

[35] Wikipedia. 454 Life Sciences. Disponível em <http://en.wikipedia.org/wiki/454_Life_Sciences#Technology>. Acesso em: 24 jul. 2012.

[36] GaTE Lab. 454 Sequencing. Disponível em <https://gate.ib.usp.br/GateWeb/?q=pt-br/system/files/454.ppt>. Acesso em: 24 jul. 2012.

[37] University of California Santa Cruz. Overview of The 454 Sequencing System. Disponível em <http://classes.soe.ucsc.edu/bme215/Spring09/PPT/BME%20215-5.pdf>. Acesso em: 24 jul. 2012.

[38] Wikipedia. Pyrosequencing. Disponível em <http://en.wikipedia.org/wiki/Pyrosequencing>. Acesso em: 24 jul. 2012.

[39] McClean, P. E. DNA Sequencing Notes. Disponível em <http://www.ndsu.edu/pubweb/~mcclean/plsc731/Genome-sequencing-PMG-overheads.pdf>. Acesso em: 24 jul. 2012.

[40] MEYER, M. et al. Targeted high-throughput sequencing of tagged nucleic acid samples. Disponível em <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1976447/>. Acesso em: 25 jul. 2012.

[41] UT GSAF. 454 - all flavors. Disponível em <https://wikis.utexas.edu/display/GSAF/454+-+all+flavors>. Acesso em: 25 jul. 2012.

[42] MOUNT, D. W. Bioinformatics: sequence and genome analysis. 1ª ed. Nova Iorque: Cold Spring Harbor, 2001. 565 p.

[43] SETUBAL, J. C.; MEIDANIS, J. Introduction to computational molecular biology. 1ª ed. Boston: PWS, 1997. 308 p.

[44] PROSDOCIMI, F. CURSO ON LINE - INTRODUÇÃO À BIOINFORMÁTICA. Disponível em <www2.bioqmed.ufrj.br/prosdocimi/FProsdocimi07_CursoBioinfo.pdf>. Acesso em: 1 ago. 2012.

[45] Wikipedia. Sequence alignment. Disponível em <http://en.wikipedia.org/wiki/Sequence_alignment>. Acesso em: 1 ago. 2012.

[46] SETUBAL, J. C.; BRAEUNING, R. In: GRUBER, A. (Org.) et al. Similarity Search. *Bioinformatics in Tropical Disease Research: A Practical and Case-Study Approach*. Disponível em <http://www.ncbi.nlm.nih.gov/books/NBK6831/>. Acesso em: 13 jan. 2012.

[47] LIMA, A. M. Alinhamentos e Busca de Similaridade. Disponível em <http://www.ime.usp.br/posbioinfo/ci2008/apresentacoes/alinhamentos-ariane.pdf>. Acesso em: 1 ago. 2012.

[48] School of engineering and applied science. Sequence alignment. Disponível em <http://www.seas.gwu.edu/~simhaweb/cs151/lectures/module12/align.html>. Acesso em: 1 ago. 2012.

[49] KORF, I.; YANDELL, M.; BEDELL, J. Blast. 1ª ed. California: O'Reilly, 2003. 368 p.

[50] ALTSCHUL, S. F. et al. Basic local alignment search tool. J Mol Biol, v. 215 , n. 3, p. 403-10, 5 Out. 1990.

[51] KENT, W. J. BLAT—The BLAST-Like Alignment Tool. Disponível em <http://genome.cshlp.org/content/12/4/656.full>. Acesso em: 11 ago. 2012.

[52] LEDERGERBER, C.; DESSIMOZ, C. Base-calling for next-generation sequencing platforms. Bioinform (2011) 12 (5): 489-497. Disponível em <http://bib.oxfordjournals.org/content/12/5/489.full>. Acesso em: 4 jan. 2013.

[53] MILLER, J. R.; KOREN, S.; SUTTON, G. Assembly algorithms for next-generation sequencing data. Genomics 95 (2010) 315–327. Disponível em <http://www.sciencedirect.com/science/article/pii/S0888754310000492>. Acesso em: 3 jan. 2013.

[54] NARZISI, G.; MISHRA, B. Comparing De Novo Genome Assembly: The Long and Short of It. PLoS ONE 6(4): e19175. Disponível em <http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0019175>. Acesso em: 3 jan. 2013.

[55] MAISINGER, K. Base-calling and quality scoring. Disponível em <http://www.ebi.ac.uk/industry/Documents/workshop-materials/newsequence291009/Basecalling-Klaus_Maisinger.pdf>. Acesso em: 25 jul. 2012.

[56] MYERS, E. W. et al. A Whole-Genome Assembly of *Drosophila*. Science 287, 2196 (2000).

[57] COSTA, G. G. L. Introdução à montagem de genomas. Disponível em <http://www.lge.ibi.unicamp.br/cursobioinfo2012/aula07.pdf>. Acesso em: 9 jan. 2013.

