UNIVERSIDADE FEDERAL DO PARANÁ

VITOR CEDRAN PIRO

DESENVOLVIMENTO DA FERRAMENTA PARA FINALIZAÇÃO DE MONTAGENS DE GENOMAS *in silico* - FGAP

CURITIBA

2014

VITOR CEDRAN PIRO

DESENVOLVIMENTO DA FERRAMENTA PARA FINALIZAÇÃO DE MONTAGENS DE GENOMAS *in silico* - FGAP

Dissertação apresentada ao Programa de Pós-Graduação em Bioinformática, Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná como requisito parcial para obtenção do grau de Mestre em Bioinformática.

Orientador: Dr. Roberto Tadeu Raittz
Co-Orientador: Dr. Helisson Faoro

CURITIBA
2014

Piro, Vitor Cedran
P671 Desenvolvimento da ferramenta para finalização de montagens de genomas *in silico* - FGAP / Vitor Cedran Piro . - Curitiba, 2014.
105 f.: il., tabs, grafs.

Orientadora: Prof. Dr. Roberto Tadeu Raittz
Co-orientador: Prof. Dr. Helisson Faoro
Dissertação (Mestrado) – Universidade Federal do Paraná, Setor de Educação Profissional e Tecnológica, Curso de Pós-Graduação em Bioinformática.

1. Genoma. 2. Finalização de genoma. 3. Fechamento de gaps. 4. Bioinformática. I. Raitz, Roberto Tadeu. II. Faoro, Helisson. III.Título. IV. Universidade Federal do Paraná.

CDD 575.113

TERMO DE APROVAÇÃO

VITOR CEDRAN PIRO

"Desenvolvimento da ferramenta para finalização de montagens de genomas in silico - FGAP"

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Bioinformática, pelo Programa de Pós-graduação em Bioinformática, Setor de Educação Profissional e Tecnológica, da Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientador: Prof. Dr. Roberto Tadeu Raittz

Coorientador: Prof. Dr. Helisson Faoro

Prof. Dr. Emanuel Maltempi de Souza
Universidade Federal do Paraná

Prof. Dr. Vasco Ariston de carvalho Azevedo
Universidade Federal de Minas Gerais

Curitiba, 11 de abril de 2014

RESUMO

A finalização é a etapa que consome mais tempo e demanda maior esforço em projetos de determinação de sequências genômicas. Diversos métodos computacionais (*in silico*) foram propostos com o objetivo de resolver problemas de correção de erros, ordenação de *contigs*, fechamento de *gaps*, validação de montagem e refinamento. A etapa de fechamento de *gaps* envolve a identificação de sequências desconhecidas entre *contigs* adjacentes. A presença destes *gaps* ocorre pela falta de *reads* no conjunto de dados sequenciados, necessitando de dados adicionais para serem resolvidos, ou pela incapacidade dos programas de montagem de resolver regiões de repetição ou baixa cobertura, casos em que o fechamento de *gaps in silico* pode ser aplicado. Apresentamos um novo programa para fechamento de *gaps* em sequências de genomas recém-montados, o FGAP, que utiliza dados obtidos de diferentes programas de montagem ou diferentes tecnologias de sequenciamento. A ferramenta busca por sequências que sobreponham finais de *contigs* de *scaffolds* propostos para descobrir a sequência dos *gaps*. O FGAP foi testado em casos controlados e em casos reais, demonstrando capacidade de melhorar montagens apenas reutilizando dados previamente obtidos. Ele também foi comparado com programas desenvolvidos para o mesmo fim, mostrando performance superior e menor tempo de execução. Diversos testes em sequências de organismos procariotos foram realizados e verificados através de validações locais com sequências de referência. A taxa de acerto manteve-se acima de 93%. Análises de métricas globais da montagem após o fechamento comprovam a eficácia do método. O programa é altamente flexível, aceita diversos conjuntos de dados e suporta leituras longas da terceira geração de sequenciamento. Ele não depende de *reads* pareados e produz arquivos de saída detalhados e intuitivos. O FGAP pode ser executado localmente ou via web e está disponível em: www.bioinfo.ufpr.br/fgap

Palavras-chave: genoma, finalização de genoma, fechamento de gaps, bioinformática.

ABSTRACT

Finishing is the most time consuming and labor intensive step in genome sequencing projects. Several computational methods (*in silico*) have been proposed aiming to solve finishing problems such as error correction, contig ordering, gap filling, assembly validation and refining. Gap filling or gap closing involves the identification of sequences to fill in gaps between adjacent contigs. The presence of such gaps may be due to the absence of the respective reads in the database, which requires new sequencing data, or to inherent inability of the assembler to deal with repeated and low coverage regions, which can be improved by *in silico* gap filling approaches. We present a new tool for gap filling in newly assembled genome sequences, named FGAP, to make use of assemblies obtained with different assemblers or from different sequencing platforms. The tool searches for sequences overlapping contig ends of a proposed scaffold aiming to discover the gap sequences. FGAP was tested in controlled and real cases, showing the capacity for improving assemblies reusing data already obtained. It was also compared against other softwares with the same purpose, showing a superior performance and shorter execution time. Several tests were made using prokaryotic genome sequences and validated locally through reference sequences. The accuracy rate was above 93%. Global metrics obtained after gap closing demonstrate the effectiveness of the method. The software is highly flexible, supports many datasets and long reads from third generation sequencing. FGAP does not depends on paired reads and generates detailed and intuitive output files. FGAP can run locally or through the web and is available at: www.bioinfo.ufpr.br/fgap

Keywords: genome, genome finishing, gap closing, bioinformatics.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

BLASTN *Nucleotide Blast*

DB *Database*

dATP, dGTP, dCTP e dTTP *Deoxinucleotídeos*

ddATP, ddGTP, ddCTP ou ddTTP *Didesoxinucleotídeos*

DDBJ *DNA Data Bank of Japan*

DNA Ácido desoxirribonucleico

ENA *European Nucleotide Archive*

EMBL *European Molecular Biology Laboratory*

GAGE *Genome Assembly Gold-standard Evaluations*

HSP *High Scoring Pairs*

MATLAB *Matrix Laboratory*

NCBI *National Center for Biotechnology Information*

OLC *Overlap-layout-consensus*

pb Pares de base

PCR *Polymerase Chain Reaction*

SMRT *Single Molecule Real Time*

SRA *Short Read Archive*

WGS *Whole-genome shotgun*

SUMÁRIO

1 INTRODUÇÃO

O sequenciamento de DNA e a obtenção de sequências de genomas são considerados grandes marcos da ciência moderna. Desde o descobrimento da estrutura da dupla hélice de DNA por Watson e Crick (WATSON; CRICK, 1953), o método de sequenciamento Sanger (SANGER; NICKLEN; COULSON, 1977), a publicação da primeira sequência de genoma (FLEISCHMANN et al., 1995) e o sequenciamento do genoma humano (LANDER et al., 2001) (VENTER et al., 2001) o campo da genômica vem sendo marcado por enormes desafios, grandes descobertas e realizações.

Obter e estudar o genoma de um organismo, possibilita um melhor entendimento sobre sua vida, sobre processos e funções biológicas, hereditariedade e evolução. Isto é possível devido aos avanços no sequenciamento de DNA, etapa que tem por finalidade determinar a sequência de nucleotídeos que formam o genoma de um organismo.

Esta determinação figura como a primeira etapa do estudo do genoma onde é realizada a obtenção, montagem, verificação e disponibilização do código, tornando possível diversos estudos posteriores. Estes estudos serão baseados na sequência genômica, sendo uma tarefa de extrema importância que requer alto grau de precisão.

A bioinformática, ciência multidisciplinar que surgiu diante do crescente número de dados biológicos obtidos do sequenciamento em larga escala e a necessidade para análise computacional e armazenamento, junto com extraordinários avanços na biologia molecular tem acelerado o passo de descoberta e evolução neste campo. Novas tecnologias de sequenciamento e novas técnicas computacionais têm melhorado rapidamente a análise e obtenção de sequências de genomas. Porém ainda é necessário um grande esforço para melhorar este processo, automatizando etapas para obter genomas com melhor qualidade, mais completos, com menor custo e em menos tempo.

O processo de determinação de uma sequência genômica é usualmente subdividido em três etapas: sequenciamento, montagem e finalização (FIGURA 1).

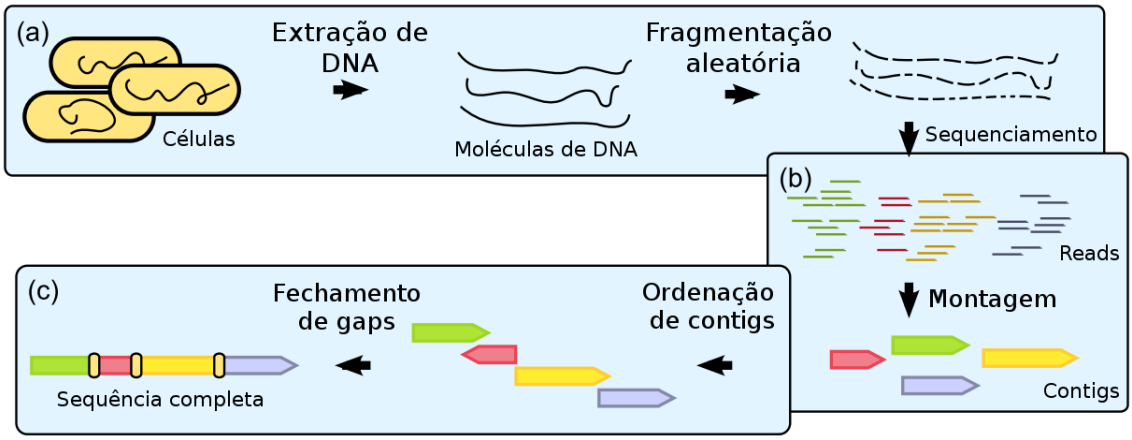


FIGURA 1: FLUXOGRAMA DO PROCESSO DE DETERMINAÇÃO DA SEQUÊNCIA GENÔMICA DE UM ORGANISMO

(a) Extração, purificação e sequenciamento de DNA (b) Montagem da sequência do genoma (c) Finalização de montagem
FONTE: Adaptada (HUSEMANN, 2011)

1.1 SEQUENCIAMENTO DE DNA

1.1.1 Primeira geração - Método Sanger

Os primeiros esforços bem-sucedidos para o sequenciamento de DNA foram realizados no laboratório do Dr. Frederick Sanger nos anos 70 (SANGER; NICKLEN; COULSON, 1977) e, paralelamente, por Maxam e Gilbert (MAXAM; GILBERT, 1977). O método de terminadores de cadeia de Sanger prevaleceu como primeira tecnologia de sequenciamento por ser menos complexo e mais adaptável para geração de dados em larga escala, comparado com os métodos de sequêncimento químico de Maxam e Gilbert.

O método Sanger visa obter a sequência de DNA utilizando a DNA Polimerase (enzima responsável por adicionar nucleotídeos a fita de DNA no processo de duplicação) durante uma reação ao incorporar nucleotídeos com pequenas modificações químicas no grupo 3'-hidroxila e no grupo 2'-hidroxila chamados de didesoxinucleotídeos. Esses nucleotídeos alterados são então incorporados aleatoriamente na extremidade 3' da cadeia polinucleotídica, causando uma terminação antecipada no enlongamento da cadeia, impedindo a adição de outros nucleotídeos.

Pelo método clássico, são necessárias quatro reações para obter a sequên-

cia de DNA. Em cada reação, são adicionados os deoxinucleotídeos normais (dATP, dGTP, dCTP e dTTP) e em cada uma delas um conjunto de didesoxinucleotídeos (ddATP, ddGTP, ddCTP ou ddTTP) em uma proporção menor do que os deoxinucleotídeos. Desta maneira, quando ocorrer a replicação de DNA, será gerada uma coleção de fragmentos com terminações específicas, sendo que cada reação terminará no seu respectivo didesoxinucleotídeo. Estes fragmentos de diversos tamanhos terão início na extremidade 5', determinada por um primer, e terão comprimentos diferentes nas extremidades 3'. O comprimento dos fragmentos determina o nucleotídeo da fita sequenciada (WATSON et al., 2006).

Inicialmente o método era realizado de maneira manual, com as reações ocorrendo separadamente e visualizada em géis de poliacrilamida em linhas separadas. As linhas eram lidas a olho nu, de baixo para cima (5' > 3'), formando assim a sequência do DNA, como mostrado na FIGURA 2.

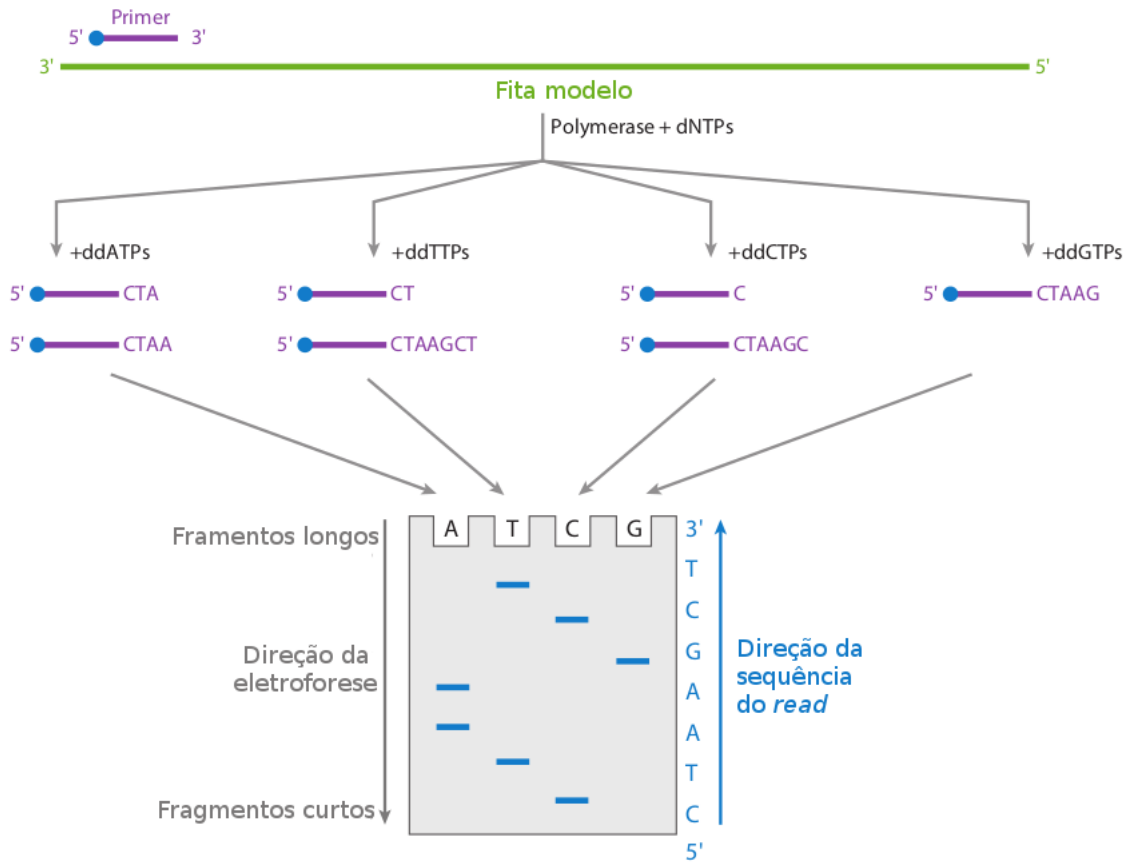


FIGURA 2: SEQUENCIAMENTO SANGER

FONTE: Adaptada (MARDIS, 2011)

Até meados dos anos 80 a técnica sofreu diversos avanços, aumentando a ve-

locidade e a quantidade de dados gerados com diversas etapas automatizadas (MARDIS, 2011). Entretanto, ainda não era eficaz o suficiente para gerar dados em larga escala. Em 1986 foi introduzido o sequenciamento fluorescente de DNA que substituiu a marcação radioativa utilizada anteriormente, diminuindo drasticamente o trabalho manual e a taxa de erros. Em 1999 foi lançado o sequenciamento por capilares, acelerando ainda mais o processo de sequenciamento por produzir mais dados em menos tempo e ter um método de utilização facilitado. Esta tecnologia foi utilizada para obtenção das sequências do genoma de organismos modelo, como o das primeiras bactérias (FLEISCHMANN et al., 1995) (BLATTNER et al., 1997), do rato (GIBBS; WEINSTOCK; METZKER, 2004) e do ser humano (LANDER et al., 2001) (VENTER et al., 2001). As técnicas e métodos mostradas até então são chamadas de primeira geração de sequenciamento ou método Sanger.

1.1.2 Segunda geração - Leituras curtas

A partir de 2005 houve uma grande mudança no paradigma de sequenciamento. Novos aparelhos surgiram com técnicas diferentes do método Sanger, que foi a base de todos os equipamentos de sequenciamento de DNA por quase 30 anos. A chamada segunda geração de sequenciamento, ou geração de leituras curtas, difere em diversos aspectos da primeira geração e se destaca pela quantidade de dados gerados por corrida (FIGURA 3) assim como pela drástica redução de custos (FIGURA 4), fatores que impulsionaram os estudos genômicos (MARDIS, 2011). Devido a características do processo de sequenciamento, esta geração trouxe o encurtamento do comprimento das leituras, o número de nucleotídeos obtido para cada fragmento sequenciado, também chamados de *reads*, dificultando o processo de montagem, apesar da grande quantidade de dados gerados.

Além de serem reduzidos, os passos preparatórios iniciais para realizar o sequenciamento são mais simples nesta geração, quando comparado aos antigos métodos. Em vez de clonagem bacteriana seguida pelo isolamento do DNA, estes equipamentos iniciam o processo de sequenciamento com a preparação de uma biblioteca utilizando DNA sintético (adaptadores) que serão ligadas aos finais dos fragmentos a serem sequenciados.

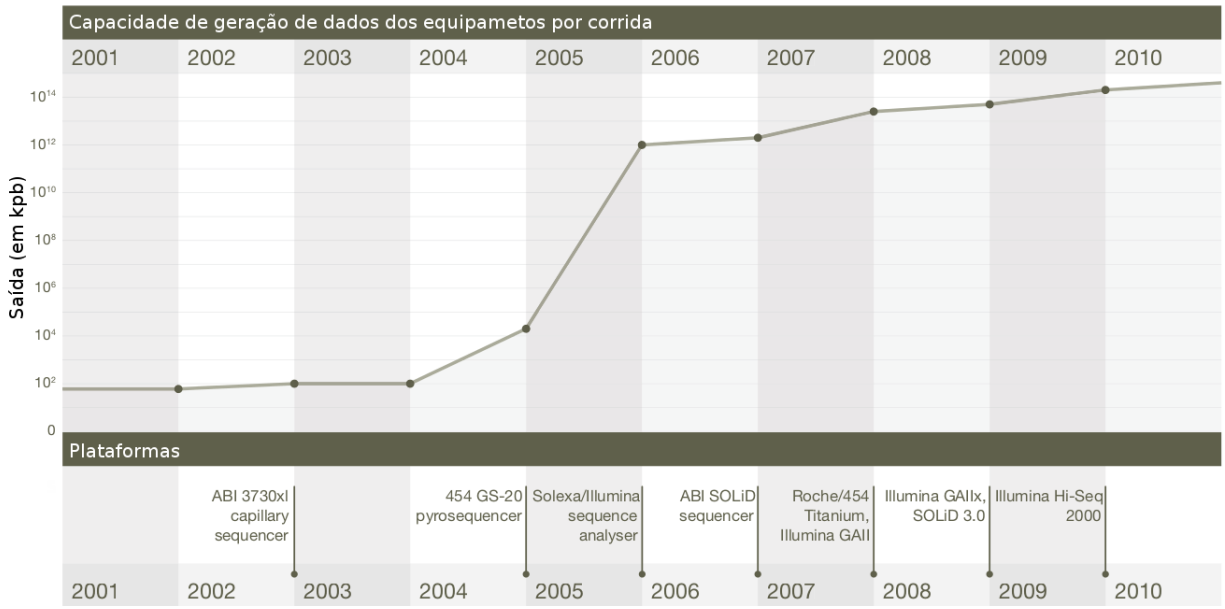


FIGURA 3: EVOLUÇÃO DO SEQUENCIAMENTO DE DNA

Evolução em escala logarítmica da capacidade de saída de dados (em kpb) dos equipamentos por corrida. O contador é iniciado na publicação do genoma humano, em 2001. Também são mostradas referências temporais do lançamento de alguns equipamentos de sequenciamento.
FONTE: Adaptada (MARDIS, 2011)

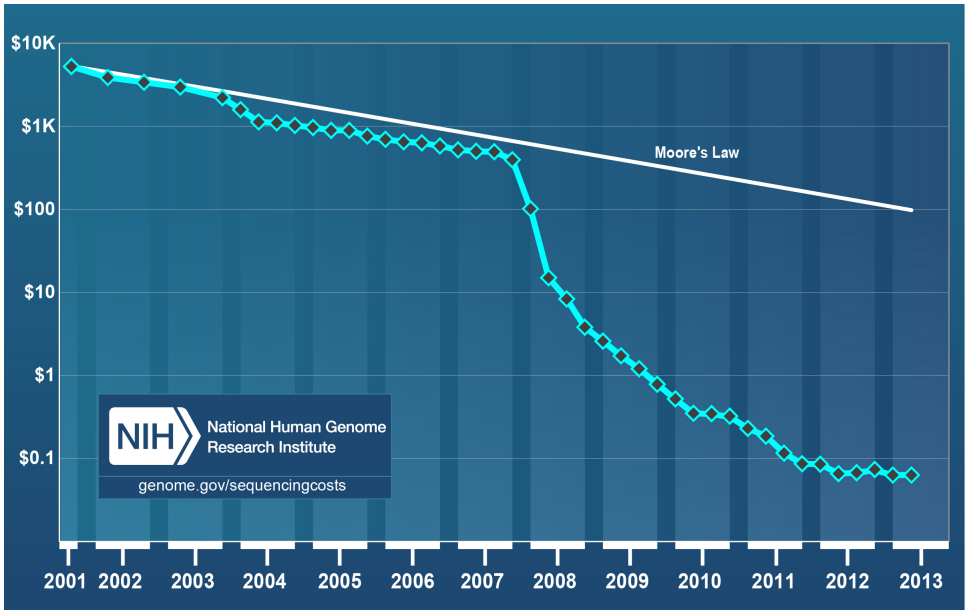


FIGURA 4: EVOLUÇÃO DOS CUSTOS DE SEQUENCIAMENTO

Evolução dos custos para obtenção de 1Mpb de DNA em dólares (*k* = mil). O contador é iniciado na publicação do genoma humano, em 2001. É feita uma comparação com a Lei de Moore (*Moore's Law*), que é baseada na redução dos custos na produção de hardwares, onde a produção que conseguir reduzir os custos conforme a lei é considerada extremamente bem-sucedida. Nota-se uma queda brusca no início do ano de 2008 ocasionada pela migração de usuários da primeira geração para a segunda geração de sequenciamento.
FONTE: (WETTERSTRAND, 2014)

TABELA 1: TECNOLOGIAS DE SEQUENCIAMENTO

	Química	Tamanho/*read*	Tempo/corrida	Gpb/corrida
Equipamentos *High-end*				
454 GS FLX+ (R)	Pirosequenciamento	1000pb	23h	0.7
HiSeq 2500 (I)	Terminadores reversíveis	2 x 125pb	6 dias	1000
5500xl SOLiD (L)	Ligação	75pb + 35pb	8 dias	150
Equipamentos *Bench-top*				
454 GS Junior (R)	Pirosequenciamento	400pb	10h	0.035
Ion PGM (L)	Detecção de próton	400pb	7h	2
Ion Proton (L)	Detecção de próton	200pb	4h	10
MiSeq (I)	Terminadores reversíveis	2 x 250pb	39h	8.5

(R): Roche; (I): Illumina; (L): Life Technologies. Dados de saída dos equipamentos atualizados em Maio/2014.
FONTE: Adaptado (LOMAN et al., 2012)

Em todas as plataformas, os fragmentos precisam ser amplificados em uma superfície sólida por PCR, produzindo diversas cópias de cada biblioteca de fragmentos. A amplificação é necessária para produção de DNA suficiente, aumentando o sinal fluorescente dos nucleotídeos modificados, pois a identificação das bases é feita através de um dispositivo óptico. É importante ressaltar que esta etapa introduz uma taxa de erro ao processo, pois a DNA Polimerase não é completamente precisa.

Os equipamentos desta geração de sequenciamento funcionam através de uma série de reações: adição de nucleotídeos modificados, detecção óptica de nucleotídeos e limpeza química para retirada de marcadores fluorescentes. Os instrumentos realizam estas reações de maneira cíclica, com todos os passos automatizados, produzindo centenas de milhões de *reads* simultaneamente. Cada plataforma difere na técnica de detecção do sinal e na reação de sequenciamento de DNA (METZKER, 2010).

Desde o início da geração, em meados de 2005, diversos equipamentos para sequenciamento de DNA foram lançados, possuindo algumas características semelhantes entre si, porém com diversas diferenças técnicas desde a etapa de preparo de amostras até o formato e quantidade de saída gerada. Estes equipamentos podem ser organizados em 2 grupos distintos para melhor classificação: instrumentos *High-end*, máquinas mais robustas com alta saída de dados e consequentemente maior custo de implementação, destinado a grandes centros de pesquisa, e os instrumentos *Bench-top* de menor custo, que produzem menos dados porém são acessíveis e tem menor tempo de execução (TABELA 1).

Nesta geração foi introduzida a técnica de sequenciamento de *reads* pareados. Ela permite a obtenção de ambas as pontas dos fragmentos sequenciados, aumentando a relação entre os dados de saída, diferentemente dos fragmentos únicos gerados até então, também conhecidos por *single-end*. Com esta técnica, cada *read* gerado terá um outro *read* relativo, com a distância e orientação entre eles conhecida. Os dados gerados podem ser *paired-end* ou *mate-pair* que, apesar de possuírem técnicas distintas tanto na preparação das amostras quanto na etapa de sequenciamento, geram dados semelhantes. As bibliotecas *paired-end* geram *reads* pareados próximos com sobreposição (fragmentos menores que o dobro do tamanho do *read*) até *reads* com tamanho de inserção curto de até 1000pb de distância entre si, geralmente com a orientação reversa (um *read* 3'-5' e o outro 5'-3'). Já as bibliotecas *mate-pair* ou *short/long jump* possuem um tamanho de fragmento maior, gerando *reads* mais distantes entre si, geralmente maiores que 1000pb. Estas bibliotecas de *reads* pareados são extremamente úteis em diversas etapas subsequentes ao sequenciamento, como na fase de montagem e finalização da sequências de genomas.

1.1.3 Terceira geração - Molécula única

Em 2011 iniciou-se a comercialização do primeiro equipamento de terceira geração, também conhecido como sequenciador de molécula única em tempo real (SMRT - *Single Molecule Real Time*). O PacBioRS da Pacific Biosciences (SCHADT; TURNER; KASARSKIS, 2010) iniciou uma nova geração, ao inovar no método de sequenciamento através da identificação das bases em molécula única (RANK et al., 2009). O processo funciona incorporando nucleotídeos marcados na molécula crescente de DNA por uma Polimerase diferenciada. Cada DNA Polimerase é presa a um ZMW (*zero-mode waveguide*) que são compartimentos extremamente pequenos distribuídos em uma placa. Este compartimento permite que seja realizada a identificação ótica quando os nucleotídeos são adicionados na fita crescente de DNA (MARDIS, 2013).

A terceira geração de sequenciamento trouxe novamente o benefício de *reads* longos, ainda maiores que os *reads* de primeira geração. Porém a alta taxa de erro e alto custo ainda são fatores limitantes desta tecnologia que é muito promissora

(TABELA 2). Mesmo em fase inicial, a capacidade para montagem genômica dos dados provenientes desta nova tecnologia já foi demonstrada (CHIN et al., 2013) (KOREN et al., 2012), pois consegue resolver regiões de repetição pequenas e médias, caso tenha cobertura de *reads* suficiente.

TABELA 2: COMPARAÇÃO DAS GERAÇÕES DE SEQUENCIAMENTO

	PRIMEIRA GERAÇÃO	SEGUNDA GERAÇÃO	TERCEIRA GERAÇÃO
Taxa de erro	Baixa (∼2%)	Baixa-Média (∼2-4%)	Alta (∼12%)
Tamanho/*read*	Longo (500-1000pb)	Curto (75-1000pb)	Longo (3000-15000pb)
Tempo/Corrida	Horas	Horas/Dias	Minutos
Produtividade	Baixa	Alta	Média

FONTE: O Autor (2014)

1.2 MONTAGEM DE GENOMAS

A montagem de genomas compreende o processo computacional de unir as leituras sequenciadas de DNA, os *reads*, para obter a sequência genômica completa de um organismo (FIGURA 1 b). Estes fragmentos sequenciados são usualmente gerados através da técnica conhecida por sequenciamento genômico *shotgun* (WGS - *whole-genome shotgun*) onde o genoma é fragmentado em pequenas partes e então submetido ao sequenciamento.

A montagem não é trivial pois os dados gerados no sequenciamento possuem erros de identificação de bases e diversas regiões de repetição que afetam o processo, principalmente quando estas regiões são maiores em extensão do que o tamanho dos *reads* (KINGSFORD; SCHATZ; POP, 2010).

Uma analogia para melhor compreensão do processo de montagem pode ser feita: considere o genoma como o texto de um livro, com cada letra representando um nucleotídeo (FIGURA 5). Pelo processo de sequenciamento WGS, o livro é cortado e lido em pequenos fragmentos horizontais aleatoriamente. Este processo é feito com várias cópias do livro idênticas entre si, salvo por alguns fragmentos perdidos e erros tipográficos. Ao final teremos diversos fragmentos do texto embaralhados e sem ordem. A montagem é o processo de unir estes fragmentos novamente e formar o texto completo do livro, considerando as partes faltantes e as diferenças tipográficas.

A montagem resume-se a unir os *reads* em sequências contíguas, chamadas

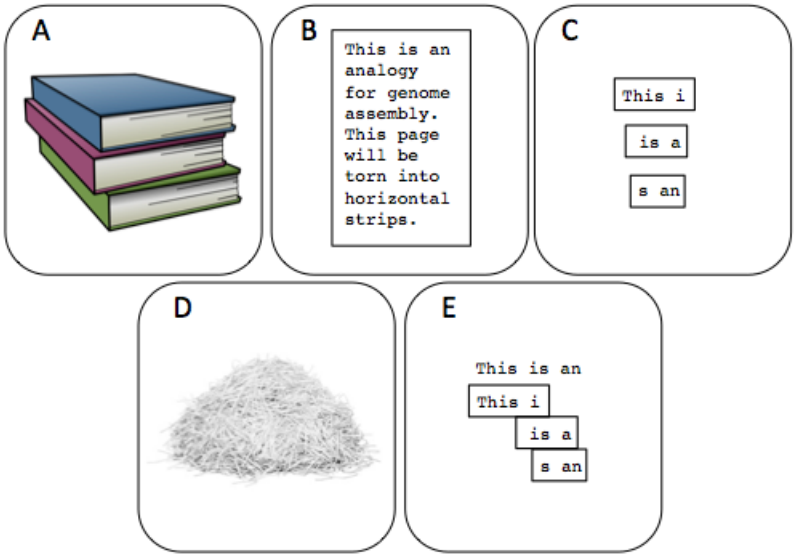


FIGURA 5: ANALOGIA AO PROCESSO DE MONTAGEM

A) Dados sequenciados são redundantes, várias cópias do mesmo livro; B) Exemplo de página do livro; C) Fragmentos extraídos aleatoriamente; D) Conjunto total de fragmentos; E) Processo de sobreposição e geração de consenso
FONTE: Adaptada (TAYLOR, 2012)

de *contigs*, representando o genoma total do organismo. Esta etapa visa a extração de uma sequência consenso após o empilhamento dos *reads* sobrepostos, considerando as variações e erros (FIGURA 6) através da redundância de dados. Fica evidente a vantagem de ter uma grande quantidade de dados, pois quanto mais fragmentos repetidos, maior a chance de ocorrer sobreposição, possibilitando a correção de erros e maior extensão da sequência. Esta medida que considera quantidade de *reads* redundantes para a montagem de genomas é chamada de cobertura. Na analogia do livro, se 10 livros fossem picotados e montados, teríamos uma cobertura de 10 vezes, ou seja, cada letra do livro estaria representada 10 vezes no conjunto total.

Para realizar a montagem de genomas, são aplicadas algumas técnicas computacionais que lidam com as características específicas dos *reads* de sequenciamento e sua quantidade. Existem duas abordagem principais: a montagem por referência, que utiliza a sequência de um genoma conhecido para auxiliar o processo. Esta técnica demanda poucos recursos computacionais porém tem seu uso restrito e é pouco utilizada. Para realizar a montagem por referência é necessário uma sequência de genoma já obtida de um organismo muito próximo e, mesmo assim, possui limitações de montagem nas regiões diferentes e na identificação de variações entre os organismos. A outra abordagem é a montagem *de novo*, feita sem auxílios, utili-

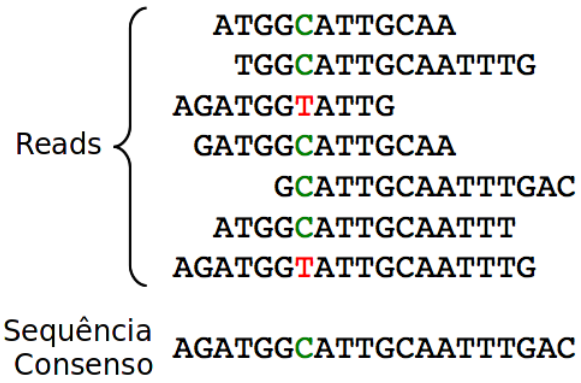


FIGURA 6: SEQUÊNCIA CONSENSO

Empilhamento de *reads* gerando uma sequência consenso. Bases em vermelho mostram erros no processo de sequenciamento. A geração da sequência consenso corrige as falhas, devido a redundância de dados.
FONTE: Adaptada (TAYLOR, 2012)

zando apenas os *reads* do próprio organismo sequenciado. A abordagem *de novo* é a ideal para montagem de sequências de genomas, apesar de seu alto custo computacional e suas limitações (POP, 2009). Ela é amplamente utilizada e é usualmente aplicada através de três paradigmas (NAGARAJAN; POP, 2013): algoritmo guloso, método OLD e grafos De Bruijn.

1.2.1 Algoritmo Guloso

O algorimo guloso ou ganancioso foi um dos primeiros a ser aplicado para o problema de montagem. O método busca unir *reads* com a melhor sobreposição possível, ou seja, une dois *reads* quando o sufixo de um é igual ao prefixo de outro. Repete o processo até não conseguir unir mais fragmentos. Toma apenas decisões locais, sem considerar a posição global do *read* em relação a sequência completa do genoma. Com isso pode gerar regiões erradas, como nos casos das repetições, onde nem sempre o *read* que tiver melhor sobreposição será o *read* correto a ser concatenado. Além disto, é computacionalmente inviável para a quantidade de *reads* gerados pelas tecnologias com alta saída de dados. É o algoritmo mais simples e intuitivo para o problema de montagem.

1.2.2 Método OLC

Do inglês *Overlap-layout-consensus* - sobreposição, leiaute e consenso. O nome do método representa as três etapas em que a técnica é baseada (POP, 2009).

A etapa de sobreposição é similar ao algoritmo guloso: é feito um alinhamento para identificar pares de *reads* sobrepostos. Como o alinhamento de todos os *reads* contra todos é uma tarefa custosa computacionalmente, métodos alternativos são utilizados, como o uso de sementes (frações dos *reads*, também chamadas de k-mers) que limitam o escopo de comparações, gerando um índice de quais *reads* devem ser comparados. A partir dos dados de pares de *reads* sobrepostos é possível gerar um grafo de sobreposição (FIGURA 7) contendo cada *read* como um nó e uma aresta conectando os nós quando existe sobreposição entre eles.

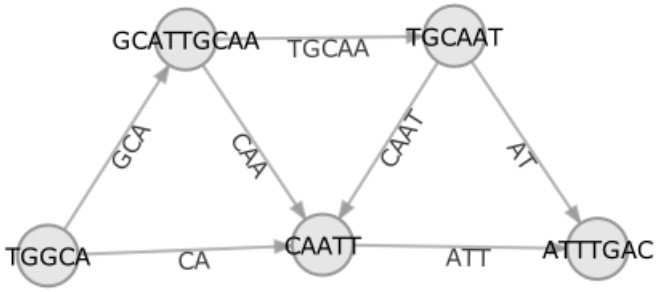


FIGURA 7: GRAFO OLC

Grafo de montagem de genomas pelo método OLC, considerando sobreposições de pelo menos 2pb. Nós são *reads* completos e arestas conectam *reads* que possuem sobreposição. Os *reads*, que geralmente são maiores, foram encurtados para melhor visualização e entendimento do problema. A sequência consenso gerada pelo grafo é "TGGCATTGCAATTTGCA"
FONTE: (TAYLOR, 2012)

A etapa de leiaute consiste em analisar o grafo e encontrar um único caminho que visite todos os nós, o caminho Hamiltoniano, consequentemente utilizando todos os *reads*. A identificação deste caminho nem sempre é possível devido a sua alta complexidade, grande quantidade de nós, regiões de repetição e erros que afetam a obtenção do caminho completo. Eventualmente, são identificados subgrafos a partir do grafo principal, onde não há bifurcações causadas por regiões de repetição. São gerados então, a partir destes subgrafos, diversos *contigs*, resultando em uma sequência genômica fragmentada.

A etapa de consenso determina a sequência de DNA a partir da disposição

dos *reads* no caminho escolhido no grafo/subgrafos, gerando as sequências finais da montagem.

Diferentes programas utilizam limites de sobreposição e técnicas de alinhamento diferentes. Montagens mais confiáveis precisam ter maior sobreposição no alinhamento entre os *reads*, porém perdem contiguidade, gerando *contigs* menores e sequências de genoma mais fragmentadas. EDENA (HERNANDEZ et al., 2008) e SGA (SIMPSON; DURBIN, 2012) são exemplos de programas que aplicam o paradigma OLC, adicionando recursos e otimizando as etapas de sobreposição e leiaute. O SGA tem recebido bastante atenção ultimamente por utilizar uma técnica chamada *string graph*, que reduz drasticamente o uso de memória e tempo de execução, tornando possível o uso do paradigma OLC na montagem de genomas maiores e com grande quantidade de *reads*.

1.2.3 Método de grafos De Bruijn

A técnica para montagem de genomas mais comum e amplamente utilizada atualmente é baseada em grafos De Bruijn (NAGARAJAN; POP, 2013). Ela tem melhor desempenho por ser menos custosa que o método OLC, conseguindo lidar com grande quantidade de dados e sendo mais sensível a regiões de repetição.

Neste método o conjunto de *reads* sequenciados é reduzido em subsequências, chamadas de k-mers (sendo k a constante que definirá o tamanho da subsequência). A partir deste conjunto é gerado um grafo tendo as arestas como as subsequências únicas (k-mers) e os nós como os prefixos e sufixos das subsequências. Desta maneira, nós que possuem sobreposição exata de tamanho $k-2$ ficam conectados (FIGURA 8).

Neste caso, o problema de montagem se resume a encontrar um caminho Euleriano no grafo gerado a partir dos k-mers (PEVZNER; TANG; WATERMAN, 2001). Este método busca um caminho que passe por todas as arestas exatamente uma vez. Diferentemente do caminho Hamiltoniano (que visita cada nó uma vez, utilizado no grafos OLC), existem algoritmos mais eficientes para encontrar o caminho Euleriano. Porém, devido à ambiguidade das regiões de repetição, diversos caminhos Eulerianos existem, sendo necessário descobrir qual é o caminho correto para representar corre-

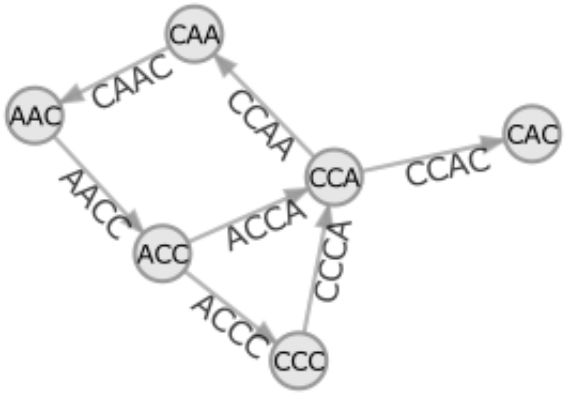


FIGURA 8: GRAFO DE BRUIJN

Um grafo De Bruijn com $k = 4$. Os k-mers, que geralmente são maiores, foram encurtados para melhor visualização e entendimento do problema. A sequência consenso gerada pelo grafo é "ACCCAACCAC".
FONTE: (TAYLOR, 2012)

tamente a sequência do genoma. Muitas vezes esta informação não está contida nos dados, como em repetições longas, fazendo com que a montagem fique fragmentada em *contigs*.

Apesar de ser uma técnica poderosa e amplamente utilizada, ela é muito influenciada por erros de sequenciamento, necessitando de etapas de pré-processamento dos *reads*. A técnica tende a ter seu uso diminuído com o constante aumento do tamanho dos *reads* trazidos pelas novas tecnologias de terceira geração, sendo substituída pelas técnicas anteriormente descritas.

1.2.4 Problemas na montagem de genomas

As técnicas para montagem da sequência genômica a partir de *reads* de sequenciamento curtos tem diversas limitações que impedem a reconstrução correta e completa da sequência dos genomas. Estas limitações não são exclusivamente causadas pela ineficiência dos atuais métodos mas são altamente influênciadas pelas características dos dados. Quanto menor o tamanho do *read*, mais complexo será descobrir do caminho correto no grafo gerado, e mais difícil a resolução de regiões de repetição (KINGSFORD; SCHATZ; POP, 2010).

Estas regiões de repetição são o principal fator de limitação nas montagens genômicas. Nos humanos elas representam mais de 50% do genoma e em bacté-

rias podem chegar até 40% (TREANGEN; SALZBERG, 2012). Os paradigmas para montagem de genoma abstraem os dados em grafos, e quando regiões de repetição ocorrem, o grafo gera caminhos ambíguos, impossibilitando a montagem correta (FIGURA 9).

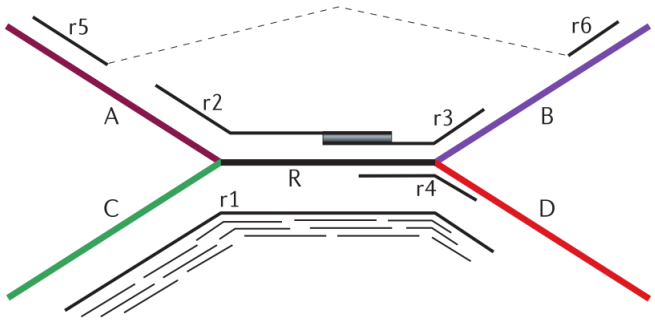


FIGURA 9: REGIÃO DE REPETIÇÃO

Grafo demonstrando região ambígua de repetição em montagem com possíveis soluções para resolução. Em preto, o traço R representa a região de repetição e os traços coloridos A-B-C-D as regiões a montante e a jusante da repetição. O traço r1 representa um *read* longo que evidenciaria a resolução da repetição como ARB e CRD. Os *reads* r2, r3 e r4 não seriam suficientes para resolver a ambiguidade da região pois seria possível gerar tanto a sequência ARB (r2 e r3) quanto a sequência ARD (r2 e r4). Os traços r5 e r6 representam *reads* pareados (*mate-pair*) que resolveriam a região como ARB e CRD.
FONTE: (NAGARAJAN; POP, 2013)

Algumas técnicas foram criadas para tentar solucionar ou diminuir problemas de ambiguidades geradas pelas regiões de repetição. Nos grafos de Bruijn, é aplicada a técnica chamada *Super-Path* (PEVZNER; TANG; WATERMAN, 2001), onde a sequência dos *reads* inteiros é reconsiderada quando há ambiguidade no grafo. Esta técnica consegue resolver repetições curtas, geralmente geradas pela fragmentação em k-mers. Para repetições longas é utilizado, quando disponível, informações de *reads* pareados como mostrado na FIGURA 9 (WETZEL; KINGSFORD; POP, 2011). Com os recentes avanços das tecnologias de terceira geração, os *reads* longos podem resolver ambiguidades caso sejam maiores que a região de repetição (BASHIR et al., 2012).

1.3 FINALIZAÇÃO DE MONTAGEM DE GENOMAS

A finalização de montagem de genomas figura como uma etapa crucial para obter a sequência completa de um genoma, dado que as técnicas de montagem e limi-

tações do sequenciamento usualmente não possibilitam a obtenção direta da sequência. A finalização é um processo iterativo que consiste em ordenar os *contigs* gerados na montagem, identificar regiões entre *contigs* e outras que não foram montadas, assim como validar a sequência obtida (FIGURA 1 c).

O processo de finalização demanda maior tempo e custo se comparado com as outras etapas do processo de montagem (NAGARAJAN et al., 2010). Porém, análises posteriores de sequências genômicas finalizados produzem resultados mais confiáveis e refinados. Além disto, a etapa de finalização tende a melhorar a qualidade das sequências, sendo por si só uma grande vantagem. Ainda há um debate em até que ponto é válido realizar a finalização devido ao seu alto custo e esforço demandado, porém é inegável que o processo de finalização melhora substancialmente a qualidade das sequências montadas dos genomas (RICKER; QIAN; FULTHORPE, 2012).

1.3.1 Finalização *in vitro*

A finalização *in vitro* é realizada em laboratório e busca a identificação de sequências ou partes do genoma através de sequenciamentos específicos e guiados. A estratégia padrão deste processo é a criação de primers de oligonucleotídeos localizados nas pontas de *contigs*, a amplificação por PCR destes fragmentos e sequenciamentos curtos, usualmente pelo método Sanger, gerando sequências que podem ampliar *contigs*, cobrir regiões desconhecidas ou fazer a ligação entre *contigs*. A finalização *in vitro* é eficiente porém demanda tempo e trabalho manual, além de ser custosa.

1.3.2 Finalização *in silico*

Com os avanços das tecnologias de sequenciamento e a facilidade de obtenção de grandes quantidades de dados, diversas abordagens para finalização *in silico* surgiram. Elas tem como objetivo melhorar a qualidade das sequências genômicas montadas sem auxílio laboratorial, extraindo informações a partir dos dados redundantes de sequenciamento, reduzindo custos e acelerando o processo de obtenção

de sequências de genoma com melhor qualidade (CHAIN et al., 2009).

O processo de *scaffolding* consiste em ordenar os *contigs*, ou seja, determinar a orientação correta entre eles (POP, 2009). Este processo gera um conjunto de sequências chamados de *scaffolds*, grupos de *contigs* com orientação conhecida porém com a sequência entre eles desconhecida. Esta sequência desconhecida entre *contigs* é chamada de lacuna, ou *gap*. A forma mais comum de gerar *scaffolds* é através de *reads* pareados. Dois *contigs* podem ser considerados adjacentes caso um *read* pareado alinhe em um *contig* e o seu par em outro. Na prática, apenas um par indicando a relação entre dois *contigs* é uma evidência fraca e normalmente são necessários mais pares para garantir uma união real. Esta técnica também é utilizada para definir a orientação relativa entre os *contigs*. Muitos programas de montagem de genomas já possuem um módulo de *scaffolding*. Existem outros programas que realizam apenas esta etapa como o SSPACE (BOETZER et al., 2011), SOPRA (DAYARIAN; MICHAEL; SENGUPTA, 2010), SCARPA (DONMEZ; BRUDNO, 2013), Bambus (POP; KOSACK; SALZBERG, 2004), Opera (GAO; SUNG; NAGARAJAN, 2011), entre outros. O *scaffolding* também pode ser realizado através de uma ou mais sequências de genoma de referência, alinhando os *contigs* contra este conjunto e inferindo a sua ordem. ABACAS (ASSEFA et al., 2009), CONTIGuator (GALARDINI et al., 2011), MAUVE - MCM (RISSMAN et al., 2009) são alguns dos programas que realizam *scaffolding* via organismos de referência.

A etapa de fechamento de *gaps* busca identificar bases não conhecidas na montagem genômica. Estas regiões são provenientes, além do processo de *scaffolding*, da baixa cobertura de *reads*, dados incompletos e erros na etapa de montagem. O uso de *reads* pareados é também muito utilizado nesta etapa de finalização. Programas como GapFiller (BOETZER; PIROVANO, 2012), GapCloser (LUO et al., 2012), IMAGE (TSAI; OTTO; BERRIMAN, 2010), CloG (YANG et al., 2011) e FinIS (GAO; BERTRAND; NAGARAJAN, 2012) utilizam técnicas similares para fechamento de *gaps*. Através do mapeamento dos *reads* pareados nos finais dos *contigs*, os programas buscam identificar o conjunto de pares que pertencem a região do *gap*. É feita então uma montagem local destes dados, utilizando esta montagem para fazer a extensão dos finais dos *contigs*, até que o *gap* seja fechado.

A identificação de erros é essencial tanto na etapa de montagem como na

etapa de finalização para que análises posteriores do genoma sejam feitas corretamente. Os erros podem ser detectados por inconsistência nos dados montados. Regiões de repetição quebradas, regiões de coberturas variáveis, entre outros indícios podem ser identificados através de análises estatísticas e uso de k-mers (KIM; LIAO; TOMB, 2001). Bibliotecas *mate-pair* são utilizadas para descobrir erros longos de montagem, utilizando as informações da orientação dos pares e da distância entre eles. Através do mapeamento destes pares na montagem, são identificados desvios nos valores de distância e orientação, geralmente por diversos pares, podendo ser inferido um erro de montagem ou *scaffolding*.

1.4 BANCOS DE DADOS PÚBLICOS

Os bancos de dados públicos armazenam e diponibilizam dados biológicos. O NCBI, um dos pioneiros no armazenamento e disponibilização de informações (COORDINATORS, 2013), possui sequências de DNA obtidas por toda a comunidade científica mundial que colabora para a criação e ampliação do banco. Este banco faz parte de uma cooperação internacional entre 3 grandes centros que garantem que todas as sequências depositadas em cada um dos bancos esteja disponível em todos, dando cobertura mundial ao repositório. Os bancos de dados unificados que fazem parte da colaboração são: ENA, EMBL, DDBJ e o *GenBank* do NCBI.

1.5 FORMATOS DE ARQUIVOS

O arquivo FASTA (PEARSON; LIPMAN, 1988) é amplamente utilizado como padrão para armazenamento de sequências. Ele é um arquivo simples que permite uma fácil manipulação tanto para usuários quanto para algoritmos. O sinal de maior ($>$) indica o início do cabeçalho da sequência. A sequência é representada pelas linhas abaixo do cabeçalho (FIGURA 10). O arquivo FASTA pode ter diversas sequências e cabeçalhos.

O FASTQ (COCK et al., 2010) é um padrão de arquivo utilizado para armazenar sequências biológicas com seus valores de qualidade. É a saída padrão de diversos equipamentos de sequenciamento que produzem, além da sequência, um

```
>gi|161508266|ref|NC_010079.1| Staphylococcus aureus subsp. aureus USA3
ACTACTGCTCAATTTTTTTACTTTTATCGATTAAAGATAGAAATACACGATGCGAGCAATCAAATTTCAT
AACATCACCATGAGTTTGGTCCGAAGCATGAGTGTTTACAATGTTCGAACACCTTATACAGTTCTTATAC
ATACTTTATAAATTATTTCCCAAACTGTTTTGATACACTCACTAACAGATACTCTATAGAAGGAAAAGTT
ATCCACTTATGCACATTTATAGTTTTCAGAATTGTGGATAATTAGAAATTACACACAAAGTTATACTATT
```

FIGURA 10: ARQUIVO FASTA

FONTE: O autor (2014)

valor de qualidade. Eles tem o objetivo de controlar a precisão com que a base foi identificada pelo sequenciador. O arquivo utiliza o símbolo arroba (@) para demarcar o início de um cabeçalho seguido de sua sequência na linha subsequente. O símbolo de soma (+) é utilizado para identificar o cabeçalho de início de qualidade (que é geralmente o mesmo do cabeçalho da sequência), tendo os valores de qualidade na linha abaixo (FIGURA 11). Os valores de qualidade estão na escala PHRED (EWING et al., 1998) que indica qual a probabilidade de uma base estar incorreta. As qualidades são codificados com caracteres ASCII para ter sua representação compactada. Alguns equipamentos utilizam intervalos de caracteres ASCII distintos para representar a escala PHRED.

```
@SRR057662.133689 EY810BK02HEOCN length=289
TCAGCGACGTTCAGGGAGGTTTCAACCACACCTTTGGCTACATCGGAGTTACGAATCA
+SRR057662.133689 EY810BK02HEOCN length=289
<<<;;<<;<A;:<CA1;@:CA1<A<B<5<;>7A?-<5<:8<8<<A;;;A<<;;A<<;<
@SRR057662.133690 EY810BK02HOLWB length=275
TCAGACCGCCGGTCGAAGGTAACTCTTTAGCTGCCAGAATCGCCCCTGCGGGTGGCGT
+SRR057662.133690 EY810BK02HOLWB length=275
<<<<<@;<A;A;;<;A=A<<B=<<<C@/<<:;<A<<<>7:::B@4&86<C@/<=6;;A
```

FIGURA 11: ARQUIVO FASTQ

FONTE: O autor (2014)

1.6 PROGRAMAS

1.6.1 BLAST

O BLAST (*Basic Local Alignment Search Tool*) é um programa de busca e comparação de sequências de DNA e aminoácidos (ALTSCHUL et al., 1990). O programa realiza uma busca local entre sequências, ou seja, procura por regiões com alto grau de similaridade, utilizando matrizes de substituição (BLOSUM, PAM, etc).