[58] PHILLIPPY, A. M.; SCHATZ, M. C.; POP, M. Genome assembly forensics: finding the elusive mis-assembly. Disponível em <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2397507/>. Acesso em: 10 jan. 2013.

[59] Davidson College. Phage assembly suite and tutorial (PHAST). PLoS ONE 6(4): e19175. Disponível em <http://gcat.davidson.edu/phast/>. Acesso em: 9 jan. 2013.

[60] CRESCENZI, P.; KANN, V. SHORTEST COMMON SUPERSTRING. Disponível em <http://www.nada.kth.se/~viggo/wwwcompendium/node166.html>. Acesso em: 3 jan. 2013.

[61] MEDVEDEV, P. et al. Computability of Models for Sequence Assembly. Disponível em <http://www.cse.psu.edu/~pashadag/wabi07.pdf>. Acesso em: 3 jan. 2013.

[62] AUSIELLO, G. et al. Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties. 1ª ed. Nova Iorque: Springer, 2003. 543 p.

[63] CRESCENZI, P.; KANN, V. A compendium of NP optimization problems. Disponível em <http://www.nada.kth.se/~viggo/wwwcompendium/>. Acesso em: 3 jan. 2013.

[64] CRESCENZI, P.; KANN, V. SHORTEST COMMON SUPERSTRING. Disponível em <http://www.nada.kth.se/~viggo/wwwcompendium/node166.html>. Acesso em: 3 jan. 2013.

[65] MNEIMNEH, S. DNA sequencing and the shortest superstring problem. Disponível em <http://www.cs.hunter.cuny.edu/~saad/courses/compbio/lectures/lecture15.pdf>. Acesso em: 3 jan. 2013.

[66] Wikipedia. NP-hard. Disponível em <http://en.wikipedia.org/wiki/NP-hard>. Acesso em: 17 jan. 2013.

[67] GAREY, M. R.; JOHNSON, D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness. 1ª ed. Nova Iorque: W. H. Freeman & Co., 1979. 338 p.

[68] DE CARVALHO, M. H. et al. Uma Introdução Sucinta a Algoritmos de Aproximação. Disponível em <http://www.ime.usp.br/~cris/aprox/livro.pdf>. Acesso em: 17 jan. 2013.

[69] BLUM, A. et al. Linear approximation of shortest superstrings. Disponível em <https://www.cs.cmu.edu/afs/cs/usr/avrim/www/Papers/superstring.pdf>. Acesso em: 20 jan. 2013.

[70] Wikipedia. APX. Disponível em <http://en.wikipedia.org/wiki/APX>. Acesso em: 20 jan. 2013.

[71] Wikipedia. Polynomial-time approximation scheme. Disponível em <http://en.wikipedia.org/wiki/Polynomial-time_approximation_scheme>. Acesso em: 20 jan. 2013.

[72] WILLIAMSON, D. P.; SHMOYS, D. B. The Design of Approximation Algorithms. Disponível em <http://www.designofapproxalgs.com/book.pdf>. Acesso em: 20 jan. 2013.

[73] WEINARD, M.; SCHNITGER, G. On the greedy superstring conjecture. Disponível em <http://www.thi.informatik.uni-frankfurt.de/~weinard/Publications/fsttcs.pdf>. Acesso em: 20 jan. 2013.

[74] CROCHEMORE, M. et al. Algorithms for Three Versions of the Shortest Common Superstring Problem. Disponível em <www.cs.ucr.edu/~stelo/cpm/cpm10/27.pdf>. Acesso em: 20 jan. 2013.

[75] KAPLAN, H.; SHAFRIR, N. The greedy algorithm for shortest superstrings. Disponível em <www.math.tau.ac.il/~haimk/papers/greedy3.5.2.ps>. Acesso em: 20 jan. 2013.

[76] TARHIO, J.; UKKONEN, E. A greedy approximation algorithm for constructing shortest common superstrings. Disponível em <http://pdn.sciencedirect.com/science?_ob=MiamiImageURL&_cid=271538&_user=10&_pii=0304397588901673&_check=y&_origin=article&_zone=toolbar&_coverDate=1988--30&view=c&originContentFamily=serial&wchp=dGLzVBA-zSkWA&md5=77c04cc3857bb1c5c4c2c8b77a7a6228&pid=1-s2.0-0304397588901673-main.pdf>. Acesso em: 20 jan. 2013.

[77] PALUSZEWSKI, M. Approximating the Shortest Superstring Problem. Disponível em <http://fileadmin.cs.lth.se/cs/Personal/Andrzej_Lingas/superstring.pdf>. Acesso em: 4 fev. 2013.

[78] SWEEDYK, Z. A $2\frac{1}{2}$-Approximation Algorithm for Shortest Superstring. SIAM J. Comput. 29(3): 954-986 (1999).

[79] DURHAM, A. M. et al. EGene: a configurable pipeline generation system for automated sequence analysis. Bioinformatics 21(12): 2812-2813.