Estas matrizes são geradas com base na probabilidade de mutações no genoma no processo evolutivo, dando peso distintos para bases diferentes alinhadas entre duas sequências. O BLAST é um dos programas de bioinformática mais utilizados pela comunidade científica para alinhamento e busca de sequências.

O algoritmo é baseado no conceito de que quanto mais segmentos similares existirem entre duas sequências e quanto maiores eles forem, mais semelhante as sequências e mais geneticamente relacionadas (homólogas) elas possivelmente serão. De maneira simplificada, o programa funciona em três etapas: 1) é feita uma subdivisão da sequência de pesquisa (*query*) em subsequências de tamanho definido (*words*). 2) As subsequências são alinhadas contra o conjunto de busca (*target*), respeitando valores de alinhamento mínimos. 3) Para cada par encontrado (*query* e *target*), o BLAST estende o alinhamento em ambos os lados da sequência *query* para encontrar alinhamentos maiores, limitando a sua extensão até um limite mínimo de pontuação de corte. Estes alinhamentos encontrados são chamados de HSPs - *High Scoring Pairs* (FIGURA 12).

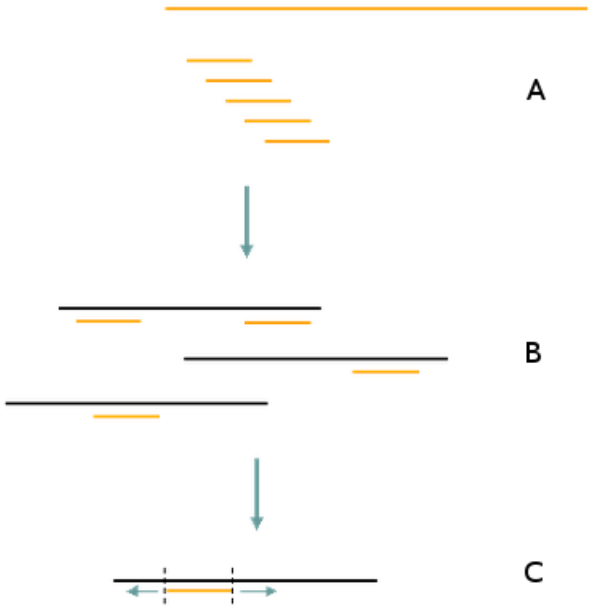


FIGURA 12: ALGORITMO DO BLAST

A) Subsequências da sequência *query*; B) Alinhamento das subsequências *query* com a(s) sequência(s) *target*; C) Extensão dos alinhamentos
FONTE: Adaptada (CGIAR, 2014)

O BLAST é disponibilizado como um pacote de programas. Ele pode ser utilizado via internet, consultando diretamente o repositório do NCBI, e via programa

executável local. O BLAST realiza alinhamentos com *gaps* (ALTSCHUL et al., 1997) e possui variações para entradas de arquivos em nucleotídeos (DNA) e aminoácidos (proteínas). Os diferentes programas são (*taget* x *query*):

- DNA x DNA (blastn)
- Proteína x Proteína (blastp)
- Proteína x DNA traduzido nas seis sequências de leituras possíveis (blastx)
- DNA traduzido nas seis sequências de leituras possíveis x Proteína (tblastn)
- DNA traduzido nas seis sequências de leituras possíveis x DNA traduzido nas seis sequências de leituras possíveis (tblastx)

1.6.2 MUMmer

O MUMmer (KURTZ et al., 2004) é um programa de alinhamento de sequências de nucleotídeos e aminoácidos. É um programa modularizado e de alto desempenho muito utilizado para comparação de sequências grandes de genomas inteiros. Ele realiza alinhamento global de múltiplas sequências e possuí módulos para alinhamentos locais, entre diversas outras funcionalidades.

1.6.3 QUAST

O QUAST (GUREVICH et al., 2013) é um programa que tem a finalidade de gerar relatórios para análises de montagens de sequências de genomas. Ele tem a capacidade de analisar diversas montagens ao mesmo tempo, com ou sem a sequência de um organismo de referência. Ele gera métricas e gráficos que permitem uma comparação ampla de diversos parâmetros envolvendo contagem e quantificação de nucleotídeos, propriedades do genoma, regiões de erros, inversões de montagem, regiões gênicas, entre outras.

1.6.4 IMAGE

O IMAGE é um programa para fechamento de *gaps*, desenvolvido no Instituto Sanger (TSAI; OTTO; BERRIMAN, 2010). O algoritmo utiliza bibliotecas de *reads mate-pair* ou *paired-end* para solucionar os *gaps*. Primeiramente os *reads* são alinhados contra a sequência do genoma desejado (em forma de *scaffold*, com *gaps*). São identificados os *reads* que alinham nas sequências dos finais de *contigs*. Os pares destes *reads* são analisados, buscando aqueles que possivelmente estão na região do *gap*, considerando o tamanho do fragmento e da inserção da biblioteca em que foi feito o sequenciamento. Estes *reads* são então separados do conjunto e é feita uma montagem local, apenas com eles. Os *contigs* gerados nesta montagem são então incorporados aos finais de *contigs* da sequência principal, através de alinhamentos entre os dois conjuntos. Este processo é repetido até o programa solucionar a região desconhecida, gerando extensões que cheguem a outra ponta do *contig*, completando o *gap*, ou apenas incorporando novas bases, estendendo a região.

1.6.5 GapFiller

O GapFiller (BOETZER; PIROVANO, 2012) é um programa que tem o objetivo de finalizar montagens genômicas através do fechamento de *gaps*. Ele tem método similar ao programa IMAGE citado anteriormente, mapeando os *reads* pareados contra as pontas dos *contigs*, identificando os pares e fechando o *gap*. Porém o GapFiller possui suporte para múltiplas bibliotecas de sequenciamento e não realiza montagens locais para estender os *gaps*. Ao invés disto, após a identificação dos pares que estão na região do *gap*, o algoritmo quebra os *reads* em k-mers e estende a ponta dos *contigs* a partir destes fragmentos. Outra diferença é que o tamanho do *gap* (quantidade de bases desconhecidas) é considerada no momento da finalização, abordagem que traz algumas vantagens pois o programa consegue estimar quantas bases serão necessárias para o fechamento do *gap*. Porém, isto pode levar a resultados incorretos caso a estimativa esteja equivocada (SAHLIN et al., 2012).

1.6.6 GapCloser

O GapCloser é um módulo do programa de montagem SOAPdenovo2 (LUO et al., 2012). O algoritmo tem o objetivo de melhorar as montagens através do fechamento de *gaps*. O método é similar as abordagens do IMAGE e GapFiller, baseando-se em alinhamento, montagem e extensão, porém foi desenvolvido para dados de sequenciamento de equipamentos Illumina, limitando a sua aplicação.

1.7 JUSTIFICATIVA

Assim como os custos para obter sequências genômicas vem caindo nos últimos anos (WETTERSTRAND, 2014) (FIGURA 4), os aparelhos de sequenciamento de DNA estão ficando acessíveis para pequenos e médios laboratórios (NAGARAJAN; POP, 2013). Com isto, existe uma quantidade crescente de pessoas e grupos que lidarão com dados de sequenciamento, sendo muitas destas com pouco conhecimento para manipulação dos dados gerados.

Existe uma escassez de programas automatizados de bioinformática, que apesar de estar em plena evolução (OUZOUNIS, 2012), ainda carece de um conjunto de ferramentas específicas para diversas etapas na montagem e finalização de sequências de genomas. É crescente a necessidade de métodos computacionais automatizados que possam analisar de maneira precisa e padronizada estes dados, gerando resultados confiáveis.

Abordagens que utilizam métodos híbridos e a reutilização dos dados já existentes são uma maneira eficaz para finalização sequências de genomas: utilizar *reads* de sequenciamento de duas diferentes tecnologias (BASHIR et al., 2012), utilizar as informações de *reads* pareados (WETZEL; KINGSFORD; POP, 2011), diversificar os algoritmos de montagem para aproveitar as vantagens de cada técnica (LIN; LIAO, 2013), entre outros, vem se tornado práticas frequentes em projetos de montagem. Porém é muitas vezes realizada de maneira manual, sendo custosa e sem padronização.

Os programas disponíveis atualmente com objetivo de melhorar montagens através do fechamento de *gaps* se baseiam no mesmo conceito de mapeamento, iden-

tificação de pares e extensão de pontas de *contigs*. Esta abordagem se mostra eficaz porém apresenta limitações, como a dependência de *reads* pareados, alto custo computacional e nenhuma delas possui um método fácil para validação dos resultados obtidos, tornando-os difíceis de serem analisados.

A grande quantidade de sequências de genomas depositados em bancos de dados públicos que continuam não finalizados (PAGANI et al., 2012) podem ser melhorados através da reutilização de dados já obtidos e métodos *in silico*, com baixo custo. Estes métodos também podem ser aplicados para reduzir o custo de posteriores finalizações *in vitro* de projetos de montagem que buscam a finalização completa de sequências de genomas.

1.8 OBJETIVOS

1.8.1 Objetivo Geral

Desenvolver uma ferramenta automatizada para finalização de montagens e fechamento de *gaps in silico*, reutilizando dados já obtidos para melhorar sequências genômicas.

1.8.2 Objetivos Específicos

Utilizar dados gerados por múltiplas tecnologias de sequenciamento, programas e técnicas de montagem diferentes e outras formas de diversificação de dados.

Validar as técnicas desenvolvidas comparando com programas atuais.

Aplicar as técnicas em projetos reais de montagem e avaliar resultado.

2 MATERIAIS E MÉTODOS

2.1 FGAP

FGAP é um programa de código aberto para fechamento automático de *gaps*. Ele busca melhorar montagens de genomas incorporando dados alternativos, analisando a região do *gap* e selecionando a melhor sequência para eliminá-lo.

O programa procura por sequências que sobreponham finais de *contigs* em rascunhos de genomas em formato de *scaffolds*, ou seja, com *gaps* entre os *contigs* já definidos. O algoritmo alinha os finais de *contigs* contra um ou mais conjunto de dados, restringe os alinhamentos por parâmetros controlados e escolhe a melhor sequência para eliminar o *gap*.

A ferramenta web para execução online e o o link para download dos arquivos compilados estão em www.bioinfo.ufpr.br/fgap/ e www.sourceforge.net/projects/fgap/, respectivamente.

2.1.1 Algoritmo

O programa tem como entrada arquivos no formato FASTA de sequências de nucleotídeos. Usualmente são os arquivos de *scaffolds* propostos gerados na etapa de montagem, chamado de rascunho da sequência do genoma ou *draft*, e um ou mais arquivos de conjuntos de dados alternativos, chamados aqui de *datasets*.

O algoritmo inicia localizando todas as bases desconhecidas na montagem, as lacunas ou *gaps*, identificadas pelo caractere "N" no arquivo *draft*. Estes *gaps* geralmente são introduzidos na etapa de *scaffolding*, unindo *contigs* adjacentes. O FGAP resgata as regiões anteriores e posteriores aos *gaps*, que são as sequências nos finais dos *contigs*. Opcionalmente é possível ignorar alguns pares de base que, dependendo do método de sequenciamento usado, podem ser de baixa qualidade (FIGURA 13).

Os finais de *contigs* são então utilizados como dados de consulta (*query*) no programa blastn (ALTSCHUL et al., 1997), e os arquivos *dataset*, são utilizados para

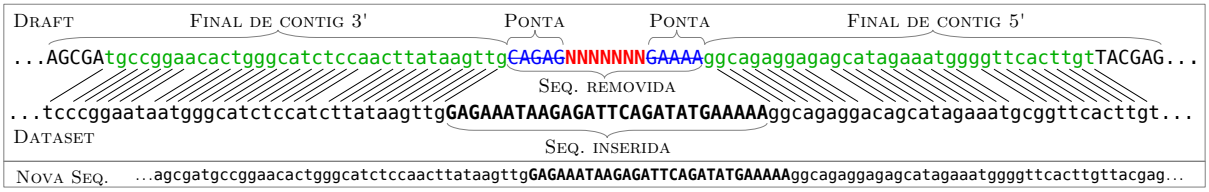


FIGURA 13: DETALHAMENTO DE UM *GAP* POSITIVO TRATADO PELO FGAP

As sequências ligadas por linhas diagonais representam os alinhamentos gerados pelo BLAST, evidenciando uma sobreposição em dois finais de *contig* com um arquivo do *dataset*
FONTE: O autor (2014)

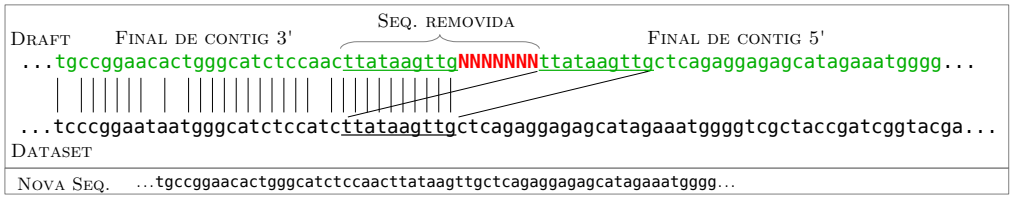


FIGURA 14: DETALHAMENTO DE UM *GAP* NEGATIVO TRATADO PELO FGAP

As sequências ligadas por linhas diagonais representam os alinhamentos gerados pelo BLAST, evidenciando uma sobreposição em dois finais de *contig* com um arquivo do *dataset* assim como a sobreposição entre o final de contig 5' e 3'
FONTE: O autor (2014)

gerar o banco de dados (*database*) (FIGURA 15). O processo visa identificar sequências na base de dados que sobreponham dois finais de *contigs*, como mostra o exemplo da FIGURA 13. Nesta consulta são aplicados os seguintes filtros para restringir alinhamentos de baixa qualidade: escore mínimo, máximo *E-value* e identidade mínima. Parâmetros específicos do BLAST como penalidades e valores de alinhamento também podem ser controlados.

A saída gerada pelo BLAST é processada e validada pelo FGAP, selecionando apenas os resultados que contenham alinhamentos em ambos os finais de *contigs* de um *gap* com um mesmo *dataset* pertencendo ao mesmo sentido da fita de DNA.

Tendo este conjunto de alinhamentos passíveis para o fechamento, considerados aqui como pré-candidatos, o FGAP analisa e classifica 3 tipos de *gaps*: (A) positivo, (B) zero e (C e D) negativo (FIGURA 16). *Gaps* positivos, ou *gaps* reais, são aqueles em que os finais de *contigs* alinharam contra *datasets* na mesma ordem em que estão no *draft*, ou seja, em sequência, com uma região de 1pb ou mais entre eles. Neste tipo de fechamento é removido parte da sequência do *draft* e inserido parte da sequência do *dataset*, substituindo a região do *gap* por bases conhecidas (FIGURA 13). Os *gaps* zero são semelhantes aos *gaps* positivos porém, quando considerado

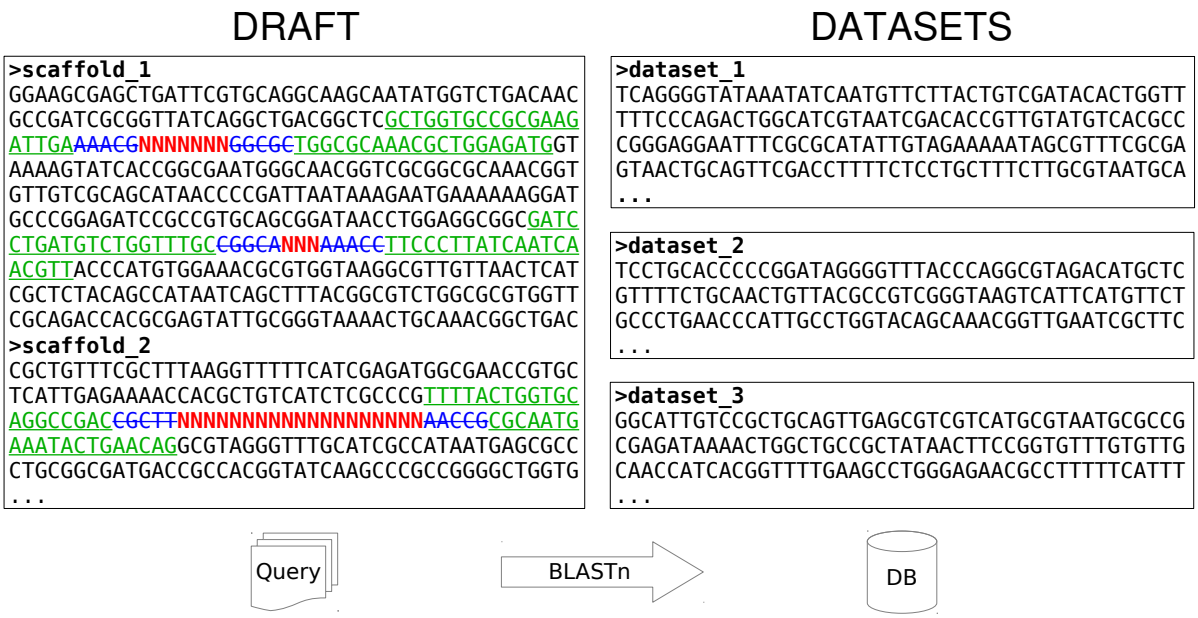


FIGURA 15: VISÃO GERAL DO FUNCIONAMENTO DO FGAP

Primeiramente são identificadas as bases desconhecidas (N) no *draft*, indicadas em vermelho. Bases em azul riscado são bases opcionalmente ignoradas. As bases em verde representam os finais dos *contigs* que serão utilizados como consulta (*query*) contra o conjunto de dados alternativos (DB - *database*). Os parâmetros foram reduzidos para melhor visualização: final de *contig*=20pb, corte de ponta=5pb.
FONTE: O autor (2014)

o alinhamento com o *dataset*, não possuem nenhuma base entre os finais de *contigs*. *Gaps* negativos representam alinhamentos onde os finais de *contigs* alinharam contra o *dataset* em regiões sobrepostas, indicando uma possível redundância dos dados no processo de montagem (FIGURA 14). Os *gaps* zero e negativos irão apenas remover parte da sequência *draft*. Nesta etapa é também limitado o número máximo de bases que podem ser removidas do *draft* e o máximo que pode ser inserida a partir do *dataset*.

Como é possível que exista mais que um pré-candidato para cada *gap*, é feita uma escolha entre os alinhamentos que possuam os melhores critérios, somando os valores de ambos os finais de *contigs*, nesta ordem: escore, cobertura da *query*, identidade, *E-value*. Caso exista empate entre todos os critérios (comum quando existem dados redundantes nos *datasets*) o candidato escolhido será aquele que estiver alinhado no maior *contig* entre os pré-candidatos. Este alinhamento é então considerado o candidato escolhido para o fechar o *gap*.

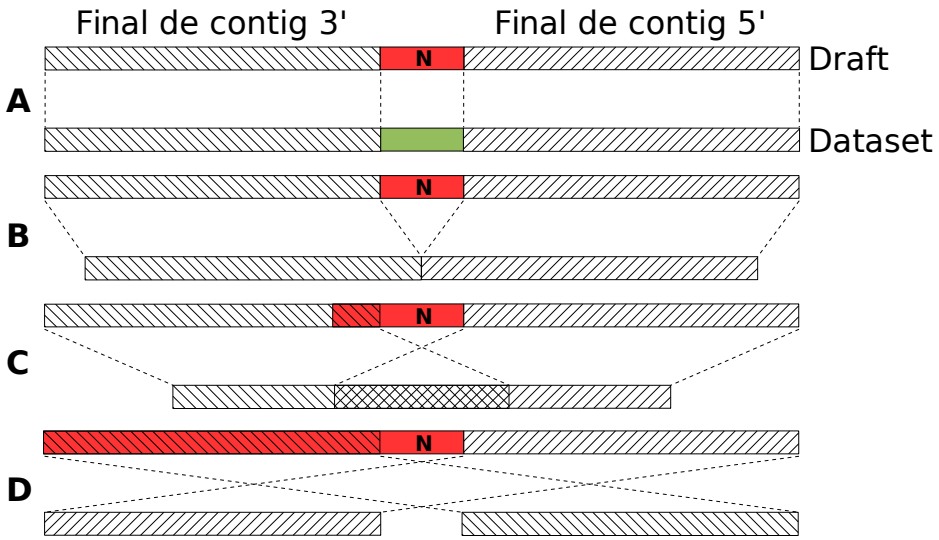


FIGURA 16: TIPOS DE *GAP* CONSIDERADOS PARA FECHAMENTO PELO FGAP

A) *Gap* positivo; B) *Gap* zero; C) e D) *Gap* negativo. As barras com listras verticais representam a parte das sequências dos finais dos *contigs* (*draft*) alinhadas com o *dataset*. A parte vermelha da barra representa a sequência removida. A parte verde da barra representa a sequência inserida.
FONTE: O autor (2014)

2.1.2 Implementação

O FGAP foi desenvolvido em MATLAB/OCTAVE (nas versões: R2012a (MATLAB, 2013) e 3.6.2 (Octave community, 2014), respectivamente), podendo ser executado em qualquer uma das duas linguagens. O programa pode ser executado localmente via código fonte, via programa compilado (com dependência da biblioteca MRC) ou através da internet, via formulário on-line. O programa utiliza as ferramentas blastn e makeblastdb do pacote BLAST+ versão 2.2.28 ou mais recente (ALTSCHUL et al., 1997).

O programa requer no mínimo 2 arquivos no formato fasta para ser executado: o arquivo da sequência do genoma pré-montado em *scaffolds* e um ou mais arquivos de dados alternativos para fechamento de *gaps*. O algoritmo é executado em várias rodadas para evitar sobreposições de finais de *contigs* de *gaps* próximos, prevenindo alterações nas sequências de busca destas regiões. O programa tem diversos parâmetros para controle de suas ações, descritos na TABELA 3

TABELA 3: PARÂMETROS DO FGAP

Parâmetro	Descrição
-d (--draft-file)	Arquivo *Draft* de entrada que terá os *gaps* fechados
-a (--datasets-files)	Arquivo(s) *Dataset(s)* de entrada que será utilizado como base de dados para o fechamento
-s (--min-score)	Escore mínimo do BLAST (*raw score*) para aceitar alinhamentos válidos
-e (--max-evalue)	*E-value* máximo do BLAST para aceitar alinhamentos válidos
-i (--min-identity)	Identidade mínima do BLAST para aceitar alinhamentos válidos
-C (--contig-end-length)	Número de bases (pb) nos finais de *contig* onde será feita o alinhamento
-T (--edge-trim-length)	Número de bases (pb) nas pontas dos finais de *contigs* ignoradas no alinhamento
-R (--max-remove-length)	Número máximo de bases (pb) que podem ser removidas do *Draft* por *gap* (apenas nos gaps positivos)
-I (--max-insert-length)	Número máximo de bases (pb) que podem ser inseridas no *Draft* por *gap*
-p (--positive-gap)	Ativa ou desativa fechamento de *gaps* positivos
-z (--zero-gap)	Ativa ou desativa fechamento de *gaps* zero
-g (--negative-gap)	Ativa ou desativa fechamento de *gaps* negativos
-c (--gap-char)	Caractere que identifica o *gap*
-b (--blast-path)	Caminho dos arquivos executáveis do BLAST
-l (--blast-alignment-parameters)	Parâmentros de alinhamento do BLAST
-r (--blast-max-results)	Número máximo de resultados obtidos por alinhamento do BLAST
-t (--threads)	Número de núcleos do processador a serem utilizados (*threads*)
-m (--more-output)	Ativa saída de arquivos adicionais
-o (--output-prefix)	Caminho e prefixo dos arquivos de saída
-h (--help)	Mostra informações de ajuda

Referente a versão 1.7
FONTE: O autor 2014

2.1.3 Arquivos de saída

Os arquivos de saída do FGAP tem o objetivo de mostrar os fechamentos realizados e alterações na sequência feitas pelo programa. A partir destes arquivos é possível controlar o que foi feito, assim como realizar análises posteriores. Os arquivos gerados pelo programa são (o * representa que estes arquivos serão gerados a cada rodada necessária feita pelo FGAP):

- prefixo_*.fasta
- prefixo_*.log
- prefixo.stats
- prefixo.final.fasta
- prefixo.before.fasta
- prefixo.after.fasta

O arquivo ".log" de cada rodada contém dados dos *gaps* fechados pelo FGAP, assim como o alinhamento de ambos os finais de *contigs* com o *dataset*, os escores de alinhamento, regiões inseridas e removidas, posições relativas das bases nos arquivos originais, entre outras informações (FIGURAS 17 e 18).

O arquivo ".fasta" representa a sequência do genoma com as alterações contidas no respectivo log da rodada. As mudanças são incrementais nos arquivos fasta. O arquivo ".final.fasta" é o último arquivo da sequência do genoma contendo todas as melhorias incrementais por rodada.

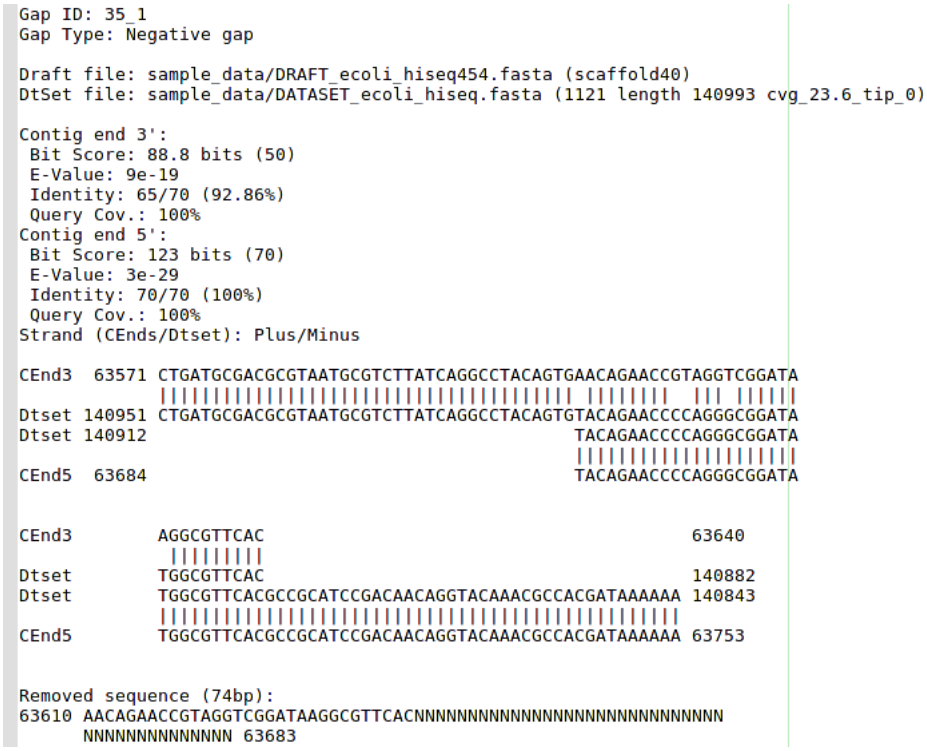
O arquivo ".stats" mostra estatísticas gerais do fechamento antes e depois do FGAP, mostrando também a variação de diversas métricas por rodada.

Os arquivos "before.fasta" e "after.fasta" contém somente as regiões dos *gaps* antes e depois do fechamento, respectivamente, para análises e validações posteriores, no formato fasta. Estes arquivos podem ser utilizados para validar os resultados obtidos pelo programa, realizando uma busca dos arquivos "after.fasta" contra uma base de sequências de genomas de referência, verificando a qualidade dos alinhamentos gerados.

```
Gap ID: 1_1
Gap Type: Positive gap

Draft file: sample_data/DRAFT_ecoli_hiseq454.fasta (scaffold1)
DtSet file: sample_data/DATASET_ecoli_454.fasta (19165 length 611 cvg_2.9_tip_1)

Contig end 3':
 Bit Score: 123 bits (70)
 E-Value: 3e-29
 Identity: 70/70 (100%)
 Query Cov.: 100%
Contig end 5':
 Bit Score: 123 bits (70)
 E-Value: 3e-29
 Identity: 70/70 (100%)
 Query Cov.: 100%
Strand (CEnds/Dtset): Plus/Plus

CEnd3 144 ACGCAATGAGTGAAAAGAGTCACGGCAGAGAAGAAATCCGTCTTCATATTGTTTGCGATG
          ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Dtset 200 ACGCAATGAGTGAAAAGAGTCACGGCAGAGAAGAAATCCGTCTTCATATTGTTTGCGATG
Dtset 270

CEnd5 228


CEnd3     TCCCTGATGA
          ||||||||||
Dtset     TCCCTGATGAacttattgatttcacgtttgaatggaaagg
Dtset               acttattgatttcacgtttgaatggaaaggACTGAAGAAATTATGCGTGG
                                                 ||||||||||||||||||||
CEnd5                                            ACTGAAGAAATTATGCGTGG


CEnd3                                                          213

Dtset                                                          299
Dtset     CAGTCTCCTTTCGGTCAATAATAGCAGAACAAAAGAAAGAGCCAGAAATG 369
          |||||||||||||||||||||||||||||||||||||||||||||||||||
CEnd5     CAGTCTCCTTTCGGTCAATAATAGCAGAACAAAAGAAAGAGCCAGAAATG 297


Removed sequence (14bp):
214 NNNNNNNNNNNNNN 227
Inserted sequence (30bp):
270 acttattgatttcacgtttgaatggaaagg 299
```

FIGURA 17: LOG - *GAP* POSITIVO

Exemplo de arquivo de log para um *gap* positivo fechado no FGAP (FIGURA 16 A). Parâmetros foram ajustados para melhor visualização.
FONTE: O autor (2014)



```
Gap ID: 35_1
Gap Type: Negative gap

Draft file: sample_data/DRAFT_ecoli_hiseq454.fasta (scaffold40)
DtSet file: sample_data/DATASET_ecoli_hiseq.fasta (1121 length 140993 cvg_23.6_tip_0)

Contig end 3':
 Bit Score: 88.8 bits (50)
 E-Value: 9e-19
 Identity: 65/70 (92.86%)
 Query Cov.: 100%
Contig end 5':
 Bit Score: 123 bits (70)
 E-Value: 3e-29
 Identity: 70/70 (100%)
 Query Cov.: 100%
Strand (CEnds/Dtset): Plus/Minus

CEnd3  63571 CTGATGCGACGCGTAATGCGTCTTATCAGGCCTACAGTGAACAGAACCGTAGGTCGGATA
             |||||||||||||||||||||||||||||||||||||||| ||||||||  ||| ||||||
Dtset 140951 CTGATGCGACGCGTAATGCGTCTTATCAGGCCTACAGTGTACAGAACCCCAGGGCGGATA
Dtset 140912                                        TACAGAACCCCAGGGCGGATA
                                                    ||||||||||||||||||||
CEnd5  63684                                        TACAGAACCCCAGGGCGGATA


CEnd3        AGGCGTTCAC                                          63640
              |||||||||
Dtset        TGGCGTTCAC                                          140882
Dtset        TGGCGTTCACGCCGCATCCGACAACAGGTACAAACGCCACGATAAAAAA 140843
             ||||||||||||||||||||||||||||||||||||||||||||||||||
CEnd5        TGGCGTTCACGCCGCATCCGACAACAGGTACAAACGCCACGATAAAAAA 63753


Removed sequence (74bp):
63610 AACAGAACCGTAGGTCGGATAAGGCGTTCACNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
      NNNNNNNNNNNNNN 63683
```

FIGURA 18: LOG - *GAP* NEGATIVO

Exemplo de arquivo de log para um *gap* negativo fechado no FGAP (FIGURA 16 A). Parâmetros foram ajustados para melhor visualização.
FONTE: O autor (2014)

2.2 CONJUNTOS DE DADOS

O FGAP foi projetado com o objetivo de reutilizar diversos conjuntos de dados referentes ao mesmo genoma, que usualmente são gerados nos processos de montagem: montagens com diferentes parâmetros e variações de k-mer, montagens com diferentes programas que utilizam paradigmas distintos, montagens oriundas de diferentes corridas de sequenciamento. *Reads* longos e quaisquer outros métodos que forneçam regiões contíguas a partir dos *reads* também podem ser utilizados. O FGAP parte do princípio que dados gerados por diferentes técnicas ou tratados de diferentes maneiras podem resolver regiões desconhecidas na montagem do genoma, e que elas se complementam.

Para a realização deste trabalho foi imprescindível a utilização de bancos de dados públicos. Os arquivos obtidos foram retirados da base de dados primária *GenBank* (BENSON et al., 2013) do NCBI (COORDINATORS, 2013). Também foram utilizados conjuntos de *reads* obtidos no banco NCBI SRA. As sequências de genes uti-

lizados como referência para validação foram retirados do banco de dados de genes, também do NCBI.

2.2.1 Programas de montagem

Diversos programas de montagem de genomas estão disponíveis atualmente e são em sua maioria de acesso livre e de código aberto. Todos possuem o mesmo objetivo final: gerar uma sequência montada a partir de *reads* provenientes dos equipamentos de sequenciamento. Porém, os programas variam em alguns aspectos: paradigmas de montagem, métodos para correção de erros, tipo e origem dos dados de entrada e nos objetivos de montagem (genoma, transcriptoma, metagonomas, etc). A escolha do programa ideal para cada tipo de dado e objetivo de montagem pode ser muitas vezes confusa (NAGARAJAN; POP, 2013). Estudos foram feitos com o objetivo de avaliar as diferenças, vantagens e desvantagens destes programas, tanto em dados simulados (EARL et al., 2011) como em dados reais (SALZBERG et al., 2012) (MAGOC et al., 2013) (BRADNAM et al., 2013), em uma diversa gama de organismos e situações. Eles visam analisar de maneira ampla os programas de montagem e, além de serem de extrema importância para avaliação dos resultados, servem como um guia para encontrar o programa correto para cada situação.

A montagem da sequência do genoma de um organismo não é um processo trivial e para obter um bom resultado final é inevitável a utilização de diversos programas com diversas variações de parâmetros em cada um deles. Equipamentos diferentes geram dados diferentes, não existindo uma regra específica para cada montagem. Este processo de tentativa e erro na busca de uma montagem ideal gera um conjunto de dados que geralmente é descartado, por ser redundante. Porém cada programa e parâmetro interferem na sequência montada, gerando um conjunto de dados valioso que fica muitas vezes inutilizado. Estes conjuntos podem ser complementares, e se utilizados de maneira coerente, podem ajudar a melhorar o estado da melhor montagem obtida.

Para este trabalho, reutilizaremos dados gerados nos projetos GAGE (SALZBERG et al., 2012) e GAGE-B (MAGOC et al., 2013). Estes projetos foram realizados com o intuito de comparar programas de montagem de genomas. O GAGE avaliou

9 programas de montagens em 4 organismos, sendo 2 procariotos e 2 eucariotos. O projeto GAGE-B avaliou apenas organismos procariotos, de 9 diferentes espécies, utilizando 8 programas de montagem.

Dentre todas as montagens dos diversos organismos disponibilizados nos projetos GAGE e GAGE-B, foram selecionados aquelas que possuem sequências de referência finalizadas ou próximas de estarem finalizadas, possibilitando uma comparação posterior dos resultados obtidos. Os dados escolhidos estão detalhados na TABELA 4.

TABELA 4: DADOS SELECIONADOS DOS PROJETOS GAGE E GAGE-B

Organismo	Projeto	Bibliotecas	Montagens
R. sphaeroides (Procarioto)	GAGE	1 PE e 1 MP (GAII)	9 scf. e 9 ctg.
S. aureus (Procarioto)	GAGE	1 PE e 1 MP (GAII)	8 scf. e 8 ctg.
Cromossomo Humano 14 (Eucarioto)	GAGE	1 PE e 2 MP (HiSeq)	9 scf. e 9 ctg.
M. abscessus (Procarioto)	GAGE-B	2 PE (HiSeq e MiSeq)	14 scf. e 16 ctg.
R. sphaeroides (Procarioto)	GAGE-B	2 PE (HiSeq e MiSeq)	14 scf. e 16 ctg.
V. cholerae (Procarioto)	GAGE-B	2 PE (HiSeq e MiSeq)	14 scf. e 16 ctg.

PE: *paired-end*; MP: *mate-pair*; GAII: Illumina Genome Analyzer II; HiSeq: Illumina HiSeq; MiSeq: Illumina MiSeq; scf: *scaffold*; ctg: *contigs*. Os dados foram obtidos em: (GAGE, 2014) e (GAGE-B, 2014). Mais detalhes dos dados utilizados e das montagens estão no APÊNDICE A - Dados GAGE e GAGE-B.
FONTE: O autor 2014

Os códigos de acesso (*GenBank accession number*) utilizados para as sequências de referência dos organismos acima são:

- *Rhodobacter sphaeroides 2.4.1* (NC_007493.2, NC_007494.2, NC_009007.1, NC_007488.2, NC_007489.1, NC_007490.2, NC_009008.1);
- *Staphylococcus aureus* subsp. aureus (NC_010079.1, NC_010063.1, NC_012417.1)
- Human chromosome 14 (NC_000014.9)
- *Mycobacterium abscessus ATCC 19977* (NC_010397.1, NC_010394.1)
- *Vibrio cholerae CO1032(5)* (NC_002505.1, NC_002506.1)

Os genes utilizados para validação de cada organismo de referência foram obtidos da base de dados de genes do NCBI, fazendo a busca pelos códigos de acesso supracitados, considerando genes com status *Current only*.

2.2.2 Corridas e bibliotecas de sequenciamento

Obter mais de uma corrida de sequenciamento em diferentes equipamentos implica em diversas vantagens para uma montagem genômica. Variações na química e preparação de amostras, variação na taxa e ocorrência de erros, variação no tamanho de *reads* e fragmentos, variação na cobertura, entre outros aspectos afetam o conjunto de *reads* gerado pelos equipamentos. Com a utilização de mais de uma corrida estas variações podem ser compensadas, principalmente nos aspectos que tendem a ser aleatórios como os erros de sequenciamento e regiões de baixa cobertura. Montagens que utilizam mais de uma corrida de sequenciamento possuem dados mais completos e precisos (RIBEIRO et al., 2012). Essa abordagem pode também ser utilizada no processo de finalização, tirando proveito de diferentes aspectos de cada corrida para complementar as montagens.

A utilização de bibliotecas de *reads mate-pair* ou *paired-end* é uma das formas mais eficientes para resolução de regiões de repetição no processo de montagem, assim como no processo de finalização de montagem de genomas. Bibliotecas com variação no tamanho do fragmento sequenciado (GNERRE et al., 2011) auxiliam no processo de *scaffolding*, dando mais confiança nas montagens e gerando *gaps* reais entre as regiões contíguas. Além disto, bibliotecas diversificadas têm um importante papel na validação da sequência genômica obtida.

Um dos conjuntos de dados utilizados neste trabalho usufruirá desta capacidade de aproveitamento de duas corridas para um mesmo organismo: a bactéria *Escherichia coli* str. K-12 substr. MG1655 (detalhes na TABELA 5). Os dados foram obtidos do NCBI SRA. Além disto, os dados do projeto GAGE-B possuem sequenciamentos e montagens distintas em dois equipamentos (Illumina HiSeq e MiSeq) e suas performances complementares para fechamentos de *gaps* com o FGAP serão avaliadas.

Código de acesso (*GenBank accession number*) da sequência do genoma de referência utilizado para a bactéria *E. coli*: NC_000913.3.

TABELA 5: DADOS DE SEQUENCIAMENTO DA BACTÉRIA *E. coli*

	Illumina HiSeq 2000	Roche 454 GS FLX
Biblioteca	*Paired-end* (inserção 200pb)	*Single-end*
Tamanho dos *reads* (Min., Máx., Média)	101, 101, 101	49, 1.090, 264
Quantidade de bases	1.023.909.114pb	65.141.579pb
Quantidade de *reads*	5.068.857	247.044
Cobertura esperada	222x	14x
Código SRA	SRR826451	SRR057662

FONTE: O autor 2014

2.2.3 *Reads* longos

O advento da terceira geração de sequenciamento possibilitou a geração de *reads* longos, quando comparado as gerações anteriores. Esta vantagem é diretamente aplicável no processo de montagem (CHIN et al., 2013), porém ainda é pouco utilizado de outras maneiras para finalizar montagens genômicas. Neste trabalho utilizaremos os dados de terceira geração como *datasets*, com o objetivo de resolver regiões previamente não montadas. Este tipo de situação é aplicável em um cenário onde já se possui uma sequêcia de genoma montado com alta qualidade que ainda possui regiões desconhecidas que não puderam ser resolvidas pelos *reads* curtos de segunda geração. Para evitar a alta complexidade de montagem, os novos dados podem ser incorporados com o FGAP.

Neste trabalho será utilizado um conjunto de *reads* obtidos em um equipamento PacBio SMRT (detalhes na TABELA 6) que complementará os dados de Illumina e 454 (TABELA 5) para a finalização de montagem da bactéria *E. coli*, referenciado como FGAP+P.

TABELA 6: DADOS DE SEQUENCIAMENTO DA BACTÉRIA *E. coli* - PACBIO

	PacBio SMRT
Biblioteca	*Single-end*
Tamanho dos *reads* (Min., Máx., Média)	114, 24.529, 3.163
Quantidade de bases	258.565.857pb
Quantidade de *reads*	81.738
Cobertura esperada	56x
Código SRA	SRR811719

FONTE: O autor 2014

2.3 VALIDAÇÃO

Para avaliar os resultados foram utilizadas duas abordagens: local e global.

2.3.1 Validação Local

A validação local leva em conta apenas as regiões alteradas pelo FGAP, buscando verificar se a modificação na sequência do genoma feita pelo programa é consistente. A validação foi realizada utilizando o pacote NUCmer do programa MUMmer (KURTZ et al., 2004) e necessita de sequências de referência dos organismos testados. O *gap* é considerado corretamente fechado quando está de acordo com as seguintes regras: 1) Continuidade: 40% ou mais dos finais de *contigs* adjacentes a região inserida pelo FGAP estão alinhadas corretamente com a referência, incluindo a inserção (para *gaps* negativos, é verificado um mínimo de 80% de cobertura do alinhamento) 2) Identidade: a região alinhada possui identidade mínima, a mesma estipulada no FGAP e 3) Melhoria (apenas *gaps* positivos): a identidade da nova região (finais de *contigs* e inserção) deve ser maior que a identidade da mesma antes do fechamento do *gap*.

2.3.2 Validação Global

A validação global leva em conta diversas métricas, que são avaliadas antes e depois do fechamento de *gaps*. Com isto é possível verificar, de maneira global, a evolução da montagem genômica. Para isto foi utilizado a análise comparativa de métricas geradas pelo programa QUAST (GUREVICH et al., 2013). Foram selecionadas para este trabalho algumas métricas relevantes para o problema de finalização de sequências genômicas, tendo algumas delas calculadas com base em uma sequência de referência. A TABELA 7 contém as métricas do QUAST que serão utilizadas neste trabalho para a validação global. Os parâmetros utilizados na execução do programa estão no APÊNDICE B - Parâmetros.

TABELA 7: MÉTRICAS DO PROGRAMA QUAST

Métrica	Finalidade	Como é calculada
# Misassemblies (Falhas de montagem)	Quantificar as falhas de montagem	Verificando-se pontos na montagem em que: 1) regiões contíguas onde as bases a jusante e a montante estão a mais de 1kpb de distância ou 2) regiões de flanqueamento se sobrepõem em mais de 1kpb ou 3) regiões de flanqueamento alinham em fitas ou cromossomos diferentes
# Local misassemblies (Falhas de montagem locais)	Quantificar falhas locais de montagem	Verificando-se pontos na montagem em que dois alinhamentos cubram a região, com menos de 1kpb de distância entre eles e estando obrigatoriamente na mesma fita e cromossomo do genoma de referência
# Genes	Quantificar genes completos e parciais	Alinhando a montagem contra uma lista de genes de referência. Genes parciais são considerados quando o alinhamento for maior que 100pb porém não for completo
# Indels per 100kpb	Quantificar média de inserção e deleção na montagem	Dividindo o número de inserções (bases excedentes) e o de deleções (bases faltantes) por 100.000pb
# Mismatches per 100kpb	Quantificar média de falha de alinhamento de bases na montagem	Dividindo o número de bases individuais não alinhadas por 100.000pb
Duplication ratio (Taxa de duplicação)	Medir a taxa de duplicação da montagem	Número total de bases alinhadas na montagem dividido pelo número total de bases alinhadas na referência
Genome fraction (% Genoma de referência)	Quantificar fração de bases presentes na referência	Uma base alinhada na referência é contada se ao menos um *contig* tiver ao menos um alinhamento nesta base. A fração destas bases é calculada, considerando o tamanho do genoma de referência
Nx (onde $0 \leq x \leq 100$)	Medir a fragmentação da montagem	Retorna o valor do maior *contig* L tal que a soma do tamanho dos *contigs* $\geq L$ representem $x\%$ do tamanho total da montagem
NGAx ou Nx corrigido (onde $0 \leq x \leq 100$)	Medir a fragmentação da montagem com base na referência	Igual ao N50, porém são consideradas regiões contíguas alinhadas com a referência em vez de *contigs* e considera o tamanho total da referência não o da montagem

A descrição de todas as métricas estão no ANEXO A - Métricas QUAST
FONTE: O autor 2014

3 RESULTADOS E DISCUSSÃO

O FGAP foi testado em diversos conjuntos de dados e diferentes situações com o objetivo de demostrar sua funcionalidade e flexibilidade, validando seus resultados de maneira ampla.

Primeiramente foi realizado um teste controlado na sequência genômica da bactéria *E. coli* com *gaps* simulados para verificar a eficácia do FGAP em uma sequência genômica finalizada e conhecida.

Depois foram feitas aplicações em dados reais: primeiramente com a bactéria *E. coli*, utilizando o FGAP com dados montados de maneira alternativa. Para este mesmo conjunto também foi testada a capacidade de aplicação de *reads* longos para o fechamento de *gaps*. Estas aplicações serviram como base para uma comparação de eficiência e performance com programas que possuem a mesma finalidade.

Por fim foram testados dados disponibilizados pelos projetos GAGE (SALZBERG et al., 2012) e GAGE-B (MAGOC et al., 2013) (TABELA 4). Estes projetos foram publicados com a finalidade de comparar programas de montagem de genomas, disponibilizando sequências de diversos organismos já montados. Com base nestas montagens foi testada a capacidade do FGAP de utilizar diferentes montadores e diferentes corridas de sequenciamento para complementar a sequência de um genoma.

3.1 TESTE CONTROLADO

Este teste foi realizado para verificar a funcionalidade do FGAP em um caso controlado, tanto no *draft* quanto no *dataset*. Assim foi possível realizar uma validação comparando a montagem final pós-fechamento com a sequência do genoma conhecida e finalizada. Neste teste foi utilizada a montagem do genoma finalizado da bactéria *E. coli* str. K-12 substr. MG1655. Nesta sequência foram inseridos aleatoriamente 500 *gaps* em toda a sua extensão. Os *gaps* foram divididos entre 300 *gaps* positivos, 100 *gaps* zero e 100 negativos. Os *gaps* inseridos possuem comprimento aleatório, limitados em no máximo 500pb. Este conjunto foi utilizado como *draft* no FGAP. O *dataset* foi gerado a partir da mesma sequência finalizada, dividida em *con-*

tigs a cada 5000pb, com intersecção de 3000pb, garantindo cobertura total do genoma com redundância e fragmentação. Os resultados obtidos são mostrados na TABELA 8.

TABELA 8: TESTE CONTROLADO

	REFERÊNCIA	*DRAFT*	FGAP
Fração do genoma (%)	100	98.33	100
Tamanho total (pb)	4641652	4694366	4641652
# Falhas de montagem locais	0	436	0
# Genes completos + parciais	4497 + 0	4076 + 404	4497 + 0
# N's	0	106098	0

Os parâmetros utilizados estão no APÊNDICE B - Parâmetros
FONTE: O Autor (2014)

Após a execução do FGAP a sequência inicial é completamente regenareda com o mesmo número de bases e com todos os genes corretamente resolvidos, reintegrando os 106.098pb retirados na simulação. Isto demonstra a capacidade do FGAP em identificar as regiões corretas em um conjunto fragmentado e redundante, incorporando os dados na sequência do genoma sempre que disponíveis no *dataset*.

3.2 APLICAÇÃO DO MÉTODO - *E. coli* str. K-12 substr. MG1655

O FGAP foi aplicado em um processo de montagem, realizado como parte deste trabalho. O objetivo foi simular uma situação real e por muitas vezes recorrente em processos de montagem. A simulação teve como base duas corridas de sequenciamento distintas da bactéria *E. coli* str. K-12 substr. MG1655 (TABELA 5). Foram testados diversos parâmetros e variações de k-mer, buscando identificar a melhor montagem (maior N50) utilizando o programa SOAPdenovo2 (LUO et al., 2012). Além de obter uma montagem principal, utilizada como *draft* nas aplicações do FGAP, foram feitas duas montagens alternativas, com os mesmos dados, em busca de conjuntos que possam complementar a montagem principal. As montagens alternativas foram feitas com os dados das duas corridas de sequenciamento separadamente, formando dois conjuntos de *contigs* que foram utilizados como *datasets* (TABELA 9).

As montagens foram então submetidas ao FGAP. Os resultados obtidos com os parâmetros padrão (APÊNDICE B - Parâmetros) foram: 97 *gaps* fechados, diminuindo as regiões desconhecidas propostas nos *scaffolds* de 123 para 26 (78%). Apli-

TABELA 9: MONTAGENS DA BACTÉRIA *E. coli*

	k-mer	*GAPS*	SEQUÊNCIAS	TAMANHO(pb)	N50(pb)
Illumina(PE) + 454(SE) [*Draft*]	81	123	41(s)/32(c)	4554392	172167
454(SE) [*Dataset*]	99	0	12407(c)	6274970	531
Illumina(SE) [*Dataset*]	81	0	564(c)	4615235	63640

Os *datasets* foram montados considerando os conjuntos como single-end, gerando apenas *contigs*; PE: *paired-end*; SE: *Single-end*; s: *scaffolds*; c: *contigs*
FONTE: O Autor (2014)

cando a validação local, 96% (94/97) das novas regiões inseridas estão de acordo com a sequência do genoma finalizado de referência da *E. coli*. Estes resultados demonstram a capacidade do FGAP em integrar dados montados de maneiras distintas, utilizando o mesmo conjunto de *reads* que gerou a melhor montagem.

3.2.1 *Reads* longos

Um conjunto de *reads* sequenciado em um equipamento de terceira geração foi utilizado diretamente como *dataset* no FGAP. Como os *reads* desta geração são maiores, variando de 100pb a mais de 20.000pb, eles funcionam como *contigs* para o FGAP, sem a necessidade de montagem. Utilizando o *draft* da bactéria *E. coli* do teste anterior, e utilizando como *dataset* os dados mostrados na TABELA 6 o FGAP fechou 121 dos 123 *gaps* propostos. Todas as inserções (100%) foram validadas localmente com a sequência do genoma finalizado de referência. Este teste demonstra o grande potencial do FGAP em trabalhar com dados provenientes das novas tecnologias de sequenciamento, incorporando dados diretamente sem a necessidade de uma nova montagem, resolvendo a maioria das regiões desconhecidas corretamente.

3.2.2 Comparação com outros programas

GapCloser, GapFiller e IMAGE são programas que tem o objetivo de melhorar montagens genômicas através de fechamento de *gaps*. Eles foram comparados com o FGAP. Esta comparação foi feita com o mesmo *draft* da bactéria *E. coli* str. K-12 substr. MG1655 anteriormente citado. Os três programas necessitam obrigatoriamente de *reads* pareados para a finalização e não possuem suporte para *reads*

longos *single-end*. Em todos eles, os *reads* Illumina foram utilizados, e os demais parâmetros estão detalhados no APÊNDICE B - Parâmetros. A TABELA 10 compara as sequências genômicas utilizando as métricas globais do QUAST, após a execução de cada programa. A validação foi feita apenas globalmente pois os programas não possuem uma saída específica identificando as bases removidas e inseridas na montagem, inviabilizando uma validação local precisa.