[80] GREEN, P. Phrap/Cross_match/Swat. Disponível em <http://www.phrap.org/phredphrapconsed.html#block_phrap>. Acesso em: 4 fev. 2013.

[81] GREEN, P. phrap/cross_match/swat documentation. Disponível em <http://www.phrap.org/phredphrap/general.html>. Acesso em: 4 fev. 2013.

[82] NCBI. The UniVec Database. Disponível em <http://www.ncbi.nlm.nih.gov/VecScreen/UniVec.html>. Acesso em: 4 fev. 2013.

[83] NCBI. Cloning vector pBeloBAC11. Disponível em <http://www.ncbi.nlm.nih.gov/nuccore/1817728>. Acesso em: 4 fev. 2013.

[84] Wikipedia. Balanceamento de carga. Disponível em <http://pt.wikipedia.org/wiki/Balanceamento_de_carga>. Acesso em: 6 fev. 2013.

[85] MOUNT, S. M. Splice Site Consensus. Disponível em <http://www.life.umd.edu/labs/mount/RNAinfo/consensus.html>. Acesso em: 7 fev. 2013.

[86] STANKE, M. blat2hints.pl. Disponível em <http://augustus.gobics.de/binaries/scripts/blat2hints.pl>. Acesso em: 10 fev. 2013.

[87] CHEVREUX, B. et al. MIRA - Sequence assembler and mapper for whole genome shotgun and EST / RNASeq sequencing data. Disponível em <http://sourceforge.net/projects/mira-assembler/>. Acesso em: 10 fev. 2013.

[88] Roche. GS De Novo Assembler. Disponível em <http://454.com/products/analysis-software/index.asp>. Acesso em: 10 fev. 2013.

[89] SOBREIRA, T. J. P.; GRUBER, A. Sequence-specific reconstruction from fragmentary databases using seed sequences: implementation and validation on SAGE, proteome and generic sequencing data. Disponível em <http://bioinformatics.oxfordjournals.org/content/24/15/1676.full>. Acesso em: 10 fev. 2013.

[90] Wikipedia. Tab-separated values. Disponível em <http://en.wikipedia.org/wiki/Tab-separated_values>. Acesso em: 11 fev. 2013.

[91] Wikipedia. PNG. Disponível em <http://pt.wikipedia.org/wiki/PNG>. Acesso em: 11 fev. 2013.

[92] PlantGDB. PlantGDB Download Portal. Disponível em <http://www.plantgdb.org/download/Download/PublicPlantSeq/Dump/S/Sorghum_bicolor/FASTA/Sorghum_bicolor.Protein.fasta.bz2>. Acesso em: 11 fev. 2013.

[93] DILLON, S. L. et al. Domestication to Crop Improvement: Genetic Resources for Sorghum and Saccharum (Andropogoneae). Disponível em <http://aob.oxfordjournals.org/content/100/5/975.full>. Acesso em: 11 fev. 2013.

[94] Wikipedia. Promoter (genetics). Disponível em <http://en.wikipedia.org/wiki/Promoter_(genetics)>. Acesso em: 11 fev. 2013.

[95] Wikipedia. Adapter. Disponível em <http://en.wikipedia.org/wiki/Adapter_(genetics)>. Acesso em: 25 jul. 2012.

[96] Wikipedia. Alinhamento de sequências. Disponível em <http://pt.wikipedia.org/wiki/Alinhamento_de_seq%C3%BC%C3%AAncias>. Acesso em: 25 jul. 2012.

[97] Wikipedia. Reação em cadeia da polimerase. Disponível em <http://pt.wikipedia.org/wiki/Rea%C3%A7%C3%A3o_em_cadeia_da_polimerase>. Acesso em: 25 jul. 2012.

[98] Wikipedia. 5' cap. Disponível em <http://en.wikipedia.org/wiki/5'_cap>. Acesso em: 25 jul. 2012.

[99] Wikipedia. Catálise. Disponível em <http://pt.wikipedia.org/wiki/Cat%C3%A1lise>. Acesso em: 25 jul. 2012.

[100] Wikipedia. Polyadenylation. Disponível em <http://en.wikipedia.org/wiki/Polyadenylation>. Acesso em: 25 jul. 2012.

[101] Wikipedia. Coding Region. Disponível em <http://en.wikipedia.org/wiki/Coding_sequence>. Acesso em: 25 jul. 2012.

[102] Wikipedia. Cell. Disponível em <http://en.wikipedia.org/wiki/Cell_(biology)>. Acesso em: 25 jul. 2012.

[103] Wikipedia. Centrômero. Disponível em <http://pt.wikipedia.org/wiki/Centr%C3%B3mero>. Acesso em: 25 jul. 2012.

[104] Wikipedia. Citoplasma. Disponível em <http://pt.wikipedia.org/wiki/Citoplasma>. Acesso em: 25 jul. 2012.