TABELA 10: COMPARAÇÃO DE RESULTADOS ENTRE PROGRAMAS PARA A BACTÉRIA *E. coli*

	MONTAGEM	FGAP	FGAP+P	GAPCLOSER	GAPFILLER	IMAGE
# *Gaps* (no *scaffold*)	123	26	2	22	25	22
# *Contigs* $\geq$ 1000pb	116	80	73	82	85	91
# Falhas de montagem	1	1	1	1	1	1
# Falhas de montagem locais	2	9	2	12	12	22
# Genes completos	4325	4377	4388	4375	4367	4386
# Genes parciais	44	34	27	35	35	69
# Indels per 100kpb	0.070	0.9	13.12	0.15	0.53	1.03
# Mismatches per 100kpb	0.18	4.110	3.36	1.19	1.160	6.55
Taxa de duplicação	1	1	1	1	1	1.003
% Genoma de referência	97.714	98.084	98.177	98.108	98.033	98.672
N50	66462	132608	172148	112396	132608	110882
NGA50	63640	114132	133062	106887	132608	107145
# Bases inseridas (pb)	-	3512	7366	3541	522	38517
Tempo execução	-	26s	1m55s	2m35s	46m45s	9h37m

As análises foram feitas em *contigs* (*scaffolds* quebrados nos *gaps* após o fechamento). FGAP = fechamento com os dados da TABELA 5. FGAP+P = fechamento de *gaps* utilizando *reads* longos. O número de genes foi calculado baseado em uma lista de referência de genes da *E. coli* com 4497 genes. A lista completa de métricas da validação global estão no APÊNDICE C - Resultados complementares.
FONTE: O Autor (2014)

Nota-se que as finalizações geradas pelo FGAP tem melhores resultados em relação ao número de *contigs*, falhas de montagens locais e tempo de execução. O FGAP unido aos *reads* longos de terceira geração se destacam no conjunto, não inserindo nenhuma falha de montagem, identificando maior número de genes e com a maior correção de genes parciais. Além disto este conjunto gera a montagem com melhor N50 e NGA50 além de ter todos os *gaps* fechados confirmados através da validação local. A maior quantidade de dados introduzido pelo conjunto dos *reads* de terceira geração unidos ao seu maior comprimento certamente possibilitam a obtenção de melhores resultados. Entretanto, este fechamento foi o que gerou maior número

de Indels por 100kpb, devido a sua conhecida alta taxa de erros (NAGARAJAN; POP, 2013). O FGAP já suporta esta tecnologia que pode ser aplicada de maneira mais vantajosa conforme evolua e tenha uma taxa de erro menor. Atualmente este conjunto de *reads* longos necessitaria de uma etapa de pré-processamento para diminuir sua taxa de erro e/ou uma correção posterior de indels inseridos, conforme já sugerido por (BASHIR et al., 2012).

Analisando apenas os conjuntos que utilizaram as corridas Illumina e 454, o FGAP foi o que inseriu menor número de erros locais, gerou o menor número de *contigs* e rodou em aproximadamente um quinto do tempo do GapCloser, o segundo programa mais rápido. O IMAGE possui maior tempo de execução e foi o que causou maior número de falhas de montagens locais, aumento de genes parciais, e falhas de alinhamento. Apesar de o IMAGE possuir maior cobertura no genoma de referência, foi o único programa que inseriu duplicações na montagem. Por fim, o número de bases inseridas por cada programa varia, devido a diferenças no método de extensão e fechamento de *gap* de cada um e também pelos diferentes erros gerados.

A FIGURA 19 mostra a variação de cada uma das métricas antes e depois do fechamento de gaps, obtidas pelo programa QUAST (detalhadas na TABELA 7).

3.3 OUTRAS APLICAÇÕES

Testes complementares foram realizados a partir de montagens geradas pelos projetos GAGE e GAGE-B. Eles possibilitaram um estudo de caso mais amplo da aplicação do FGAP, dado a variação de montagens disponibilizadas assim como a diversificação de organismos disponíveis. Além de utilizar as diversas montagens disponíveis para fechar *gaps* entre si, também foram testadas conjuntos de dados de diferentes sequenciadores e como eles se complementam.

3.3.1 GAGE

As montagens do projeto GAGE utilizadas nos testes com o FGAP foram geradas a partir de 9 programas de montagem (detalhes das montagens no APÊNDICE A - Dados GAGE e GAGE-B):

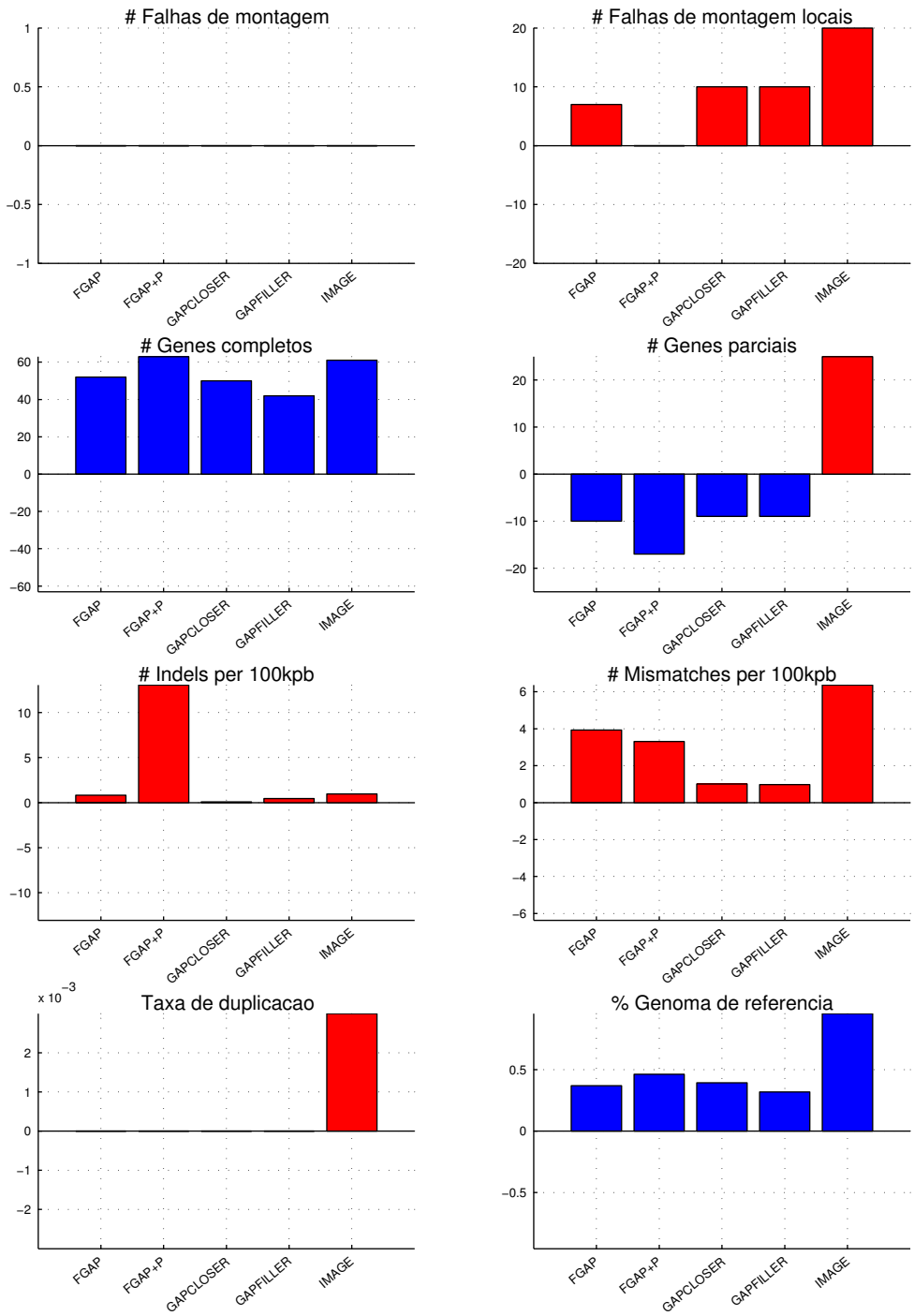


FIGURA 19: VARIAÇÃO DAS MÉTRICAS - *E. coli* - PROGRAMAS DE FECHAMENTO DE GAPS

Cada gráfico representa a variação de uma métrica. Para cada programa descrito no eixo x é apresentada a diferença de valor entre a métrica da montagem posterior ao fechamento de gaps em relação a montagem original. Barras azuis representam melhorias, tanto em métricas que tenham valores positivos (ex: genes completos) ou negativos (ex: genes parciais), e barras vermelhas representam piora na métrica.
Fonte: O autor (2014)

- ABySS
- ABySS2
- ALLPATHS-LG
- Bambus2
- CABOG
- MSR-CA
- SGA
- SOAPdenovo
- Velvet

Foram testados 2 organismos com os dados GAGE: as bactérias *R. sphaeroides* e *S. aureus*. Cada montador gerou diversas montagens por organismo. Todos eles possuem um arquivo contendo *contigs* e, nos casos dos programas que possuem módulo para *scaffolding* embutido, outro arquivo em *scaffolds*. Para cada organismo, cada arquivo de *scaffolds* de cada montagem foi submetido ao FGAP, totalizando 17 execuções. Como *datasets* foram utilizados os conjuntos gerados por todos os *contigs* de todos os montadores daquele organismo, inclusive o arquivo de *contig* do próprio *scaffold* que está sendo finalizado (FIGURA 20).

O objetivo deste teste foi verificar como as montagens podem ser melhoradas a partir de dados gerados em diferentes montadores que utilizaram o mesmo conjunto de *reads*. Aqui, as montagens não foram comparadas com uma montagem principal, mas sim cada montagem foi comparada com ela mesma após o fechamento de *gaps* com o FGAP, buscando avaliar a evolução de cada sequência genômica a partir de dados reaproveitados. Os resultados obtidos estão na TABELA 11.

Através desta estratégia foi possível reduzir significantemente o número de *gaps* utilizando o FGAP. Considerando os dois conjuntos analisados, foram fechados 51% dos *gaps*, com média de acerto de 86%, demonstrando que diferentes montadores têm a capacidade de complementar montagens a partir do mesmo conjunto inicial de dados.

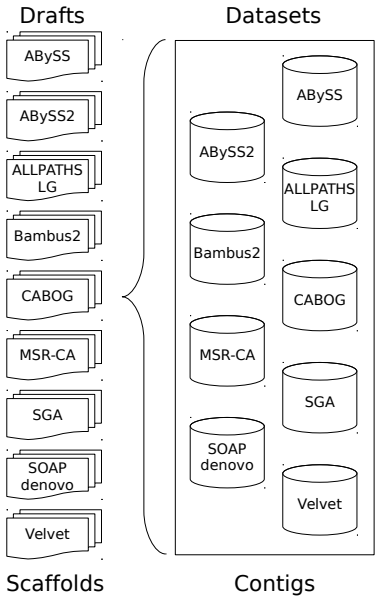


FIGURA 20: TESTE GAGE

Cada montagem em *scaffold* foi submetida ao FGAP utilizando todas as outras montagens em *contigs* como *dataset*.
Fonte: O autor (2014)

TABELA 11: RESULTADOS GAGE

	R. sphaeroides			*S. aureus*		
Montadores	# *gaps*	# f/v	% acerto	# *gaps*	# f/v	% acerto
ABySS	323	231/221	96%	69	40/38	95%
ABySS2	292	245/179	73%	35	30/24	80%
ALLPATHS-LG	170	32/26	81%	48	34/26	76%
Bambus2	85	25/20	80%	99	67/59	88%
CABOG	193	47/45	96%	-	-	-
MSR-CA	356	92/76	83%	81	58/41	70%
SGA	938	277/259	95%	654	397/392	99%
SOAPdenovo	38	17/15	88%	9	3/3	100%
Velvet	427	330/247	75%	128	89/70	79%

A coluna "# *gaps*" representa o total de *gaps* nos *scaffolds* antes do fechamento. A coluna "# f/v" representa o total de *gaps* fechados pelo FGAP e validados localmente, respectivamente. A montagem do programa CABOG não foi disponibilizada no projeto GAGE.
FONTE: O Autor (2014)

As FIGURAS 21 e 22 mostram a variação nas métricas globais de cada montagem após o fechamento de *gaps* pelo FGAP. Nota-se uma melhora geral no número de genes completos identificados, diminuindo em proporção similar os genes parciais, mostrando que o FGAP permite a resolução de genes quebrados que estão em regiões de *gaps*, assim como identificar novos. O número de falhas locais também é visivelmente menor após o fechamento de *gaps*, resolvendo regiões antes não identificadas e *gaps* curtos. No geral, um pequeno número de mismatches são inseridos nas

montagens, possivelmente por regiões erradas ou com baixa qualidade no *dataset*. Os *gaps* fechados incorretamente também influênciam neste aumento. O número de indels e a taxa de duplicação melhoram após a aplicação do FGAP utilizando montagens alternativas. Nota-se que algumas montagens têm um pior desempenho em diversas métricas quando submetidas ao FGAP, como o Velvet no organismo *R. sphaeroides*, indicando montagem ou processo de *scaffolding* pouco confiáveis. A lista completa de métricas da validação global estão no APÊNDICE C - Resultados complementares.

3.3.2 GAGE - Cromossomo Humano 14

O projeto GAGE também disponibilizou dados de organismos eucariotos. Utilizamos as montagens do cromossomo humano 14. Submetemos este conjunto no FGAP da mesma maneira que os testes supracitados dos organismos procariotos. Os resultados estão na TABELA 12.

TABELA 12: RESULTADOS GAGE - CROMOSSOMO HUMANO 14

	Cromossomo Humano 14			
Montadores	# *gaps*	# fechados	% fechamento	Tempo de execução
ABySS	1061	456	43%	28m
ABySS2	2820	1750	62%	28h33m
ALLPATHS-LG	4307	1689	39%	42h04m
Bambus2	11809	382	3%	18h46m
CABOG	3043	639	21%	31h01m
MSR-CA	30622	19415	63%	50h50m
SGA	21459	11194	52%	14h23m
SOAPdenovo	8544	2090	25%	12h12m
Velvet	51567	11412	22%	20h11m

A coluna "# *gaps*" representa o total de *gaps* nos *scaffolds* antes do fechamento. A coluna "# fechados" representa o total de *gaps* fechados pelo FGAP.
FONTE: O Autor (2014)

O conjunto de dados do cromossomo humano 14 teve redução média de 37% nos *gaps*, considerando todas as montagens. Devido a limitações técnicas e extenso tempo de execução do programa MUMMER em genomas complexos, não foi possível finalizar a validação local para este conjunto de dados. A FIGURA 23 mostra a variação das métricas globais de cada montagem após o fechamento de *gaps*.

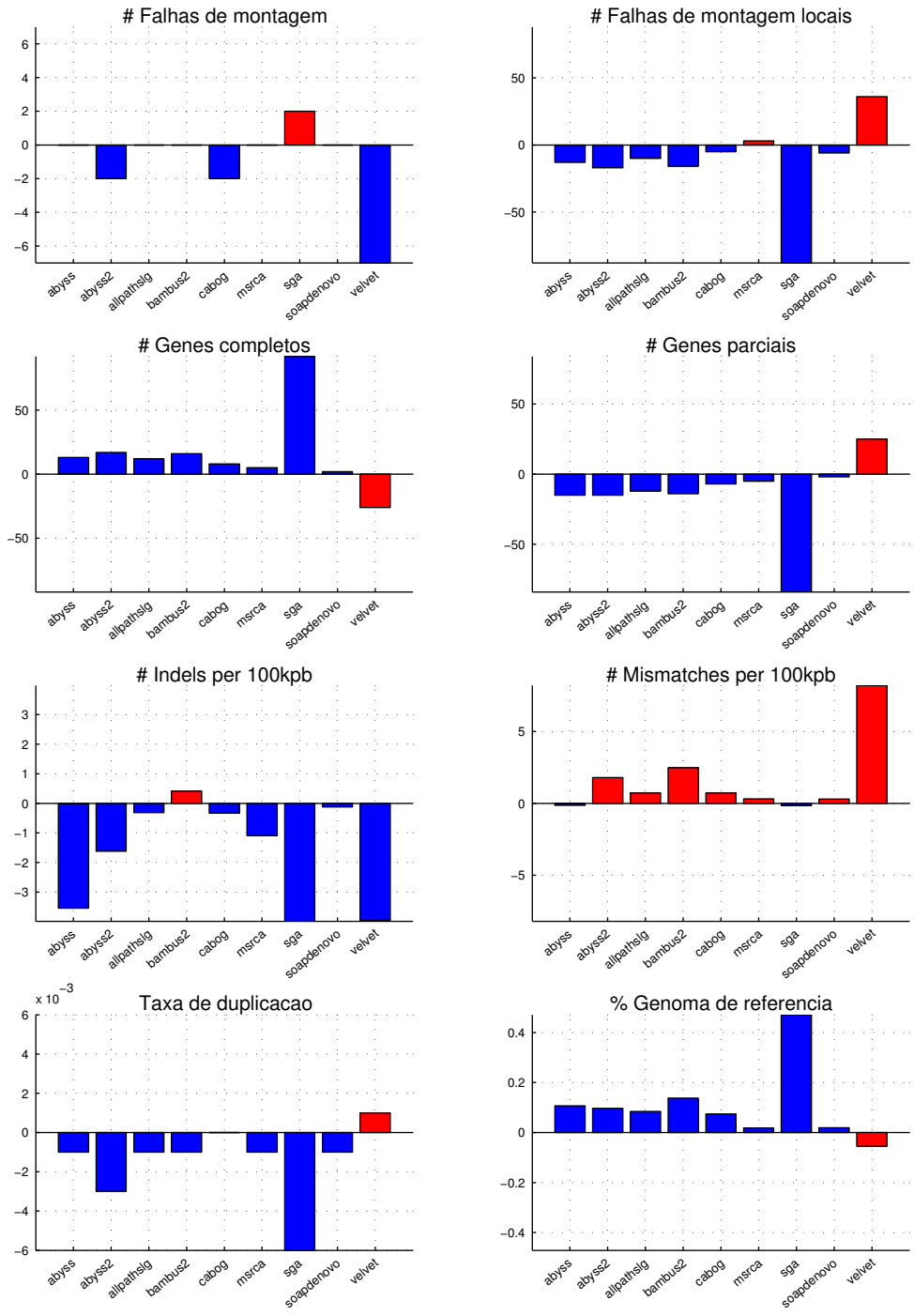


FIGURA 21: VARIAÇÃO DAS MÉTRICAS - *R. sphaeroides* - GAGE

Cada gráfico representa a variação de uma métrica. Para cada programa descrito no eixo x é apresentada a diferença de valor entre a métrica da montagem anterior e posterior ao fechamento de gaps. Barras azuis representam melhorias, tanto em métricas que tenham valores positivos (ex: genes completos) ou negativos (ex: genes parciais), e barras vermelhas representam piora na métrica.
Fonte: O autor (2014)

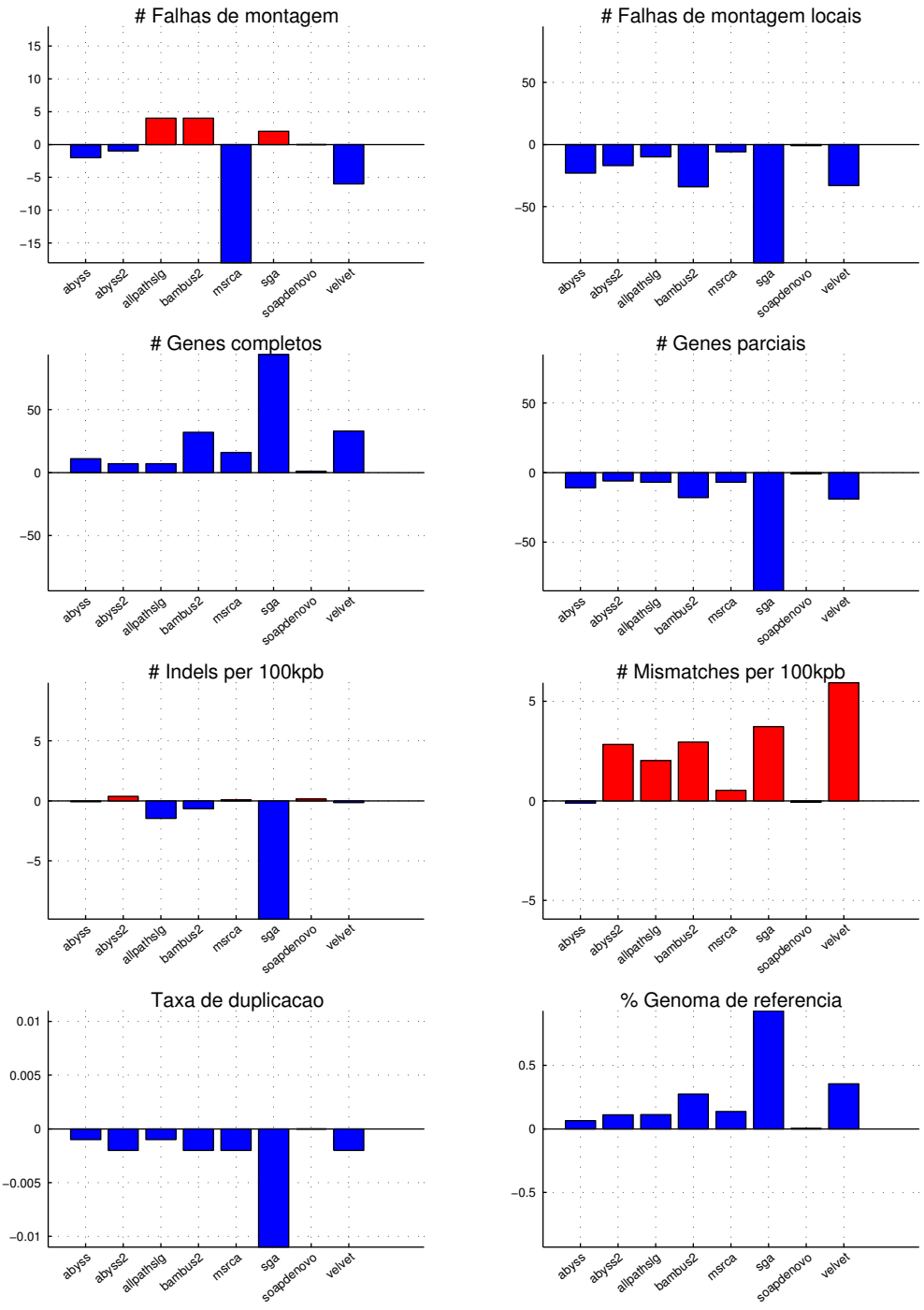


FIGURA 22: VARIAÇÃO DAS MÉTRICAS - *S. aureus* - GAGE

Cada gráfico representa a variação de uma métrica. Para cada programa descrito no eixo x é apresentada a diferença de valor entre a métrica da montagem anterior e posterior ao fechamento de gaps. Barras azuis representam melhorias, tanto em métricas que tenham valores positivos (ex: genes completos) ou negativos (ex: genes parciais), e barras vermelhas representam piora na métrica.
Fonte: O autor (2014)

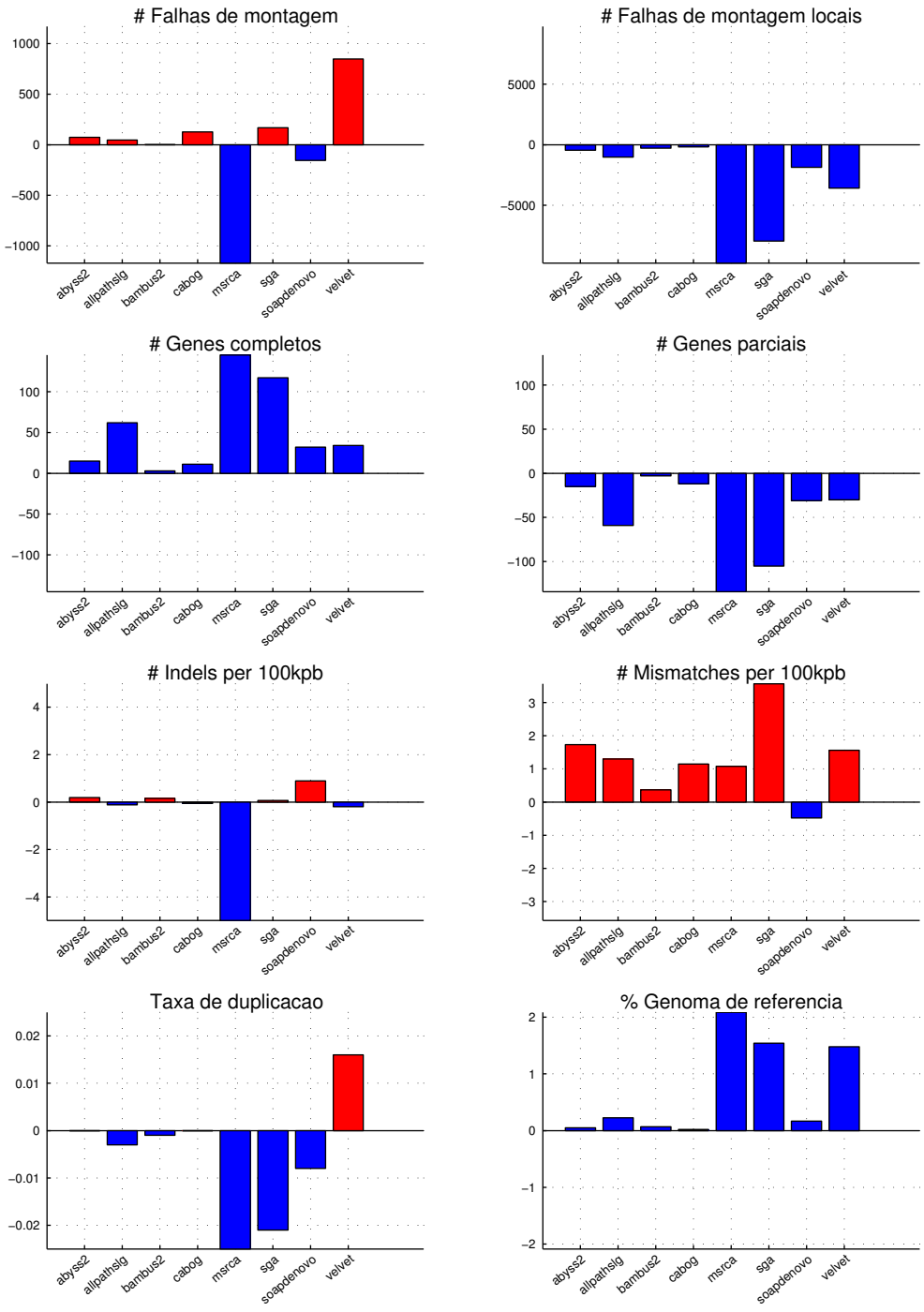


FIGURA 23: VARIAÇÃO DAS MÉTRICAS - Cromossomo humano 14 - GAGE

Cada gráfico representa a variação de uma métrica. Para cada programa descrito no eixo x é apresentada a diferença de valor entre a métrica da montagem anterior e posterior ao fechamento de gaps. Barras azuis representam melhorias, tanto em métricas que tenham valores positivos (ex: genes completos) ou negativos (ex: genes parciais), e barras vermelhas representam piora na métrica.
Fonte: O autor (2014)

3.3.3 GAGE-B

As montagens do projeto GAGE-B utilizadas nos testes com o FGAP foram geradas a partir de 8 programas de montagem (detalhes das montagens no APÊNDICE A - Dados GAGE e GAGE-B):

- ABySS
- CABOG
- Mira
- MSR-CA
- SGA
- SOAPdenovo
- Spades
- Velvet

Os organismos selecionados no projeto GAGE-B foram as bactérias: *M. abscessus*, *R. sphaeroides* e *V. cholerae*. Diferentemente do projeto GAGE, no GAGE-B foram feitas montagens a partir de duas corridas de sequenciamento: MiSeq e HiSeq, ambos da Illumina, para cada organismo. A partir destes conjuntos foi possível comparar a complementariedade de montagens oriundas de tecnologias e sequenciamentos distintos. Os dados foram disponibilizados em *contigs* e *scaffolds*. As montagens geradas por este projeto são visivelmente de maior qualidade do que as montagens produzidas no projeto GAGE. Isto se deve a rápida evolução tanto dos equipamentos de sequenciamento quanto dos algoritmos de montagem, dado que o GAGE-B foi publicado mais de 2 anos depois do GAGE, além de utilizar bibliotecas e equipamentos diferentes. Mesmo com sequências genômicas de maior qualidade e menor quantidade de *gaps* o FGAP consegue melhorar substancialmente as montagens.