[105] Wikipedia. Coverage. Disponível em <http://en.wikipedia.org/wiki/Shotgun_sequencing#Coverage>. Acesso em: 25 jul. 2012.

[106] NCBI. Contamination in Sequence Databases. Disponível em <http://www.ncbi.nlm.nih.gov/VecScreen/contam.html>. Acesso em: 4 fev. 2012.

[107] Wikipedia. Contig. Disponível em <http://en.wikipedia.org/wiki/Contig>. Acesso em: 25 jul. 2012.

[108] Wikipedia. Cromatídio. Disponível em <http://pt.wikipedia.org/wiki/Cromat%C3%ADdio>. Acesso em: 26 jul. 2012.

[109] Wikipedia. Cultivar. Disponível em <http://pt.wikipedia.org/wiki/Cultivar>. Acesso em: 10 fev. 2013.

[110] Wikipedia. Nucleic acid denaturation. Disponível em <http://en.wikipedia.org/wiki/Denaturation_(biochemistry)#Nucleic_acid_denaturation>. Acesso em: 26 jul. 2012.

[111] Wikipedia. DNA. Disponível em <http://en.wikipedia.org/wiki/DNA>. Acesso em: 26 jul. 2012.

[112] Wikipedia. Emulsão. Disponível em <http://pt.wikipedia.org/wiki/Emuls%C3%A3o>. Acesso em: 26 jul. 2012.

[113] UFSC. Enzimas. Disponível em <http://www.enq.ufsc.br/labs/probio/disc_eng_bioq/trabalhos_pos2003/const_microorg/enzimas.htm>. Acesso em: 26 jul. 2012.

[114] Wikipedia. Expressed sequence tag. Disponível em <http://en.wikipedia.org/wiki/Expressed_sequence_tag>. Acesso em: 5 fev. 2013.

[115] VestibulandoWeb. Célula Eucarionte. Disponível em <http://www.vestibulandoweb.com.br/biologia/teoria/celula-eucarionte.asp>. Acesso em: 26 jul. 2012.

[116] Wikipedia. Formato FASTA. Disponível em <http://pt.wikipedia.org/wiki/Formato_FASTA>. Acesso em: 4 fev. 2013.

[117] Wikipedia. Gene. Disponível em <http://pt.wikipedia.org/wiki/Gene>. Acesso em: 26 jul. 2012.

[118] DNA Sequencing Core. How do we Sequence DNA?. Disponível em <http://seqcore.brcf.med.umich.edu/doc/educ/dnapr/sequencing.html>. Acesso em: 26 jul. 2012.

[119] Genome News Network. What types of genome maps are there?. Disponível em <http://www.genomenewsnetwork.org/resources/whats_a_genome/Chp3_2.shtml>. Acesso em: 26 jul. 2012.

[120] Mouse Genome Informatics. GENETIC MAPS COME IN VARIOUS FORMS. Disponível em <http://www.informatics.jax.org/silver/chapters/7-1.shtml>. Acesso em: 26 jul. 2012.

[121] EGassembler. EGassembler Tutorial. Disponível em <http://egassembler.hgc.jp/cgi-bin/eassembler4.cgi?pmode=help&i_param=tutorial>. Acesso em: 4 fev. 2012.

[122] Wikipedia. Paired-end tag. Disponível em <http://en.wikipedia.org/wiki/Mate_pair>. Acesso em: 11 ago. 2012.

[123] Wikipedia. Cell membrane. Disponível em <http://en.wikipedia.org/wiki/Cell_membrane>. Acesso em: 26 jul. 2012.

[124] Wikipedia. Monômero. Disponível em <http://pt.wikipedia.org/wiki/Mon%C3%B4mero>. Acesso em: 26 jul. 2012.

[125] Wikipedia. Nucleotide. Disponível em <http://en.wikipedia.org/wiki/Nucleotide>. Acesso em: 26 jul. 2012.

[126] Wikipedia. Otimização. Disponível em <http://pt.wikipedia.org/wiki/Otimiza%C3%A7%C3%A3o>. Acesso em: 17 jan. 2013.

[127] Wikipedia. P versus NP problem. Disponível em <http://en.wikipedia.org/wiki/P_versus_NP_problem>. Acesso em: 20 jan. 2013.

[128] Wikipedia. Pipeline. Disponível em <http://en.wikipedia.org/wiki/Pipeline_(software)>. Acesso em: 26 jul. 2012.

[129] Infopédia. Polimerização. Disponível em <http://www.infopedia.pt/$polimerizacao>. Acesso em: 26 jul. 2012.

[130] UFSC. Polímeros. Disponível em <http://www.qmc.ufsc.br/qmcweb/artigos/polimeros.html>. Acesso em: 26 jul. 2012.

[131] Wikipedia. Substring. Disponível em <http://en.wikipedia.org/wiki/Substring>. Acesso em: 12 jan. 2013.

[132] Wikipedia. Iniciador. Disponível em <http://pt.wikipedia.org/wiki/Iniciador>. Acesso em: 26 jul. 2012.

[133] Wikipedia. Procarionte. Disponível em <http://pt.wikipedia.org/wiki/Procarionte>. Acesso em: 26 jul. 2012.

[134] UFSC. Proteínas. Disponível em <http://www.enq.ufsc.br/labs/probio/disc_eng_bioq/trabalhos_pos2003/const_microorg/proteinas.htm>. Acesso em: 26 jul. 2012.

[135] Wikipedia. RNA. Disponível em <http://en.wikipedia.org/wiki/RNA>. Acesso em: 11 ago. 2012.

[136] Wikipedia. DNA sequencer. Disponível em <http://en.wikipedia.org/wiki/DNA_sequencer>. Acesso em: 11 ago. 2012.

[137] Chemistry of Life. splice-site. Disponível em <http://chemistryolife.blogspot.com.br/2007/12/splice-site_06.html>. Acesso em: 7 fev. 2013.

[138] Wikipedia. Splicing. Disponível em <http://pt.wikipedia.org/wiki/Splicing>. Acesso em: 11 ago. 2012.

[139] Wikipedia. String (computer science). Disponível em <http://en.wikipedia.org/wiki/String_(computer_science)>. Acesso em: 12 jan. 2013.

[140] Wikipedia. Subsequence. Disponível em <http://en.wikipedia.org/wiki/Subsequence>. Acesso em: 13 jan. 2013.

[141] Wikipedia. Telomere. Disponível em <http://en.wikipedia.org/wiki/Telomere>. Acesso em: 11 ago. 2012.

[142] Wikipedia. Sticky and blunt ends. Disponível em <http://en.wikipedia.org/wiki/Sticky_and_blunt_ends>. Acesso em: 11 ago. 2012.

[143] The Maize Full Length cDNA Project. Glossary. Disponível em <http://www.maizecdna.org/outreach/glossary.html>. Acesso em: 11 ago. 2012.

[144] Gene Codes Corporation. Sequence Trimming. Disponível em <http://genecodes.com/sequencher-features/sequence-trimming>. Acesso em: 6 fev. 2013.

[145] Universidade de São Paulo. Manual do Calouro 2012. Disponível em <http://biton.uspnet.usp.br/marketing/manual_2012.pdf>. Acesso em: 23 jul. 2012.

[146] Wikipedia. Standard Flowgram Format. Disponível em <http://en.wikipedia.org/wiki/Standard_Flowgram_Format>. Acesso em: 11 fev. 2013.

[147] Wikipedia. Earliest deadline first scheduling. Disponível em <http://en.wikipedia.org/wiki/Earliest_deadline_first_scheduling>. Acesso em: 10 fev. 2013.

[148] FERREIRA, C. E. Roteiro para preparação de monografias. Disponível em <http://www.ime.usp.br/~cef/mac499-12/rot-monografias.html>. Acesso em: 10 fev. 2013.

[149] Wikipedia. Dynamical system. Disponível em <http://en.wikipedia.org/wiki/Dynamical_system>. Acesso em: 2 dez. 2012.

Parte II

Parte Subjetiva

9 Desafios e frustrações

9.1 Em relação ao curso

Entrei no BCC em 2007, após prestar 3 vezes o vestibular e fazer 2 anos de cursinho. Escolhi o BCC por achar que teria alguma facilidade por já ter feito um curso técnico em informática[61] (o que se revelou um engano logo no primeiro semestre, pois tive que aprender quase tudo a partir do início) e também por ter apoio para isso (principalmente por causa do mercado de trabalho). A computação "por si só" nunca foi uma das coisas mais atraentes pra mim, já que para resolver qualquer problema relevante é necessário ter uma boa base conceitual relativa ao seu domínio (não basta conhecer algoritmos eficientes para resolver um problema se a modelagem do mesmo não for bem feita).

Logo na primeira semana do curso, ouvi o professor Paulo Cordaro[62] falar sobre a pós-graduação em bioinformática, e foi a primeira vez que tive contato com o assunto (que eu nem imaginava existir). A partir de então resolvi me preparar para entrar na área, e imaginei que ter uma formação interdisciplinar seria imprescindível[63]. Passei a cursar disciplinas em outros institutos (via requerimento de matrícula, principalmente no IB, no IQ e na Poli), o que me gerou (felizmente) várias experiências enriquecedoras, mas também a pior experiência acadêmica que tive na vida (que envolveu uma expulsão arbitrária e humilhante de uma sala de aula, mesmo com um requerimento em andamento).

Infelizmente, ainda existem docentes que aparentam não saber o que é uma Universidade. A seguir está transcrita uma parte do Manual do Calouro 2012 [145] sobre o assunto:

"A criação da Universidade - que surgiu no século 12, na Europa - representou a concretização do conceito platônico de espírito. No seu mais famoso livro, *A República*, o filósofo grego Platão (427-347 antes de Cristo) afirma que a alma é 'de certo modo todas as coisas divinas e humanas e deve travar relações com tudo o que é'. Segundo Platão, *o espírito precisa estar aberto para tudo o que existe no mundo - e não apenas uma parte dele*[64].