A TABELA 13 mostra os resultados do FGAP nos dados GAGE-B, executados da seguinte maneira: as montagens em *scaffolds* feitas com os dados HiSeq foram utilizados como *draft* por terem, em média, NGA50 maior, que é o valor N50 corrigido

(TABELA 7). Para cada montagem HiSeq (*scaffold*) foi utilizado um *dataset* da montagem MiSeq (*contigs*) proveniente do mesmo programa de montagem (FIGURA 24). Assim foi possível verificar o benefício de mais de uma corrida de sequenciamento, sem os custos de execução de diversos programas de montagem.

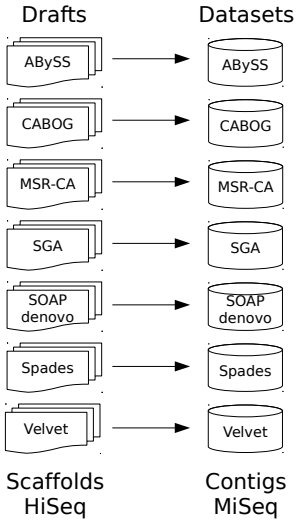


FIGURA 24: TESTE GAGE-B

Cada montagem em *scaffold* HiSeq foi submetida ao FGAP utilizando a montagem em *contigs* MiSeq do mesmo programa como *dataset*.
Fonte: O autor (2014)

TABELA 13: RESULTADOS GAGE-B

	M. abscessus			*R. sphaeroides*			*V. cholerae*		
Montadores	# *gaps*	# f/v	% acerto	# *gaps*	# f/v	% acerto	# *gaps*	# f/v	% acerto
ABySS	84	49/49	100%	41	9/7	78%	160	127/103	81%
CABOG	20	9/9	100%	219	88/88	100%	19	5/5	100%
MSR-CA	7	1/1	100%	5	0/0	-	3	1/0	0%
SGA	0	-	-	0	-	-	0	-	-
SOAPdenovo	7	2/2	100%	207	31/30	97%	27	6/3	50%
Spades	0	-	-	763	388/360	93%	0	-	-
Velvet	81	68/68	100%	363	316/316	100%	143	129/129	100%

A coluna "# *gaps*" representa o total de *gaps* nos *scaffolds* antes do fechamento. A coluna "# f/v" representa o total de *gaps* fechados e validados localmente utilizando o FGAP, respectivamente. O montador Mira não foi utilizado neste teste pois gerou apenas arquivos de *contigs*.
FONTE: O Autor (2014)

As montagens de corridas diferenciadas se complementam com alta taxa de acerto pelo FGAP, apesar de fechar um número menor de *gaps*, certamente influenciada pela melhor qualidade de dados gerados pelos equipamentos e montadores. Foram fechados 57% dos *gaps*, considerando todas as montagens, tendo 95% deste conjunto validado localmente, em média. Todos os *gaps* fechados da montagem do

organismo *M. abscessus* foram validados, sem inserir nenhum erro. Os dados SOAPdenovo tem um número reduzidos de *gaps* pois já foram disponibilizadas depois de ter a montagem finalizada pelo GapCloser, que faz parte do pacote SOAPdenovo2. Mesmo assim foi possível finalizar alguns *gaps* destas sequências. Algumas montagens não possuiam *gaps* nos *scaffolds* produzidos.

3.3.4 GAGE e GAGE-B - *R. sphaeroides*

A mesma espécie da bactéria *R. sphaeroides* foi disponibilizada em ambos os projetos GAGE e GAGE-B, gerando mais dados que os demais organismos. Foi possível, através destes conjuntos, testar a eficiência do FGAP quando disponíveis mais de 2 corridas de sequenciamento e ao mesmo tempo, diversas montagens. Para isto, as montagens em *scaffolds* do organismo *R. sphaeroides* do projeto GAGE foram utilizadas como *draft*. As montagens em *contigs* do projeto GAGE-B do mesmo organismo foram utilizados como *datasets*, gerando um total de 16 arquivos, 8 provenientes dos dados Illumina HiSeq e 8 do MiSeq (FIGURA 25). O resultado das montagens GAGE fechadas através do FGAP com os dados GAGE-B estão detalhadas na TABELA 14.

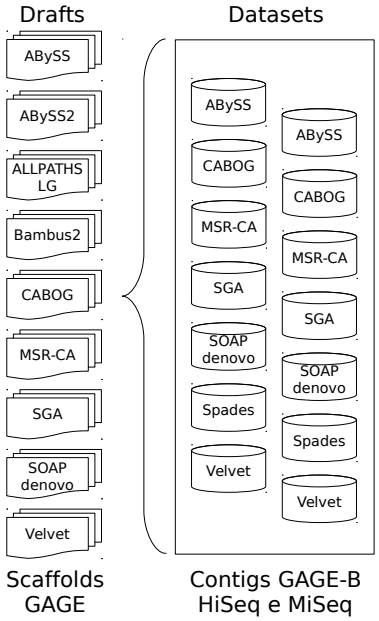


FIGURA 25: TESTE GAGE e GAGE-B

Cada montagem em *scaffold* GAGE foi submetida ao FGAP utilizando as montagens em *contigs* MiSeq e HiSeq do projeto GAGE-B como *dataset*.
Fonte: O autor (2014)

TABELA 14: RESULTADOS GAGE E GAGE-B - *R. sphaeroides*

	R. sphaeroides		
Montadores	# *gaps*	# f/v	% acerto
ABySS	323	236/234	99%
ABySS2	292	250/248	99%
ALLPATHS-LG	170	122/121	99%
Bambus2	85	31/31	100%
CABOG	193	113/110	97%
MSR-CA	356	160/153	96%
SGA	938	277/276	99%
SOAPdenovo	38	26/26	100%
Velvet	437	344/342	99%

A coluna "# *gaps*" representa o total de *gaps* nos *scaffolds* antes do fechamento. A coluna "# f/v" representa o total de *gaps* fechados e validados localmente utilizando o FGAP, respectivamente.
FONTE: O Autor (2014)

Este teste foi o que gerou os melhores resultados quando considerada a evolução das montagens e a média de de *gaps* fechados/validados, certamente pela grande variedade de dados e de montagens disponíveis. Dos 2832 *gaps* de todas as montagens, 1559 foram fechados (55%), com uma taxa de acerto de 98%. Esta aplicação do FGAP mostra-se promissora para projetos com diversas corridas de sequenciamento de diferentes tecnologias. A FIGURA 26 mostra a variação nos parâmetros globais após o fechamento de *gaps* pelo FGAP. Em geral, todas as métricas apresentaram melhorias, comprovando a capacidade do FGAP de fechar *gaps* corretamente, corrigindo regiões gênicas, aproximando a montagem do tamanho da sequência de referência e diminuindo o número de falhas na montagem, utilizando dados reaproveitados. A lista completa de métricas da validação global estão no APÊNDICE C - Resultados complementares.

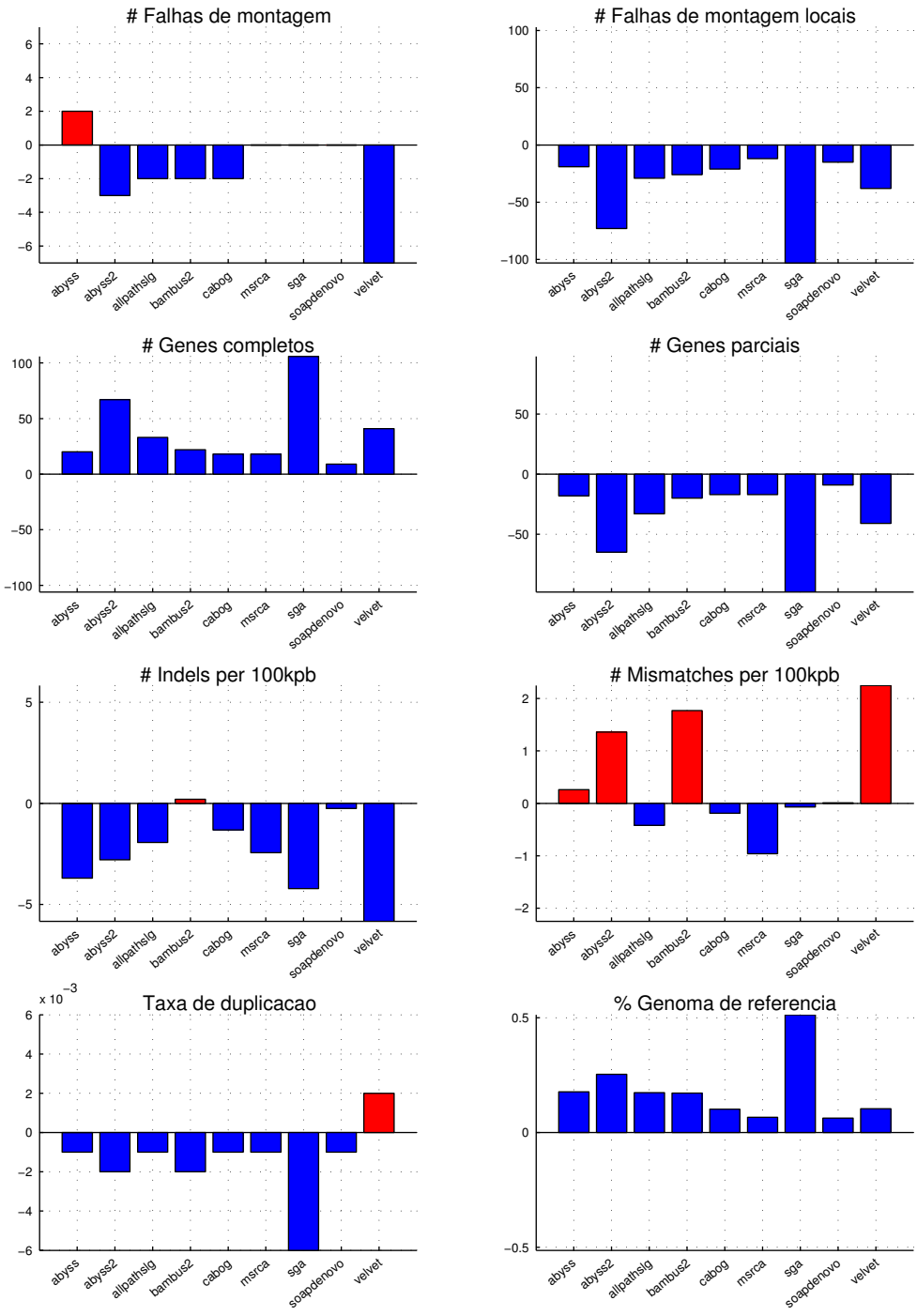


FIGURA 26: VARIAÇÃO DAS MÉTRICAS - *R. sphaeroides* - GAGE e GAGE-B

Cada gráfico representa a variação de uma métrica. Para cada programa descrito no eixo x é apresentada a diferença de valor entre a métrica da montagem anterior e posterior ao fechamento de gaps. Barras azuis representam melhorias, tanto em métricas que tenham valores positivos (ex: genes completos) ou negativos (ex: genes parciais), e barras vermelhas representam piora na métrica.
FONTE: O autor (2014)

4 CONCLUSÃO

Neste trabalho foi apresentado o programa FGAP, um novo método para finalização de montagens genômicas, que visa integrar diversos conjuntos de dados para realizar o fechamento de *gaps*. Mostramos que através do FGAP é possível diminuir o número de *gaps* de uma sequência genômica recém-montada, melhorando consideravelmente diversas características e métricas, sempre reutilizando dados, seja variando os programas de montagem, utilizando *reads* longos ou diversas corridas de sequenciamento. Todos os *gaps* fechados foram verificados localmente e o programa teve taxa de 93% de acerto na média dos casos dos projetos GAGE e GAGE-B.

O programa tem a capacidade de finalizar *gaps* propostos em *scaffolds*, tanto de *gaps* positivos, quando há um número real de bases não identificadas, quanto de *gaps* negativos, quando as pontas dos *contigs* estão duplicadas e possuem sobreposição, abrangendo a grande maioria de *gaps* reais propostos nas montagens atuais. Testes controlados comprovaram a precisão do programa. Após a execução do FGAP em uma sequência genômica com *gaps* simulados foi possível obter a sequência completa em seu estado original.

Devido ao método de escolha dos candidatos para fechamento de *gaps* do FGAP, a aplicação de *gaps* zero e negativos não se mostrou vantajosa quando executada em uma primeira etapa de finalização. Isto porque alguns finais de *contigs* tendem a se sobrepor, induzindo a falsos *gaps* negativos, quando na verdade, são regiões de repetição reais. Os *gaps* zero também acabaram abaixando a taxa de acerto do programa pois, em sua maioria, reproduziam erros dos *datasets* na montagem principal. Estes tipos de *gaps* foram corretamente fechados no caso controlado, mostrando que o FGAP tem a capacidade de identificar corretamente finais de *contigs* sobrepostos e *gaps* zero quando estão corretos no *dataset*, porém é recomendável rodar o programa com estes parâmetros ativos apenas para casos específicos, onde a sobreposição de pontas já é conhecida ou em uma segunda rodada de fechamentos, onde os *gaps* positivos já foram resolvidos.

O FGAP, quando comparado com programas de mesma finalidade disponíveis, mostrou-se superior nos resultados e no tempo de execução. Ele é uma boa alternativa para diversas situações, pois não tem seu uso restrito a *reads* pareados, e

consegue reaproveitar conjuntos de dados que muitas vezes ficam subutilizados nos processos de montagem. É também o único programa de fechamento de *gaps* que tem suporte direto aos *reads* de terceira geração que, através dos testes realizados, mostrou-se ser um dos melhores métodos para a finalização da montagem do genoma proposto da bactéria *E. coli*, fechando quase todos os seus *gaps* sem a inserção de nenhum erro.

Quando diversos montadores foram utilizados em um mesmo conjunto de *reads*, o FGAP é capaz de absorver dados complementares e fechar *gaps* na montagem principal. O mesmo acontece para conjuntos de diversas tecnologias de sequenciamento. Quanto mais dados disponíveis, melhores os resultados obtidos. Testes realizados no organismo *R. sphaeroides* mostram que, corridas geradas com equipamentos e programas mais recentes têm grande capacidade de gerar dados complementares para montagens não finalizadas.

A taxa de acerto do FGAP é influenciada pela qualidade dos dados disponíveis nos *datasets*, como mostram os testes GAGE e GAGE-B: quando utilizados apenas dados do projeto GAGE, com tecnologias de sequenciamento e programas de montagens mais antigos, a taxa de acerto manteve-se em 86% na média dos casos. Já nos testes realizados utilizando *datasets* mais recentes e de melhor qualidade do projeto GAGE-B, a taxa de acerto foi de 97%.

O FGAP foi testado em diversas sequências de genomas procariotos. Porém testes preliminares realizados no cromossomo humano 14 mostram a sua funcionalidade para organismos eucariotos, sendo necessário um estudo aprofundado nos parâmetros a serem utilizados para melhorar a taxa de acertos, assim como otimizar o algoritmo para reduzir o tempo de execução.

4.1 Trabalhos futuros

Foram apresentadas algumas maneiras de obter dados para fechar *gaps* através do FGAP. Estes conjuntos de dados são cruciais para um fechamento correto e, quanto maior e mais variado o conjunto, mais possibilidades de fechamento existirão. O FGAP não se limita aos conjuntos citados neste trabalho e diversas outras abordagens podem ser utilizadas para incorporar dados em uma sequência genômica

recém-montada. Algumas técnicas que potencialmente gerariam bons conjuntos para finalização são:

- União de *reads* pareados: trabalhos publicados (MAGOC; SALZBERG, 2011) (NADALIN; VEZZI; POLICRITI, 2012) demonstram a capacidade de gerar *reads* longos a partir da união de *reads* pareados, visando facilitar o processo de montagem. Estes dados poderiam ser aplicados como *dataset* em fechamentos com o FGAP.

- Integrador de *contigs*: dado a grande quantidade de *contigs* gerados por diversos programas de montagem, uma etapa de pré-seleção destes *contigs* poderia ser realizada, visando gerar um conjunto conciso de melhor qualidade para o fechamento de *gaps*, como descrito em (LIN; LIAO, 2013).

- Auxílio de organismo de referência: quando montagens de genoma de organismos próximos estão disponíveis, é possível gerar sequências consenso mapeando os *reads* do organismo desejado contra a(s) referência(s) (GNERRE et al., 2009), gerando um conjunto de *contigs* para ser utilizado como dataset no FGAP.

Quando o FGAP recebe um grande conjunto de dados com alto nível de redundância, como no caso de dados de vários programas de montagem do mesmo organismo como datasets, é comum cada *gap* possuir diversos pré-candidatos para o fechamento. Na versão atual, o FGAP seleciona o alinhamento que tenha o melhor escore, seguido pela melhor cobertura da *query*, identidade, e-value e tamanho do *contig* (no *dataset*). Porém em alguns casos, este método pode levar a escolha de um candidato não ideal, reduzindo a taxa de acerto do programa. Alguns testes preliminares com resultados promissores foram feitos aplicando redes neurais artificiais, treinando a rede em um conjunto de *gaps* conhecidos e validados, baseados em diversas características como escore, distância do *gap*, identidade, entre outros dados relevantes para o fechamento do *gap*. Esta rede visa identificar a melhor opção entre os pré-candidatos e selecionar o que tem a maior probabilidade de fechar o *gap* corretamente, que não é necessariamente o que tenha melhores escores. Esta técnica também poderia ser aplicada para identificar e evitar o fechamento de *gaps* falsos, em scaffolds erroneamente gerados. A geração de uma sequência consenso entre os pré-candidatos para fechar o *gap* também poderia ser utilizada como alternativa.

O FGAP foi o programa mais rápido quando comparado com outros programas similares, mas possui espaço para otimizações, tanto no código fonte quanto na linguagem utilizada. Linguagens que são otimizadas para trabalhar com arquivos, como C ou Perl, poderiam ter uma performance superior. Em organismos procariotos, o programa tem tempo de execução, em média, na casa de segundos/minutos em um notebook de configurações básicas. Porém quando executado para sequências maiores, como o de organismos eucariotos, fica com tempo de execução na casa de horas. Isto ocorre, em partes, devido ao tempo de execução do BLAST, que é o processo que demanda mais tempo do algoritmo. Uma alternativa ao BLAST poderia diminuir o tempo de execução para sequências genômicas maiores e mais complexas.

Por fim, testes preliminares mostram que o FGAP poderia ser acoplado a um algoritmo de scaffolding. Para isto, seriam necessário uma ou mais montagens de referência, onde os *contigs* (ou scaffolds já gerados) seriam reordenados e submetidos ao FGAP, em um processo iterativo, visando fechar um número maior de *gaps* muitas vezes não identificados em scaffolds iniciais.

REFERÊNCIAS

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

ALTSCHUL, S. F.; MADDEN, T. L.; SCHäFFER, A. A.; ZHANG, J.; ZHANG, Z.; MILLER, W.; LIPMAN, D. J. **Gapped BLAST and PSI-BLAST: A new generation of protein database search programs.** *Nucleic acids research*, v. 25, n. 17, p. 3389–402, set. 1997. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=146917>.

ASSEFA, S.; KEANE, T. M.; OTTO, T. D.; NEWBOLD, C.; BERRIMAN, M. **ABACAS: algorithm-based automatic contiguation of assembled sequences.** *Bioinformatics (Oxford, England)*, v. 25, n. 15, p. 1968–9, ago. 2009. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2712343>.

BASHIR, A.; KLAMMER, A. A.; ROBINS, W. P.; CHIN, C.-S.; WEBSTER, D.; PAXINOS, E.; HSU, D.; ASHBY, M.; WANG, S.; PELUSO, P.; SEBRA, R.; SORENSON, J.; BULLARD, J.; YEN, J.; VALDOVINO, M.; MOLLOVA, E.; LUONG, K.; LIN, S.; LAMAY, B.; JOSHI, A.; ROWE, L.; FRACE, M.; TARR, C. L.; TURNSEK, M.; DAVIS, B. M.; KASARSKIS, A.; MEKALANOS, J. J.; WALDOR, M. K.; SCHADT, E. E. **A hybrid approach for the automated finishing of bacterial genomes.** *Nature biotechnology*, v. 30, n. 7, p. 701–707, jul. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22750883>.

BENSON, D. A.; CAVANAUGH, M.; CLARK, K.; KARSCH-MIZRACHI, I.; LIPMAN, D. J.; OSTELL, J.; SAYERS, E. W. **GenBank.** *Nucleic acids research*, v. 41, n. Database issue, p. D36–42, jan. 2013. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3531190>.

BLATTNER, F.; PLUNKETT, G.; BLOCH, C.; PERNA, N. **The complete genome sequence of Escherichia coli K-12**. *Science*, v. 277, n. 5331, p. 1453–1462, set. 1997. Disponível em: <http://www.sciencemag.org/cgi/doi/10.1126/science.277.5331.1453>.

BOETZER, M.; HENKEL, C. V.; JANSEN, H. J.; BUTLER, D.; PIROVANO, W. **Scaffolding pre-assembled contigs using SSPACE.** *Bioinformatics (Oxford, England)*, v. 27, n. 4, p. 578–9, fev. 2011. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/21149342>.

BOETZER, M.; PIROVANO, W. **Toward almost closed genomes with GapFiller.** *Genome biology*, v. 13, n. 6, p. R56, jun. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22731987>.

BRADNAM, K. R.; FASS, J. N.; ALEXANDROV, A.; BARANAY, P.; BECHNER, M.; BIROL, I.; BOISVERT, S.; CHAPMAN, J. A.; CHAPUIS, G.; CHIKHI, R.; CHITSAZ, H.; CHOU, W.-C.; CORBEIL, J.; Del Fabbro, C.; DOCKING, T. R.; DURBIN, R.; EARL, D.; EMRICH, S.; FEDOTOV, P.; FONSECA, N. A.; GANAPATHY, G.; GIBBS,

R. A.; GNERRE, S.; GODZARIDIS, E.; GOLDSTEIN, S.; HAIMEL, M.; HALL, G.; HAUSSLER, D.; HIATT, J. B.; HO, I. Y.; HOWARD, J.; HUNT, M.; JACKMAN, S. D.; JAFFE, D. B.; JARVIS, E. D.; JIANG, H.; KAZAKOV, S.; KERSEY, P. J.; KITZMAN, J. O.; KNIGHT, J. R.; KOREN, S.; LAM, T.-W.; LAVENIER, D.; LAVIOLETTE, F.; LI, Y.; LI, Z.; LIU, B.; LIU, Y.; LUO, R.; MACCALLUM, I.; MACMANES, M. D.; MAILLET, N.; MELNIKOV, S.; NAQUIN, D.; NING, Z.; OTTO, T. D.; PATEN, B.; PAULO, O. S.; PHILLIPPY, A. M.; PINA-MARTINS, F.; PLACE, M.; PRZYBYLSKI, D.; QIN, X.; QU, C.; RIBEIRO, F. J.; RICHARDS, S.; ROKHSAR, D. S.; RUBY, J. G.; SCALABRIN, S.; SCHATZ, M. C.; SCHWARTZ, D. C.; SERGUSHICHEV, A.; SHARPE, T.; SHAW, T. I.; SHENDURE, J.; SHI, Y.; SIMPSON, J. T.; SONG, H.; TSAREV, F.; VEZZI, F.; VICEDOMINI, R.; VIEIRA, B. M.; WANG, J.; WORLEY, K. C.; YIN, S.; YIU, S.-M.; YUAN, J.; ZHANG, G.; ZHANG, H.; ZHOU, S.; KORF, I. F. **Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species.** *GigaScience*, v. 2, n. 1, p. 10, jan. 2013. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3844414>.

CGIAR. ***CGIAR Generation Challenge Programme's learning module***. 2014. Disponível em: <https://www.integratedbreeding.net/courses/genomics-and-comparative-genomics/www.generationcp.org/genomics/indexda85.html?page=1173>.

CHAIN, P. S. G.; GRAFHAM, D. V.; FULTON, R. S.; FITZGERALD, M. G.; HOSTETLER, J.; MUZNY, D.; ALI, J.; BIRREN, B.; BRUCE, D. C.; BUHAY, C.; COLE, J. R.; DING, Y.; DUGAN, S.; FIELD, D.; GARRITY, G. M.; GIBBS, R.; GRAVES, T.; HAN, C. S.; HARRISON, S. H.; HIGHLANDER, S.; HUGENHOLTZ, P.; KHOURI, H. M.; KODIRA, C. D.; KOLKER, E.; KYRPIDES, N. C.; LANG, D.; LAPIDUS, A.; MALFATTI, S. A.; MARKOWITZ, V.; METHA, T.; NELSON, K. E.; PARKHILL, J.; PITLUCK, S.; QIN, X.; READ, T. D.; SCHMUTZ, J.; SOZHAMANNAN, S.; STERK, P.; STRAUSBERG, R. L.; SUTTON, G.; THOMSON, N. R.; TIEDJE, J. M.; WEINSTOCK, G.; WOLLAM, A.; DETTER, J. C.; RESPONSIBILI, S.; SNAPE, J.; TIWARI, B.; SERVICE, S. D.; SANSONE, S. A.; QUACKENBUSH, J.; GILES, J.; LAU, F. **Genome Project Standards in a New Era of Sequencing**. *Science (New York, N.Y.)*, v. 326, n. October, p. 4–5, 2009.

CHIN, C.-S.; ALEXANDER, D. H.; MARKS, P.; KLAMMER, A. A.; DRAKE, J.; HEINER, C.; CLUM, A.; COPELAND, A.; HUDDLESTON, J.; EICHLER, E. E.; TURNER, S. W.; KORLACH, J. **Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data.** *Nature methods*, v. 10, n. 6, p. 563–9, jun. 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23644548>.

COCK, P. J. A.; FIELDS, C. J.; GOTO, N.; HEUER, M. L.; RICE, P. M. **The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants.** *Nucleic acids research*, v. 38, n. 6, p. 1767–71, abr. 2010. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2847217>.