[61] algo que fiz sob pressão, porque um irmão meu já tinha feito esse mesmo curso e conseguiu um emprego na época.

[62] o então diretor do IME.

[63] pensei em cursar Ciências Moleculares, mas não o fiz por não ter certeza se teria apoio e se queria seguir a carreira acadêmica.

[64] ênfase minha.

"Foi essa concepção do homem e da alma que presidiu à fundação da Universidade de Paris, em 1215. Nela, analisava-se qualquer objeto de estudo - fosse o corpo humano, a política ou as Sagradas Escrituras - sempre em relação com todo o Universo e com ampla liberdade. 'Esse é o verdadeiro conceito de Universidade', afirma o medievalista Jean Lauand, professor da Faculdade de Educação da USP. '*Se não houver essa conexão com o todo e essa liberdade, não é uma Universidade*'[64]. "

Em relação à graduação, foi frustrante ouvir frequentemente frases como "vou acabar o curso em 5 anos[65], mas é porque quero fazer estágio", sendo que minha dedicação ao curso foi integral e irei precisar de 6 anos (no mínimo) para concluí-lo. Vários fatores contribuíram para isso, mas acredito que os principais foram a dedicação exigida nas disciplinas (tipicamente maior que o tempo disponível e/ou registrado oficialmente nas ementas) e o relativo isolamento das pessoas que fazem o curso (embora eu também assuma a minha parte da culpa e me inclua no grupo dos "eremitas").

A grande maioria das disciplinas do curso dá apenas créditos-aula e nenhum crédito-trabalho, o que é incompatível com o fato de que vários tipos de trabalhos (exercícios-programas, listas de exercício, projetos e estudos individuais) são feitos fora da aula (e exigem uma quantidade de tempo considerável para serem realizados). Some-se a isso o fato de que os alunos costumam fazer em torno de 5 disciplinas por semestre[66], e que normalmente os professores não sabem quantas (e quais) disciplinas os alunos fazem[67], o que acaba gerando cargas de trabalho imensas para o pouco tempo livre[68] que temos. As ditas "semanas de *break*" [69] ajudam bastante, mas às vezes não são suficientes para evitar a reprovação nas disciplinas mais exigentes (o que atrasa a conclusão do curso).

Outro problema foi minha falta de integração com grande parte da minha turma (BCC 2007), algo que acabou ocorrendo por falta de iniciativa de ambos os lados (afinal, eu também não sou a pessoa mais sociável do mundo). Por conta disso, várias vezes me vi sem qualquer esperança de passar em determinadas disciplinas (por não ter a quem recorrer), o que me levava à reprovação ou ao trancamento de matrícula [70]. Meu aproveitamento era sempre melhor quando tinha a oportunidade de fazer disciplinas com pessoas que eu conhecia e com quem eu me relacionava, até porque nos ajudávamos.

O curso possui uma parte teórica bem acentuada e desenvolvida (o que pra mim foi bom, pois o meu perfil é mesmo mais teórico), mas o problema fica por conta da parte prática. Várias disciplinas exigem a execução de projetos e exercícios-programa, mas o conhecimento tecnológico necessário

[65] a duração ideal é de 4 anos.

[66] o que consome quase todo o horário da semana.

[67] infelizmente nem todos estão no período ideal, o que acaba gerando grades horárias variadas entre os alunos.

[68] geralmente finais de semana e feriados.

[69] normalmente 3 por semestre, nas quais não costuma haver aulas das disciplinas de computação (que são as de sigla MAC0XXX).

[70] quando a esperança era perdida a tempo.

(explicações sobre a utilização de APIs[71], arcabouços[72] e linguagens de programação) normalmente não é coberto durante as aulas. Por mais que se diga que os conceitos que aprendemos durante a graduação nos permitem "dominar rapidamente toda e qualquer tecnologia"[73], não é fácil aprender algo do dia para a noite[74], e geralmente é mais difícil aprender sozinho[75]. Portanto, além da carga horária fora da aula utilizada para fazer os trabalhos, muitas vezes também é preciso aprender como utilizar as tecnologias envolvidas, o que acaba consumindo ainda mais tempo. Em suma, acho que a parte prática mereceria um pouco mais de "atenção supervisionada", principalmente em relação ao desenvolvimento para web (que não é coberto nas disciplinas obrigatórias).

9.2 Em relação ao TCC

As dificuldades do trabalho de conclusão de curso (TCC) contêm as da IC (já que o primeiro é baseado na segunda): aprofundamento e sedimentação de conhecimentos de biologia molecular (talvez não tanto pela parte acadêmica [76], mas mais pela parte burocrática[77]), pré-processamento dos dados (a saída do sequenciador Roche/454 não estava no formato FASTA[78]; foi necessário aprender a usar o EGene para a fase de mascaramento) e desenvolvimento dos *pipelines* (pois eu não tinha muita prática com a linguagem Perl).