COORDINATORS, N. R. **Database resources of the National Center for Biotechnology Information.** *Nucleic acids research*, v. 41, n. Database issue, p. D8–D20, jan. 2013. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3531099>.

DAYARIAN, A.; MICHAEL, T. P.; SENGUPTA, A. M. **SOPRA: Scaffolding algorithm for paired reads via statistical optimization.** *BMC bioinformatics*, v. 11, p. 345, jan. 2010. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2909219>.

DONMEZ, N.; BRUDNO, M. **SCARPA: scaffolding reads with practical algorithms.** *Bioinformatics (Oxford, England)*, v. 29, n. 4, p. 428–34, fev. 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23274213>.

EARL, D.; BRADNAM, K.; St John, J.; DARLING, A.; LIN, D.; FASS, J.; YU, H. O. K.; BUFFALO, V.; ZERBINO, D. R.; DIEKHANS, M.; NGUYEN, N.; ARIYARATNE, P. N.; SUNG, W.-K.; NING, Z.; HAIMEL, M.; SIMPSON, J. T.; FONSECA, N. A.; BIROL, Ä.; DOCKING, T. R.; HO, I. Y.; ROKHSAR, D. S.; CHIKHI, R.; LAVENIER, D.; CHAPUIS, G.; NAQUIN, D.; MAILLET, N.; SCHATZ, M. C.; KELLEY, D. R.; PHILLIPPY, A. M.; KOREN, S.; YANG, S.-P.; WU, W.; CHOU, W.-C.; SRIVASTAVA, A.; SHAW, T. I.; RUBY, J. G.; SKEWES-COX, P.; BETEGON, M.; DIMON, M. T.; SOLOVYEV, V.; SELEDTSOV, I.; KOSAREV, P.; VOROBYEV, D.; RAMIREZ-GONZALEZ, R.; LEGGETT, R.; MACLEAN, D.; XIA, F.; LUO, R.; LI, Z.; XIE, Y.; LIU, B.; GNERRE, S.; MACCALLUM, I.; PRZYBYLSKI, D.; RIBEIRO, F. J.; YIN, S.; SHARPE, T.; HALL, G.; KERSEY, P. J.; DURBIN, R.; JACKMAN, S. D.; CHAPMAN, J. A.; HUANG, X.; DERISI, J. L.; CACCAMO, M.; LI, Y.; JAFFE, D. B.; GREEN, R. E.; HAUSSLER, D.; KORF, I.; PATEN, B. **Assemblathon 1: a competitive assessment of de novo short read assembly methods.** *Genome research*, v. 21, n. 12, p. 2224–41, dez. 2011. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3227110>.

EWING, B.; HILLIER, L.; WENDL, M. C.; GREEN, P. **Base-Calling of Automated Sequencer Traces Using Phred . I . Accuracy Assessment**. *Genome research*, p. 175–185, 1998.

FLEISCHMANN, R. D.; ADAMS, M. D.; WHITE, O.; CLAYTON, R. A.; KIRKNESS, E. F.; KERLAVAGE, A. R.; BULT, C. J.; TOMB, J. F.; DOUGHERTY, B. A.; MERRICK, J. M. **Whole-genome random sequencing and assembly of Haemophilus influenzae Rd.** *Science (New York, N.Y.)*, v. 269, n. 5223, p. 496–512, jul. 1995. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/7542800>.

GAGE. ***GAGE - Genome Assembly Gold-standart Evaluation***. 2014. Disponível em: <http://gage.cbcb.umd.edu/results/index.html>.

GAGE-B. ***GAGE-B - Genome Assembly Gold-standart Evaluation***. 2014. Disponível em: <http://ccb.jhu.edu/gage_b/>.

GALARDINI, M.; BIONDI, E. G.; BAZZICALUPO, M.; MENGONI, A. **CONTIGuator: a bacterial genomes finishing tool for structural insights on draft genomes**. *Source code for biology and medicine*, v. 6, n. 1, p. 11, jan. 2011. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3133546>.

GAO, S.; BERTRAND, D.; NAGARAJAN, N. **FinIS : Improved in silico Finishing Using an Exact Quadratic Programming Formulation**. *LNBI*, v. 7534, p. 314–325, 2012.

GAO, S.; SUNG, W.-K.; NAGARAJAN, N. **Opera: reconstructing optimal genomic scaffolds with high-throughput paired-end sequences.** *Journal of computational biology : a journal of computational molecular cell biology*, v. 18, n. 11, p. 1681–91, nov. 2011. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3216105>.

GIBBS, R.; WEINSTOCK, G.; METZKER, M. **Genome sequence of the Brown Norway rat yields insights into mammalian evolution**. *Nature*, v. 428, n. 6982, p. 493–521, abr. 2004. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/15057822>.

GNERRE, S.; LANDER, E. S.; LINDBLAD-TOH, K.; JAFFE, D. B. **Assisted assembly: how to improve A de novo genome assembly by using related species.** *Genome biology*, v. 10, n. 8, p. R88, jan. 2009. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2745769>.

GNERRE, S.; MACCALLUM, I.; PRZYBYLSKI, D.; RIBEIRO, F. J.; BURTON, J. N.; WALKER, B. J.; SHARPE, T.; HALL, G.; SHEA, T. P.; SYKES, S.; BERLIN, A. M.; AIRD, D.; COSTELLO, M.; DAZA, R.; WILLIAMS, L.; NICOL, R.; GNIRKE, A.; NUSBAUM, C.; LANDER, E. S.; JAFFE, D. B. **High-quality draft assemblies of mammalian genomes from massively parallel sequence data.** *Proceedings of the National Academy of Sciences of the United States of America*, v. 108, n. 4, p. 1513–8, jan. 2011. Disponível em: <http://www.pnas.org/cgi/content/abstract/108/4/1513>.

GUREVICH, A.; SAVELIEV, V.; VYAHHI, N.; TESLER, G. **QUAST: Quality Assessment Tool for Genome Assemblies.** *Bioinformatics (Oxford, England)*, v. 29, n. 8, p. 1072–1075, fev. 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23422339>.

HERNANDEZ, D.; FRANCOIS, P.; FARINELLI, L.; OSTERAS, M.; SCHRENZEL, J. De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer. *Genome research*, v. 18, n. 5, p. 802–9, maio 2008. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2336802>.

HUSEMANN, P. ***Bioinformatic Approaches for Genome Finishing***. Tese (Doutorado) — Bielefeld University, Germany, 2011.

KIM, S.; LIAO, L.; TOMB, J. **A Probabilistic Approach to Sequence Assembly Validation**. *BIOKDD*, 2001. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.3311>.

KINGSFORD, C.; SCHATZ, M. C.; POP, M. **Assembly complexity of prokaryotic genomes using short reads.** *BMC bioinformatics*, v. 11, p. 21, jan. 2010. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2821320>.

KOREN, S.; SCHATZ, M. C.; WALENZ, B. P.; MARTIN, J.; HOWARD, J. T.; GANAPATHY, G.; WANG, Z.; RASKO, D. A.; MCCOMBIE, W. R.; JARVIS, E. D.; PHILLIPPY, A. M. **Hybrid error correction and de novo assembly of single-molecule sequencing reads.** *Nature biotechnology*, v. 30, n. 7, p. 693–700, jul. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22750884>.

KURTZ, S.; PHILLIPPY, A.; DELCHER, A. L.; SMOOT, M.; SHUMWAY, M.; ANTONESCU, C.; SALZBERG, S. L. **Versatile and open software for comparing large genomes.** *Genome biology*, v. 5, n. 2, p. R12, jan. 2004. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=395750>.

LANDER, E. S.; LINTON, L. M.; BIRREN, B.; NUSBAUM, C.; ZODY, M. C.; BALDWIN, J.; DEVON, K.; DEWAR, K.; DOYLE, M.; FITZHUGH, W.; FUNKE, R.; GAGE, D.; HARRIS, K.; HEAFORD, A.; HOWLAND, J.; KANN, L.; LEHOCZKY, J.; LEVINE, R.; MCEWAN, P.; MCKERNAN, K.; MELDRIM, J.; MESIROV, J. P.; MIRANDA, C.; MORRIS, W.; NAYLOR, J.; RAYMOND, C.; ROSETTI, M.; SANTOS, R.; SHERIDAN, A.; SOUGNEZ, C.; STANGE-THOMANN, N.; STOJANOVIC, N.; SUBRAMANIAN, A.; WYMAN, D.; ROGERS, J.; SULSTON, J.; AINSCOUGH, R.; BECK, S.; BENTLEY, D.; BURTON, J.; CLEE, C.; CARTER, N.; COULSON, A.; DEADMAN, R.; DELOUKAS, P.; DUNHAM, A.; DUNHAM, I.; DURBIN, R.; FRENCH, L.; GRAFHAM, D.; GREGORY, S.; HUBBARD, T.; HUMPHRAY, S.; HUNT, A.; JONES, M.; LLOYD, C.; MCMURRAY, A.; MATTHEWS, L.; MERCER, S.; MILNE, S.; MULLIKIN, J. C.; MUNGALL, A.; PLUMB, R.; ROSS, M.; SHOWNKEEN, R.; SIMS, S.; WATERSTON, R. H.; WILSON, R. K.; HILLIER, L. W.; MCPHERSON, J. D.; MARRA, M. A.; MARDIS, E. R.; FULTON, L. A.; CHINWALLA, A. T.; PEPIN, K. H.; GISH, W. R.; CHISSOE, S. L.; WENDL, M. C.; DELEHAUNTY, K. D.; MINER, T. L.; DELEHAUNTY, A.; KRAMER, J. B.; COOK, L. L.; FULTON, R. S.; JOHNSON, D. L.; MINX, P. J.; CLIFTON, S. W.; HAWKINS, T.; BRANSCOMB, E.; PREDKI, P.; RICHARDSON, P.; WENNING, S.; SLEZAK, T.; DOGGETT, N.; CHENG, J. F.; OLSEN, A.; LUCAS, S.; ELKIN, C.; UBERBACHER, E.; FRAZIER, M.; GIBBS, R. A.; MUZNY, D. M.; SCHERER, S. E.; BOUCK, J. B.; SODERGREN, E. J.; WORLEY, K. C.; RIVES, C. M.; GORRELL, J. H.; METZKER, M. L.; NAYLOR, S. L.; KUCHERLAPATI, R. S.; NELSON, D. L.; WEINSTOCK, G. M.; SAKAKI, Y.; FUJIYAMA, A.; HATTORI, M.; YADA, T.; TOYODA, A.; ITOH, T.; KAWAGOE, C.; WATANABE, H.; TOTOKI, Y.; TAYLOR, T.; WEISSENBACH, J.; HEILIG, R.; SAURIN, W.; ARTIGUENAVE, F.; BROTTIER, P.; BRULS, T.; PELLETIER, E.; ROBERT, C.; WINCKER, P.; SMITH, D. R.; DOUCETTE-STAMM, L.; RUBENFIELD, M.; WEINSTOCK, K.; LEE, H. M.; DUBOIS, J.; ROSENTHAL, A.; PLATZER, M.; NYAKATURA, G.; TAUDIEN, S.; RUMP, A.; YANG, H.; YU, J.; WANG, J.; HUANG, G.; GU, J.; HOOD, L.; ROWEN, L.; MADAN, A.; QIN, S.; DAVIS, R. W.; FEDERSPIEL, N. A.; ABOLA, A. P.; PROCTOR, M. J.; MYERS, R. M.; SCHMUTZ, J.; DICKSON, M.; GRIMWOOD, J.; COX, D. R.; OLSON, M. V.; KAUL, R.; RAYMOND, C.; SHIMIZU, N.; KAWASAKI, K.; MINOSHIMA, S.; EVANS, G. A.; ATHANASIOU, M.; SCHULTZ, R.; ROE, B. A.; CHEN, F.; PAN, H.; RAMSER, J.; LEHRACH, H.; REINHARDT, R.; MCCOMBIE, W. R.; BASTIDE, M. de la; DEDHIA, N.; BLöCKER, H.; HORNISCHER, K.; NORDSIEK, G.; AGARWALA, R.; ARAVIND, L.; BAILEY, J. A.; BATEMAN, A.; BATZOGLOU, S.; BIRNEY, E.; BORK, P.; BROWN, D. G.; BURGE, C. B.; CERUTTI, L.; CHEN, H. C.; CHURCH, D.; CLAMP, M.; COPLEY, R. R.; DOERKS, T.; EDDY, S. R.; EICHLER, E. E.; FUREY, T. S.; GALAGAN, J.; GILBERT, J. G.; HARMON, C.; HAYASHIZAKI, Y.; HAUSSLER, D.; HERMJAKOB, H.; HOKAMP, K.; JANG, W.; JOHNSON, L. S.; JONES, T. A.; KASIF, S.; KASPRYZK, A.; KENNEDY, S.; KENT, W. J.; KITTS, P.; KOONIN, E. V.; KORF, I.; KULP, D.; LANCET, D.; LOWE, T. M.; MCLYSAGHT, A.; MIKKELSEN, T.; MORAN, J. V.; MULDER, N.; POLLARA, V. J.; PONTING, C. P.; SCHULER, G.; SCHULTZ, J.; SLATER, G.; SMIT, A. F.; STUPKA, E.; SZUSTAKOWSKI, J.; THIERRY-MIEG, D.; THIERRY-MIEG, J.; WAGNER, L.; WALLIS, J.; WHEELER, R.; WILLIAMS, A.; WOLF, Y. I.; WOLFE, K. H.; YANG, S. P.; YEH, R. F.;

COLLINS, F.; GUYER, M. S.; PETERSON, J.; FELSENFELD, A.; WETTERSTRAND, K. A.; PATRINOS, A.; MORGAN, M. J.; JONG, P. de; CATANESE, J. J.; OSOEGAWA, K.; SHIZUYA, H.; CHOI, S.; CHEN, Y. J.; SZUSTAKOWKI, J. **Initial sequencing and analysis of the human genome.** *Nature*, v. 409, n. 6822, p. 860–921, fev. 2001. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/11237011>.

LIN, S.-H.; LIAO, Y.-C. **CISA: contig integrator for sequence assembly of bacterial genomes.** *PloS one*, v. 8, n. 3, p. e60843, jan. 2013. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3610655>.

LOMAN, N. J.; CONSTANTINIDOU, C.; CHAN, J. Z. M.; HALACHEV, M.; SERGEANT, M.; PENN, C. W.; ROBINSON, E. R.; PALLEN, M. J. **High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity.** *Nature reviews. Microbiology*, v. 10, n. 9, p. 599–606, set. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22864262>.

LUO, R.; LIU, B.; XIE, Y.; LI, Z.; HUANG, W.; YUAN, J.; HE, G.; CHEN, Y.; PAN, Q.; LIU, Y.; TANG, J.; WU, G.; ZHANG, H.; SHI, Y.; LIU, Y.; YU, C.; WANG, B.; LU, Y.; HAN, C.; CHEUNG, D. W.; YIU, S.-M.; PENG, S.; XIAOQIAN, Z.; LIU, G.; LIAO, X.; LI, Y.; YANG, H.; WANG, J.; LAM, T.-W.; WANG, J. **SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.** *GigaScience*, v. 1, n. 1, p. 18, jan. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3626529>.

MAGOC, T.; PABINGER, S.; CANZAR, S.; LIU, X.; SU, Q.; PUIU, D.; TALLON, L. J.; SALZBERG, S. L. **GAGE-B: an evaluation of genome assemblers for bacterial organisms.** *Bioinformatics (Oxford, England)*, v. 29, n. 14, p. 1718–25, jul. 2013. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3702249>.

MAGOC, T.; SALZBERG, S. L. **FLASH: fast length adjustment of short reads to improve genome assemblies.** *Bioinformatics (Oxford, England)*, v. 27, n. 21, p. 2957–63, nov. 2011. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3198573>.

MARDIS, E. R. **A decade's perspective on DNA sequencing technology.** *Nature*, v. 470, n. 7333, p. 198–203, fev. 2011. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/21307932>.

MARDIS, E. R. **Next-generation sequencing platforms.** *Annual review of analytical chemistry (Palo Alto, Calif.)*, v. 6, n. 1, p. 287–303, jun. 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23560931>.

MATLAB. ***MATLAB R2012a***. The MathWorks Inc., 2013. Disponível em: <http://www.mathworks.com/products/matlab/>.

MAXAM, A. M.; GILBERT, W. **A new method for sequencing DNA**. *Proceedings of the National Academy of Sciences of the United States of America*, v. 74, p. 560–564, jan. 1977. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/1422074>.

METZKER, M. L. **Sequencing technologies - the next generation.** *Nature reviews. Genetics*, v. 11, n. 1, p. 31–46, jan. 2010. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/19997069>.

NADALIN, F.; VEZZI, F.; POLICRITI, A. **GapFiller: a de novo assembly approach to fill the gap within paired reads.** *BMC bioinformatics*, v. 13 Suppl 1, n. Suppl 14, p. S8, jan. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3439727>.

NAGARAJAN, N.; COOK, C.; Di Bonaventura, M.; GE, H.; RICHARDS, A.; BISHOP-LILLY, K. A.; DESALLE, R.; READ, T. D.; POP, M. **Finishing genomes with limited resources: lessons from an ensemble of microbial genomes.** *BMC genomics*, v. 11, p. 242, jan. 2010. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2864248>.

NAGARAJAN, N.; POP, M. **Sequence assembly demystified**. *Nature reviews. Genetics*, v. 14, n. 3, p. 157–67, jan. 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23358380>.

Octave community. ***GNU Octave 3.6.2***. 2014. Disponível em: <www.gnu.org/software/octave/>.

OUZOUNIS, C. a. **Rise and demise of bioinformatics? Promise and progress.** *PLoS computational biology*, v. 8, n. 4, p. e1002487, abr. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3343106>.

PAGANI, I.; LIOLIOS, K.; JANSSON, J.; CHEN, I.-M. A.; SMIRNOVA, T.; NOSRAT, B.; MARKOWITZ, V. M.; KYRPIDES, N. C. **The Genomes OnLine Database (GOLD) v.4: status of genomic and metagenomic projects and their associated metadata.** *Nucleic acids research*, v. 40, n. Database issue, p. D571–9, jan. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3245063>.

PEARSON, W. R.; LIPMAN, D. J. **Improved tools for biological sequence comparison.** *Proceedings of the National Academy of Sciences of the United States of America*, v. 85, n. 8, p. 2444–8, abr. 1988. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=280013>.

PEVZNER, P. A.; TANG, H.; WATERMAN, M. S. **An Eulerian path approach to DNA fragment assembly.** *Proceedings of the National Academy of Sciences of the United States of America*, v. 98, n. 17, p. 9748–9753, 2001. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/11504945>.

POP, M. **Genome assembly reborn: recent computational challenges.** *Briefings in bioinformatics*, v. 10, n. 4, p. 354–66, jul. 2009. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2691937>.

POP, M.; KOSACK, D. S.; SALZBERG, S. L. **Hierarchical scaffolding with Bambus.** *Genome research*, v. 14, n. 1, p. 149–59, jan. 2004. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=314292>.

RANK, D.; BAYBAYAN, P.; BETTMAN, B.; BIBILLO, A.; BJORNSON, K.; CHAUDHURI, B.; CHRISTIANS, F.; CICERO, R.; CLARK, S.; DALAL, R.; DIXON, J.; FOQUET, M.; GAERTNER, A.; HARDENBOL, P.; HEINER, C.; HESTER, K.; HOLDEN, D.; KEARNS, G.; KONG, X.; KUSE, R.; LACROIX, Y.; LIN, S.; LUNDQUIST, P.; MA, C.; MARKS, P.; MAXHAM, M.; MURPHY, D.; PARK, I.; PHAM, T.; PHILLIPS, M.; ROY, J.; SEBRA, R.; SHEN, G.; SORENSON, J.; TOMANEY, A.; TRAVERS, K.; TRULSON, M.; VIECELI, J.; WEGENER, J.; WU, D.; YANG, A.; ZACCARIN, D.; ZHAO, P.; ZHONG, F.; KORLACH, J.; TURNER, S. **Real-Time DNA Sequencing from Single Polymerase Molecules**. v. 323, n. January, p. 133–138, 2009. Disponível em: <http://www.sciencemag.org/content/323/5910/133.abstract>.

RIBEIRO, F. J.; PRZYBYLSKI, D.; YIN, S.; SHARPE, T.; GNERRE, S.; ABOUELLEIL, A.; BERLIN, A. M.; MONTMAYEUR, A.; SHEA, T. P.; WALKER, B. J.; YOUNG, S. K.; RUSS, C.; NUSBAUM, C.; MACCALLUM, I.; JAFFE, D. B. **Finished bacterial genomes from shotgun sequence data.** *Genome research*, v. 22, n. 11, p. 2270–7, nov. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3483556>.

RICKER, N.; QIAN, H.; FULTHORPE, R. **The limitations of draft assemblies for understanding prokaryotic adaptation and evolution.** *Genomics*, jun. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22750556>.

RISSMAN, A. I.; MAU, B.; BIEHL, B. S.; DARLING, A. E.; GLASNER, J. D.; PERNA, N. T. **Reordering contigs of draft genomes using the Mauve aligner.** *Bioinformatics (Oxford, England)*, v. 25, n. 16, p. 2071–3, ago. 2009. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2723005>.

SAHLIN, K.; STREET, N.; LUNDEBERG, J.; ARVESTAD, L. **Improved gap size estimation for scaffolding algorithms.** *Bioinformatics (Oxford, England)*, v. 28, n. 17, p. 2215–22, set. 2012. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22923455>.

SALZBERG, S. L.; PHILLIPPY, A. M.; ZIMIN, A.; PUIU, D.; MAGOC, T.; KOREN, S.; TREANGEN, T. J.; SCHATZ, M. C.; DELCHER, A. L.; ROBERTS, M.; MARÇAIS, G.; POP, M.; YORKE, J. a. **GAGE: A critical evaluation of genome assemblies and assembly algorithms.** *Genome research*, v. 22, n. 3, p. 557–67, mar. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3290791>.

SANGER, F.; NICKLEN, S.; COULSON, A. R. **DNA sequencing with chain-terminating inhibitors**. *Proc. Natl. Acad. Sci. USA*, v. 74, n. 12, p. 5463–5467, 1977. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/1422003>.

SCHADT, E. E.; TURNER, S.; KASARSKIS, A. **A window into third-generation sequencing.** *Human molecular genetics*, v. 19, n. R2, p. R227–40, out. 2010. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/20858600>.

SIMPSON, J. T.; DURBIN, R. **Efficient de novo assembly of large genomes using compressed data structures.** *Genome research*, v. 22, n. 3, p. 549–56, mar. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3290790>.

TAYLOR, L. ***PHAST (PHAGE ASSEMBLY SUITE AND TUTORIAL): A WEB-BASED GENOME ASSEMBLY TEACHING TOOL***. 2012. Undergraduate Thesis in Computational Biology at Davidson College. Disponível em: <http://gcat.davidson.edu/phast/>.

TREANGEN, T. J.; SALZBERG, S. L. **Repetitive DNA and next-generation sequencing: computational challenges and solutions.** *Nature reviews. Genetics*, v. 13, n. 1, p. 36–46, jan. 2012. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3324860>.

TSAI, I. J.; OTTO, T. D.; BERRIMAN, M. **Improving draft assemblies by iterative mapping and assembly of short reads to eliminate gaps.** *Genome biology*, v. 11, n. 4, p. R41, jan. 2010. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2884544>.

VENTER, J. C.; ADAMS, M. D.; MYERS, E. W.; LI, P. W.; MURAL, R. J.; SUTTON, G. G.; SMITH, H. O.; YANDELL, M.; EVANS, C. A.; HOLT, R. A.; GOCAYNE, J. D.; AMANATIDES, P.; BALLEW, R. M.; HUSON, D. H.; WORTMAN, J. R.; ZHANG, Q.; KODIRA, C. D.; ZHENG, X. H.; CHEN, L.; SKUPSKI, M.; SUBRAMANIAN, G.; THOMAS, P. D.; ZHANG, J.; Gabor Miklos, G. L.; NELSON, C.; BRODER, S.; CLARK, A. G.; NADEAU, J.; MCKUSICK, V. A.; ZINDER, N.; LEVINE, A. J.; ROBERTS, R. J.; SIMON, M.; SLAYMAN, C.; HUNKAPILLER, M.; BOLANOS, R.; DELCHER, A.; DEW, I.; FASULO, D.; FLANIGAN, M.; FLOREA, L.; HALPERN, A.; HANNENHALLI, S.; KRAVITZ, S.; LEVY, S.; MOBARRY, C.; REINERT, K.; REMINGTON, K.; ABU-THREIDEH, J.; BEASLEY, E.; BIDDICK, K.; BONAZZI, V.; BRANDON, R.; CARGILL, M.; CHANDRAMOULISWARAN, I.; CHARLAB, R.; CHATURVEDI, K.; DENG, Z.; Di Francesco, V.; DUNN, P.; EILBECK, K.; EVANGELISTA, C.; GABRIELIAN, A. E.; GAN, W.; GE, W.; GONG, F.; GU, Z.; GUAN, P.; HEIMAN, T. J.; HIGGINS, M. E.; JI, R. R.; KE, Z.; KETCHUM, K. A.; LAI, Z.; LEI, Y.; LI, Z.; LI, J.; LIANG, Y.; LIN, X.; LU, F.; MERKULOV, G. V.; MILSHINA, N.; MOORE, H. M.; NAIK, A. K.; NARAYAN, V. A.; NEELAM, B.; NUSSKERN, D.; RUSCH, D. B.; SALZBERG, S.; SHAO, W.; SHUE, B.; SUN, J.; WANG, Z.; WANG, A.; WANG, X.; WANG, J.; WEI, M.; WIDES, R.; XIAO, C.; YAN, C.; YAO, A.; YE, J.; ZHAN, M.; ZHANG, W.; ZHANG, H.; ZHAO, Q.; ZHENG, L.; ZHONG, F.; ZHONG, W.; ZHU, S.; ZHAO, S.; GILBERT, D.; BAUMHUETER, S.; SPIER, G.; CARTER, C.; CRAVCHIK, A.; WOODAGE, T.; ALI, F.; AN, H.; AWE, A.; BALDWIN, D.; BADEN, H.; BARNSTEAD, M.; BARROW, I.; BEESON, K.; BUSAM, D.; CARVER, A.; CENTER, A.; CHENG, M. L.; CURRY, L.; DANAHER, S.; DAVENPORT, L.; DESILETS, R.; DIETZ, S.; DODSON, K.; DOUP, L.; FERRIERA, S.; GARG, N.; GLUECKSMANN, A.; HART, B.; HAYNES, J.; HAYNES, C.; HEINER, C.; HLADUN, S.; HOSTIN, D.; HOUCK, J.; HOWLAND, T.; IBEGWAM, C.; JOHNSON, J.; KALUSH, F.; KLINE, L.; KODURU, S.; LOVE, A.; MANN, F.; MAY, D.; MCCAWLEY, S.; MCINTOSH, T.; MCMULLEN, I.; MOY, M.; MOY, L.; MURPHY, B.; NELSON, K.; PFANNKOCH, C.; PRATTS, E.; PURI, V.; QURESHI, H.; REARDON, M.; RODRIGUEZ, R.; ROGERS, Y. H.; ROMBLAD, D.; RUHFEL, B.; SCOTT, R.; SITTER, C.; SMALLWOOD, M.; STEWART, E.; STRONG, R.; SUH, E.; THOMAS, R.; TINT, N. N.; TSE, S.; VECH, C.; WANG, G.; WETTER, J.; WILLIAMS, S.; WILLIAMS, M.; WINDSOR, S.; WINN-DEEN, E.; WOLFE, K.; ZAVERI, J.; ZAVERI, K.; ABRIL, J. F.; GUIGó, R.; CAMPBELL, M. J.; SJOLANDER, K. V.; KARLAK, B.; KEJARIWAL, A.; MI, H.; LAZAREVA, B.; HATTON, T.; NARECHANIA, A.; DIEMER, K.; MURUGANUJAN, A.; GUO, N.; SATO, S.;

BAFNA, V.; ISTRAIL, S.; LIPPERT, R.; SCHWARTZ, R.; WALENZ, B.; YOOSEPH, S.; ALLEN, D.; BASU, A.; BAXENDALE, J.; BLICK, L.; CAMINHA, M.; CARNES-STINE, J.; CAULK, P.; CHIANG, Y. H.; COYNE, M.; DAHLKE, C.; MAYS, A.; DOMBROSKI, M.; DONNELLY, M.; ELY, D.; ESPARHAM, S.; FOSLER, C.; GIRE, H.; GLANOWSKI, S.; GLASSER, K.; GLODEK, A.; GOROKHOV, M.; GRAHAM, K.; GROPMAN, B.; HARRIS, M.; HEIL, J.; HENDERSON, S.; HOOVER, J.; JENNINGS, D.; JORDAN, C.; JORDAN, J.; KASHA, J.; KAGAN, L.; KRAFT, C.; LEVITSKY, A.; LEWIS, M.; LIU, X.; LOPEZ, J.; MA, D.; MAJOROS, W.; MCDANIEL, J.; MURPHY, S.; NEWMAN, M.; NGUYEN, T.; NGUYEN, N.; NODELL, M.; PAN, S.; PECK, J.; PETERSON, M.; ROWE, W.; SANDERS, R.; SCOTT, J.; SIMPSON, M.; SMITH, T.; SPRAGUE, A.; STOCKWELL, T.; TURNER, R.; VENTER, E.; WANG, M.; WEN, M.; WU, D.; WU, M.; XIA, A.; ZANDIEH, A.; ZHU, X. **The sequence of the human genome.** *Science (New York, N.Y.)*, v. 291, n. 5507, p. 1304–51, fev. 2001. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/11181995>.