Também foi difícil conciliar o trabalho com todas as disciplinas da graduação (5 no primeiro semestre de 2012 e 6 no segundo semestre), que ocuparam grande parte do tempo (com conteúdos não triviais e/ou projetos trabalhosos). Como tinha muitas coisas para fazer em paralelo, acabei usando a política EDF (*earliest deadline first*)[79][147] para o processamento de tarefas, o que se mostrou razoavelmente eficiente (pois apenas precisei trancar uma disciplina em 2012) mas extremamente custoso (pois acabei ficando de recuperação em duas disciplinas, o que consumiu as férias).

O desenvolvimento da monografia foi extremamente trabalhoso, pois é difícil explicar um trabalho de natureza interdisciplinar de modo a "ser entendido por um aluno de graduação sem experiência na área" (como exigido no roteiro para preparação de monografias[148]). Além da elaboração do texto (e organização das respectivas citações), um tempo razoável foi gasto para procurar e modificar figuras (que geralmente facilitam o entendimento).

[71] acrônimo para *Application Programming Interface* (Interface de Programação de Aplicativos).
[72] mais conhecidos como *frameworks*.
[73] "guia do bicho" 2007 (IME-USP).
[74] pelo menos eu não aprendo ...
[75] fiz dois cursos de verão do IME sobre Java que me ajudaram bastante a entender um pouco melhor a linguagem.
[76] pois tive aulas com professores excelentes.
[77] principalmente devido à já mencionada expulsão arbitrária de uma sala de aula...
[78] e sim no formato SFF[146].
[79] que consiste em escolher para executar a tarefa que estiver mais próxima do seu prazo de entrega (*deadline*).

Devido a essas dificuldades acabei não dedicando muito tempo ao *blog*, pois imaginei que as outras atividades fossem mais importantes. Apesar disso, aconteceram várias atividades que deveriam ter sido registradas no *blog*, como as reuniões com meu orientador (que ocorriam semanalmente).

10 Disciplinas relevantes e conceitos utilizados

Aqui serão descritas quais foram as disciplinas cursadas que foram mais relevantes (direta ou indiretamente) para a execução do trabalho.

10.1 Cursadas no IME

- **MAC0110 - Introdução à Computação** e **MAC0122 - Princípios de Desenvolvimento de Algoritmos**. Foram essas disciplinas que permitiram um maior conhecimento das técnicas e estruturas de dados utilizadas comumente em computação, além proporcionarem experiência de programação com a linguagem C (o que foi útil para entender um pouco melhor a linguagem Perl, utilizada no desenvolvimento dos *pipelines*).

- **MAC0211 - Laboratório de Programação I**. Foi nessa disciplina que vi pela primeira vez expressões regulares, concatenação de programas via *pipelines*, a linguagem Perl (todos esses conceitos foram utilizados nos *pipelines* desenvolvidos) e LaTeX (utilizado para fazer a monografia, o pôster e a apresentação).

- **MAC0422 - Sistemas Operacionais** e **MAC0431 - Introdução à Computação Paralela e Distribuída**. Em MAC0422 foram introduzidos os conceitos de processo e paralelização, ambos utilizados nas fases de alinhamento e montagem inicial do *pipeline* de montagem. Em MAC0431 tais conceitos foram aprofundados, considerando a parte de análise de dependência (imprescindível para obter uma versão paralela de um algoritmo a partir de uma versão sequencial).

- **MAC0316 - Conceitos Fundamentais de Linguagens de Programação** e **MAC0319 - Programação Funcional Contemporânea**[80]. São essas as disciplinas responsáveis por facilitar o aprendizado de linguagens de programação, pois ensinam os conceitos fundamentais relacionados a elas (e por isso também foram úteis para aprender mais sobre Perl, a linguagem utilizada para desenvolver os *pipelines*).

- **MAC0465 - Biologia Computacional**. Foi essa disciplina que tratou dos aspectos computacionais de problemas relacionados ao trabalho, como alinhamento e montagem de sequên-

[80] que cursei sob a sigla MAC0434, na época em que MAC0319 ainda não havia sido criada.

cias. Com isso, foi possível entender um pouco melhor a natureza desses problemas e as limitações dos algoritmos existentes para resolvê-los.

- **MAC0325 - Otimização Combinatória** e **MAC0450 - Algoritmos de Aproximação**. Tais disciplinas foram fundamentais para treinar técnicas de modelagem de problemas, algo que não é feito de forma tão explícita nas outras disciplinas do curso. Também foi bom cursá-las entender um pouco mais a importância de programação linear (que vai bem além do método simplex), já que é uma técnica que pode ser usada tanto para projeto quanto para análise de algoritmos. Muitos problemas que aparecem em biologia computacional são NP-difíceis (e, pior ainda, trabalham com quantidades enormes de dados) e, portanto, algoritmos de aproximação são fundamentais para poder lidar com eles.

10.2 Cursadas em outras unidades

- **QFL0605 - Química Geral**. Essa disciplina tratou de expor os tipos de forças intramoleculares (ligações químicas) e intermoleculares, além de conceitos de termodinâmica (entropia, entalpia, energia livre) e cinética (estudo da velocidade de reações químicas). Esses conceitos foram fundamentais para entender melhor bioquímica e biologia molecular.

- **BIO0228 - Genética Humana**. Essa disciplina trata de estudar estuda a passagem das características biológicas e físicas de geração para geração. Aqui foi introduzido o conceito de gene e foram estudados os padrões de herança de diversas características. Tais fundamentos foram úteis para um melhor entendimento de biologia molecular.

- **QBQ0204 - Bioquímica e Biologia Molecular**. Essa disciplina foi focada em bioquímica. Os principais conceitos expostos foram as propriedades de biomoléculas (aminoácidos, peptídeos, proteínas, lipídeos e carboidratos) e as estruturas de vias metabólicas, e também foram úteis para um melhor entendimento de biologia molecular.

- **QBQ0317 - Biologia Molecular** e **BIB0525 - Biologia Molecular de Plantas**. Ambas as disciplinas foram cruciais para uma compreensão mais profunda dos processos biológicos relacionados ao trabalho (principalmente duplicação, transcrição e tradução). A abordagem das disciplinas também foi bem interessante: além das aulas teóricas, existia tempo *durante o horário de aula* para resolver exercícios, o que facilitava bastante a consolidação dos conceitos (principalmente porque os professores ficavam disponíveis para tirar dúvidas). Em BIB0525, a maioria dos exercícios era baseada em artigos científicos recentes e as resoluções eram sempre discutidas com toda a sala, o que enriquecia bastante a aula (em alguns casos, as discussões nos faziam perceber que havia mais de uma resposta possível para um mesmo

exercício)[81].

- **BIO0208 - Processos Evolutivos**. A evolução trata de entender quais mecanismos são responsáveis por gerar e moldar a variação genética existente, e por isso é fundamental para o entendimento de todas as áreas da biologia. Em particular, foi útil para descobrir alguns possíveis causas da variação do tamanho e da complexidade de genomas (que são alguns dos fatores relacionados ao problema da montagem de sequências), como proliferação de transposons, duplicações gênicas e inserção/remoção de bases.

- **QBQ2507 - Biologia Molecular Computacional**. Além de tratar de algumas técnicas computacionais usadas para resolver problemas de biologia molecular, a disciplina também falou da interpretação biológica de alinhamentos e de árvores filogenéticas (discutindo as hipóteses evolutivas supostas pelos algoritmos), e portanto foi complementar a MAC0465 (Biologia Computacional). Também foi dada alguma atenção à parte tecnológica, com indicação de softwares comumente utilizados para resolver os problemas da área. Assim sendo, a disciplina foi útil para adquirir um pouco mais de experiência em bioinformática.

- **PTC2422 - Modelos de Sistemas Biológicos**. Assim como MAC0325 (Otimização Combinatória) e MAC0450 (Algoritmos de Aproximação), essa disciplina também foi útil para treinar a parte de modelagem de problemas (em particular, problemas de natureza biológica, embora o que foi visto na disciplina sirva para quaisquer tipos de sistemas dinâmicos[82]). Também foi bom cursá-la para entender um pouco mais a importância de MAT0221 (Cálculo Diferencial e Integral IV), já que as modelagens quase sempre usavam equações diferenciais (conteúdo visto em MAT0221).

11 Planos para continuação na área

Pretendo continuar os estudos na área, fazendo pós-graduação em bioinformática. Acredito que as oportunidades que posso ter num programa de pós-graduação interdisciplinar podem ser bem interessantes academicamente, tanto em computação quanto em biologia. Aproveitando as experiências proporcionadas pelos 2 anos e meio de iniciação científica e 6 anos (razoavelmente intensos) de graduação[83], pretendo iniciar o doutorado direto já no primeiro semestre de 2013 [84].

[81] acho que essa forma de conduzir o curso deveria servir de exemplo, pois fiz várias disciplinas do IME em que entreguei listas de exercícios que muitas vezes nem voltavam corrigidas - ou, quando voltavam, a correção não indicava quais os motivos dos erros e quais seriam as soluções corretas -, o que não colaborou muito para o meu aprendizado.

[82] sistema dinâmico é o que possui uma regra determinística para determinar quais os estados futuros a partir do estado atual[149].

[83] e também para tentar "compensar" todo o tempo gasto até o fim da graduação, que totaliza (no mínimo) 8 anos se for considerado o tempo de preparação para a aprovação no vestibular (2 anos).

[84] se tudo der certo ...