WATSON, J. D.; BAKER, T. A.; BELL, S. P.; GANN, A.; LEVINE, M.; R, L. ***Biologia Molecular do Gene***. 5a ed. ed. [S.l.]: Artmed Editora SA, 2006.

WATSON, J. D.; CRICK, F. H. C. **Molecular structure of nucleic acids**. *Nature*, v. 171, p. 737–738, 1953. Disponível em: <http://www.springerlink.com/index/1L70PRG454K3U011.pdf>.

WETTERSTRAND, K. A. ***DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP)***. 2014. Disponível em: <http://www.genome.gov/sequencingcosts/>.

WETZEL, J.; KINGSFORD, C.; POP, M. **Assessing the benefits of using mate-pairs to resolve repeats in de novo short-read prokaryotic assemblies.** *BMC bioinformatics*, v. 12, n. 1, p. 95, jan. 2011. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3103447>.

YANG, X.; MEDVIN, D.; NARASIMHAN, G.; YODER-HIMES, D.; LORY, S. **CloG: A pipeline for closing gaps in a draft assembly using short reads**. *2011 IEEE 1st International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, p. 202–207, fev. 2011. Disponível em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5729881>.

APÊNDICE A
DADOS GAGE E GAGE-B

GAGE - Human chromosome 14

Reads:

	Illumina HiSeq 2000	Illumina HiSeq 2000	Illumina HiSeq 2000
Biblioteca	Paired-end (inserção 155pb)	Paired-end (inserção 2283-2803pb)	Mate-pair (inserção 35300pb)
Tamanho dos reads	101	101	76-101
Quantidade de reads	36.504.800	22.669.408	2.405.064
Cobertura esperada	42x	26x	1.3x

Códigos SRA: SRR067780, SRR067784, SRR067785, SRR067787, SRR067789, SRR067791 - SRR067793, SRR067771, SRR067773, SRR067776 - SRR067779, SRR067781, SRR067786, SRR068214, SRR068211, SRR068335

Montagens:

Assembly	abyss	abyss2	allpathslg	bambus2	cabog	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	900081	94950	**418**	1792	498	1476	30975	38477	61455
# contigs (≥ 1000 bp)	19948	9916	**137**	569	471	722	3429	3905	997
Total length (≥ 0 bp)	110452310	101532115	87688255	78613510	86486279	89678572	94748652	106080245	**143819757**
Total length (≥ 1000 bp)	59359666	91118212	87618324	78108132	86479323	89298536	84319568	98469228	**138451694**
# contigs	31582	12775	**174**	847	474	1056	9586	7264	1463
Largest contig	30053	137146	**81646936**	2671981	2260562	4208965	551622	1849511	4628722
Total length	67724594	93157107	87646728	78283056	86481568	89531681	88557645	100880746	**138771192**
Reference length	107043718	107043718	107043718	107043718	107043718	107043718	107043718	107043718	107043718
GC (%)	39.76	40.83	40.77	40.18	40.80	40.47	40.14	40.73	39.82
Reference GC (%)	40.83	40.83	40.83	40.83	40.83	40.83	40.83	40.83	40.83
N50	3355	17137	**81646936**	372757	401279	893428	82616	381286	854836
NG50	1397	14294	**81646936**	234445	296621	759545	61509	365011	1060522
N75	1591	8453	**81646936**	197251	203923	537214	36433	130244	429062
NG75	-	4779	81646936	-	71963	182823	8209	99432	741932
L50	5688	1527	**1**	54	61	25	295	77	48
LG50	15015	1970	**1**	103	91	36	425	85	31
L75	13185	3444	**1**	128	137	57	695	185	102
LG75	-	5071	1	-	258	98	1386	225	61
# misassemblies	**37**	293	461	5008	269	6923	171	8156	12832

# misassembled contigs	29	270	**18**	311	111	521	111	632	214
Misassembled contigs length	**237738**	5571464	87219645	74088134	42995746	86514853	8688013	89392506	99584186
# local misassemblies	**635**	1535	3470	11429	2083	15794	17308	10620	18390
# unaligned contigs	93 + 124 part	55 + 80 part	**0 + 3 part**	54 + 61 part	2 + 1 part	24 + 165 part	13 + 326 part	11 + 205 part	34 + 476 part
Unaligned length	276546	186345	36941	196911	**29832**	349582	1195953	1317623	23491992
Genome fraction (%)	61.766	78.049	78.701	62.737	**80.220**	75.865	70.646	77.634	68.725
Duplication ratio	1.020	1.113	1.040	1.163	**1.008**	1.100	1.155	1.201	1.567
# N's per 100 kbp	859.44	1018.34	3734.63	13247.26	**267.20**	6810.92	14499.38	10166.39	45797.46
# mismatches per 100 kbp	84.42	96.61	**66.56**	102.30	101.05	226.18	87.40	152.37	107.09
# indels per 100 kbp	**9.24**	17.45	22.71	21.80	24.55	41.58	18.22	24.41	28.34
# genes	577 + 842 part	934 + 666 part	1105 + 456 part	620 + 779 part	**1146 + 429 part**	865 + 693 part	685 + 822 part	849 + 725 part	640 + 888 part
# predicted genes (unique)	68786	84218	**661372**	66589	83109	118020	74444	180913	587273
# predicted genes ($\geq$ 0 bp)	69457	91305	**663519**	66590	83335	118666	74446	187827	589011
# predicted genes ($\geq$ 300 bp)	6117	11071	10252	8830	**12968**	11317	8116	11786	7228
# predicted genes ($\geq$ 1500 bp)	38	47	33	32	53	42	33	**61**	25
# predicted genes ($\geq$ 3000 bp)	8	9	6	6	9	8	8	**13**	5
Largest alignment	30053	115219	**2028069**	293688	1222515	349137	421478	201997	164649
NA50	3260	16483	**519085**	36820	286628	41132	55995	20001	879
NGA50	1348	13672	**403084**	13349	220007	31428	35706	17931	3367
NA75	1541	7895	**267802**	7469	136975	16606	9767	4851	-
NGA75	-	4300	87417	-	53595	1439	-	2458	-
LA50	5783	1586	**48**	542	88	577	392	1324	14478
LGA50	15403	2047	**69**	1178	129	818	599	1486	5689
LA75	13512	3605	107	1703	197	1401	1219	3700	**-**
LGA75	-	5364	194	-	370	3521	-	5038	-

GAGE - *Rhodobacter sphaeroides 2.4.1*

Reads:

	Illumina GAII	Illumina GAII
Biblioteca	Paired-end (inserção 180pb)	Mate-pair (inserção 3500pb)
Tamanho dos reads	101	101
Quantidade de reads	2.050.868	2.050.868
Cobertura esperada	45x	45x
Código SRA	SRR081522	SRR034528

Montagens:

Assembly	abyss	abyss2	allpathslg	bambus2	cabog	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	2714	480	**38**	92	130	44	2096	312	382
# contigs (≥ 1000 bp)	1030	301	32	92	130	**25**	610	56	96
Total length (≥ 0 bp)	5160167	5331930	4609785	4428612	4259679	4498559	**5614693**	4627058	4615068
Total length (≥ 1000 bp)	4729869	**5269010**	4607830	4428612	4259679	4488930	4898653	4566372	4558426
# contigs	1352	344	**33**	92	130	36	1208	76	115
Largest contig	87855	139822	**3192334**	2438508	1352519	2975504	148756	1154134	770958
Total length	4968921	5301461	4608763	4428612	4259679	4495726	**5328387**	4579801	4572546
Reference length	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977
GC (%)	68.10	68.31	68.73	68.76	69.15	68.80	68.45	68.75	68.74
Reference GC (%)	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79
N50	8036	47314	**3192334**	2438508	245073	2975504	44205	660164	353027
NG50	8859	50751	**3192334**	2438508	65690	2975504	50675	660164	353027
N75	2711	22632	**913837**	199932	22145	535984	7984	298217	139594
NG75	3531	30232	**913837**	199932	18418	535984	19992	298217	139594
L50	161	40	**1**	**1**	3	**1**	38	3	5
LG50	139	33	**1**	**1**	4	**1**	30	3	5
L75	429	80	**2**	5	31	**2**	110	5	9
LG75	339	60	**2**	5	43	**2**	66	5	9
# misassemblies	85	17	15	11	20	29	**3**	13	41
# misassembled contigs	70	12	6	4	8	14	**2**	7	8
Misassembled contigs length	1012703	574166	4447871	2971543	2427350	3765235	**31764**	1927959	1840832
# local misassemblies	77	113	74	379	**67**	79	725	217	88
# unaligned contigs	5 + 116 part	0 + 3 part	**0 + 0 part**	0 + 1 part	**0 + 0 part**	0 + 2 part	0 + 19 part	**0 + 0 part**	1 + 6 part

Unaligned length	23522	8230	**0**	4716	**0**	1377	69226	**0**	28344
Genome fraction (%)	94.218	98.577	**99.242**	94.939	91.954	96.217	90.773	98.732	97.433
Duplication ratio	1.141	1.168	1.009	1.013	**1.008**	1.015	1.259	1.009	1.013
# N's per 100 kbp	2302.47	1185.62	467.31	1288.01	505.84	725.76	21499.94	**228.42**	1898.40
# mismatches per 100 kbp	21.46	19.76	**2.91**	3.14	26.22	17.75	3.88	18.62	6.14
# indels per 100 kbp	6.63	4.92	4.12	**3.00**	5.03	8.40	4.97	6.87	7.55
# genes	3226 + 1162 part	4126 + 322 part	**4383 + 73 part**	3886 + 465 part	3998 + 148 part	4262 + 76 part	2781 + 1528 part	4227 + 219 part	4233 + 166 part
# predicted genes (unique)	5322	4684	4470	4414	4186	4361	**5423**	4532	4507
# predicted genes ($\geq$ 0 bp)	**5799**	5394	4480	4414	4186	4377	5423	4555	4517
# predicted genes ($\geq$ 300 bp)	**4606**	4571	3935	3822	3706	3848	4240	3979	3906
# predicted genes ($\geq$ 1500 bp)	439	**569**	534	503	502	523	336	523	517
# predicted genes ($\geq$ 3000 bp)	28	**52**	43	34	41	41	18	47	40
Largest alignment	87855	139631	**1443602**	1080080	676329	1208391	113353	1152195	655005
NA50	6175	44346	**928821**	343187	64491	542198	10165	539770	223489
NGA50	7136	48055	**928821**	343187	55312	535460	17695	539770	223489
NA75	2337	20625	**541057**	191703	21391	222861	717	257833	82611
NGA75	2923	28065	**541057**	157104	16011	222861	1358	223478	82611
LA50	197	41	**2**	4	8	3	81	3	6
LGA50	170	34	**2**	4	11	4	54	3	6
LA75	531	86	**4**	9	41	6	918	6	14
LGA75	427	64	**4**	10	55	6	372	7	14

GAGE - *Staphylococcus aureus* subsp. aureus

Reads:

	Illumina GAII	Illumina GAII
Biblioteca	Paired-end (inserção 180pb)	Mate-pair (inserção 3500pb)
Tamanho dos reads	101	37
Quantidade de reads	1.294.104	3.494.070
Cobertura esperada	45x	45x
Código SRA	SRR022868	SRR022865

Montagens:

Assembly	abyss	abyss2	allpathslg	bambus2	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	5012	125	19	**17**	**17**	546	175	173
# contigs (≥ 1000 bp)	198	50	11	16	**9**	132	45	19
Total length (≥ 0 bp)	**3893185**	3821622	2880676	2862930	2872905	3128388	2924135	2877995
Total length (≥ 1000 bp)	3687143	**3810221**	2879481	2862465	2869038	2927767	2888596	2856004
# contigs	206	52	**11**	16	13	299	64	26
Largest contig	130192	346557	1435559	1426293	**2411914**	286534	518710	989718
Total length	3692703	**3811756**	2879481	2862465	2871405	3051005	2902967	2860883
Reference length	2903081	2903081	2903081	2903081	2903081	2903081	2903081	2903081
GC (%)	32.60	32.60	32.65	32.55	32.70	32.44	32.57	32.57
Reference GC (%)	32.73	32.73	32.73	32.73	32.73	32.73	32.73	32.73
N50	27695	110663	1091731	1083792	**2411914**	149421	331598	762333
NG50	32467	170210	1091731	1083792	**2411914**	208206	331598	762333
N75	15495	63323	1091731	1083792	**2411914**	60721	172582	528332
NG75	22673	101536	1091731	1083792	**2411914**	67930	172582	528332
L50	40	10	2	2	**1**	7	4	2
LG50	27	6	2	2	**1**	6	4	2
L75	84	21	2	2	**1**	14	7	3
LG75	53	12	2	2	**1**	13	7	3
# misassemblies	10	16	**0**	5	49	3	32	38
# misassembled contigs	6	12	**0**	3	5	2	12	4
Misassembled contigs length	178901	938576	**0**	2682283	2806230	113622	2348756	2518079
# local misassemblies	55	**26**	40	115	42	334	52	78
# unaligned contigs	1 + 2 part	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	0 + 1 part	1 + 1 part
Unaligned length	7935	**0**	**0**	**0**	**0**	**0**	4055	1270
Genome fraction (%)	97.540	**99.219**	98.855	97.720	98.232	94.869	98.550	97.901
Duplication ratio	1.302	1.325	**1.004**	1.009	1.009	1.108	1.015	1.007
# N's per 100 kbp	1520.97	247.50	345.31	1020.27	360.56	9852.72	**167.31**	618.27
# mismatches per 100 kbp	12.22	8.26	3.97	1.48	20.90	**1.09**	24.23	14.78
# indels per 100 kbp	**1.09**	1.32	2.51	7.86	3.75	11.47	2.76	2.92
# genes	2630 + 102 part	**2765 + 23 part**	2739 + 33 part	2662 + 77 part	2733 + 33 part	2208 + 495 part	2728 + 46 part	2694 + 60 part
# predicted genes (unique)	2805	2777	2715	2759	2695	**3155**	2756	2717
# predicted genes (≥ 0 bp)	3567	**3612**	2723	2759	2712	3155	2775	2720
# predicted genes (≥ 300 bp)	3037	**3113**	2354	2377	2350	2529	2397	2351
# predicted genes (≥ 1500 bp)	375	**388**	304	288	301	213	302	301
# predicted genes (≥ 3000 bp)	36	**40**	31	29	32	22	34	30
Largest alignment	129888	345868	**1434670**	1381159	430063	257756	328221	281676
NA50	26776	101536	**1082616**	675931	220020	98338	172575	149050
NGA50	31703	137725	**1082616**	675931	220020	134849	172575	142399
NA75	14367	57665	**1082616**	393556	67740	27653	106659	70753
NGA75	21925	83719	**1082616**	393556	64114	35382	106659	70753

LA50	41	11	**2**	**2**	5	8	6	7
LGA50	28	7	**2**	**2**	5	7	6	8
LA75	88	23	**2**	3	11	22	12	15
LGA75	55	14	**2**	3	12	18	12	15

GAGE-B - *Mycobacterium abscessus ATCC 19977*

Reads:

	Illumina HiSeq 2000
Biblioteca	Paired-end (inserção 335pb)
Tamanho dos reads	100
Quantidade de reads	2.841.005
Cobertura esperada	115x
Código SRA	SRR315382

Montagens:

Assembly	abyss	cabog	msrca	sga	soapdenovo	spades	velvet
# contigs ($\geq$ 0 bp)	98	109	**59**	363	136	73	108
# contigs ($\geq$ 1000 bp)	59	109	**50**	281	61	52	53
Total length ($\geq$ 0 bp)	5136030	5116469	5146153	5140237	5151033	**5464743**	5150090
Total length ($\geq$ 1000 bp)	5126457	5116469	5140515	5107561	5133645	**5457060**	5136322
# contigs	64	109	**56**	301	66	57	59
Largest contig	**965933**	269229	375149	101254	474277	628906	622957
Total length	5130175	5116469	5144874	5120987	5137467	**5460110**	5140427
Reference length	5090491	5090491	5090491	5090491	5090491	5090491	5090491
GC (%)	64.10	64.10	64.13	64.11	64.09	64.06	64.11
Reference GC (%)	64.15	64.15	64.15	64.15	64.15	64.15	64.15
N50	147937	94359	246830	28734	150256	215724	**248309**
NG50	147937	94359	246830	28781	150256	223056	**262034**
N75	113145	50959	**122899**	17153	95162	114286	95160
NG75	113145	50959	**136061**	17362	95162	125893	95160
L50	**8**	19	9	55	10	9	**8**
LG50	8	19	9	54	10	8	**7**
L75	18	37	**17**	110	20	18	**17**
LG75	18	37	**16**	109	20	**16**	17
# misassemblies	11	10	11	**3**	10	8	9
# misassembled contigs	7	10	10	**3**	7	7	8
Misassembled contigs length	1764233	956535	1758713	**114888**	1380458	1288081	1695861
# local misassemblies	26	13	3	**1**	20	7	82
# unaligned contigs	0 + 1 part	0 + 5 part	**0 + 0 part**	0 + 3 part	1 + 1 part	0 + 1 part	**0 + 0 part**

Unaligned length	11899	65564	**0**	66157	17284	54215	**0**
Genome fraction (%)	99.134	98.892	**99.367**	98.834	99.199	99.352	99.158
Duplication ratio	1.014	**1.004**	1.017	1.005	1.014	1.069	1.018
# N's per 100 kbp	67.83	8.09	2.72	**0.00**	13.84	**0.00**	262.49
# mismatches per 100 kbp	9.77	8.79	50.41	**1.08**	1.71	4.96	3.03
# indels per 100 kbp	2.10	6.21	4.27	**0.40**	0.67	1.50	0.83
# genes	4889 + 76 part	4835 + 128 part	**4913 + 54 part**	4711 + 244 part	4888 + 74 part	4908 + 57 part	4827 + 130 part
# predicted genes (unique)	5032	5150	5058	**5179**	5021	5027	5061
# predicted genes ($\geq$ 0 bp)	5035	5150	5060	5181	5021	**5327**	5061
# predicted genes ($\geq$ 300 bp)	4593	4647	4620	4640	4578	**4861**	4604
# predicted genes ($\geq$ 1500 bp)	588	564	595	566	597	**631**	588
# predicted genes ($\geq$ 3000 bp)	70	69	71	66	73	**75**	72
Largest alignment	**663800**	238341	309535	101254	474277	628906	522498
NA50	127410	89623	187809	27712	147925	**209754**	147755
NGA50	127410	89623	187809	27760	147925	**213402**	147755
NA75	79589	42892	88132	15475	76845	**92091**	76430
NGA75	79589	42892	88132	15983	79786	**117110**	79647
LA50	**10**	21	**10**	57	12	**10**	**10**
LGA50	10	21	10	56	12	**9**	10
LA75	22	40	**19**	116	24	20	22
LGA75	22	40	19	114	23	**17**	21

GAGE-B - *Rhodobacter sphaeroides 2.4.1*

Reads:

	Illumina HiSeq 2000
Biblioteca	Paired-end (inserção 220pb)
Tamanho dos reads	101
Quantidade de reads	4.802.518
Cobertura esperada	210x
Código SRA	SRR522244

Montagens:

Assembly	abyss	cabog	msrca	sga	soapdenovo	spades	velvet
# contigs ($\geq$ 0 bp)	1513	320	**125**	662	7229	299	435
# contigs ($\geq$ 1000 bp)	459	320	**84**	452	472	104	251
Total length ($\geq$ 0 bp)	4635249	4220437	4489304	4023509	**5222286**	4668602	4566780
Total length ($\geq$ 1000 bp)	4509636	4220437	4468482	3953373	4393539	**4611740**	4511166
# contigs	518	320	**101**	481	555	129	283

Largest contig	66408	132060	358962	75323	57198	**511523**	119965
Total length	4552480	4220437	4479773	3974213	4455778	**4630435**	4535262
Reference length	4602977	4602977	4602977	4602977	4602977	4602977	4602977
GC (%)	68.73	68.95	68.87	68.93	68.88	68.83	68.78
Reference GC (%)	68.79	68.79	68.79	68.79	68.79	68.79	68.79
N50	13523	23585	**196511**	13674	15950	127911	36086
NG50	13460	21196	**196511**	11833	15520	127911	34182
N75	8674	12169	**102004**	7216	8188	73118	19195
NG75	8502	9843	**76502**	4750	7371	74016	18000
L50	97	57	**9**	89	90	10	39
LG50	99	65	**9**	114	94	10	40
L75	204	117	**17**	185	187	22	83
LG75	208	143	**18**	264	202	21	86
# misassemblies	27	6	5	**1**	2	6	20
# misassembled contigs	22	5	4	**1**	2	6	11
Misassembled contigs length	136091	110666	36508	**8334**	13073	188233	59273
# local misassemblies	11	33	6	**1**	125	9	333
# unaligned contigs	**0 + 1 part**	0 + 4 part	**0 + 1 part**	**0 + 1 part**	7 + 2 part	0 + 2 part	**0 + 1 part**
Unaligned length	**27**	146	29	54	6500	103	43
Genome fraction (%)	97.987	90.894	97.028	86.226	96.402	**99.313**	97.594
Duplication ratio	1.009	1.009	1.003	**1.001**	1.003	1.013	1.010
# N's per 100 kbp	37.12	103.78	2.23	**0.00**	48.12	14.99	916.02
# mismatches per 100 kbp	8.98	21.65	72.05	**1.11**	24.47	6.13	10.11
# indels per 100 kbp	0.73	6.74	1.05	**0.23**	4.28	1.33	0.69
# genes	4032 + 370 part	3793 + 341 part	4298 + 58 part	3511 + 513 part	3812 + 593 part	**4374 + 79 part**	3913 + 518 part
# predicted genes (unique)	4725	4354	4379	4163	**4811**	4508	4802
# predicted genes ($\geq$ 0 bp)	4747	4379	4382	4165	**4811**	4573	4803
# predicted genes ($\geq$ 300 bp)	4053	3747	3855	3562	4034	4022	**4065**
# predicted genes ($\geq$ 1500 bp)	504	480	541	444	474	**550**	476
# predicted genes ($\geq$ 3000 bp)	34	33	**45**	25	34	43	33
Largest alignment	66408	132060	358962	75323	57186	**511523**	119041
NA50	13475	23406	**196511**	13674	15950	127911	33823
NGA50	13460	20580	**196511**	11793	15517	127911	33086
NA75	8674	12157	**102004**	7216	7924	68531	17995
NGA75	8441	9626	**76502**	4750	7370	73118	17575
LA50	98	58	**9**	89	90	10	40
LGA50	99	67	**9**	114	94	10	41
LA75	204	119	**17**	185	188	22	85
LGA75	209	146	**18**	264	202	21	87

GAGE-B - *Vibrio cholerae CO1032(5)*

Reads:

	Illumina HiSeq 2000
Biblioteca	Paired-end (inserção 335pb)
Tamanho dos reads	101
Quantidade de reads	1.960.045
Cobertura esperada	110x
Código SRA	SRR227312

Montagens:

Assembly	abyss	cabog	msrca	sga	soapdenovo	spades	velvet
# contigs ($\geq$ 0 bp)	249	109	**102**	430	317	122	182
# contigs ($\geq$ 1000 bp)	87	108	77	274	**58**	95	65
Total length ($\geq$ 0 bp)	**4126440**	3856143	3994677	3892845	3972061	4017092	3961318
Total length ($\geq$ 1000 bp)	**4099616**	3855980	3980219	3822893	3923264	4004247	3927326
# contigs	102	108	88	331	**75**	106	85
Largest contig	523887	256726	555664	108154	**574497**	246251	524347
Total length	**4109111**	3855980	3989222	3861930	3935433	4012066	3940257
Reference length	4033464	4033464	4033464	4033464	4033464	4033464	4033464
GC (%)	47.57	47.53	47.50	47.55	47.52	47.42	47.50
Reference GC (%)	47.49	47.49	47.49	47.49	47.49	47.49	47.49
N50	217596	67078	**246505**	24152	200529	98274	172545
NG50	217596	67009	**246505**	23429	181109	98274	172545
N75	80568	34758	85155	12815	**102918**	52916	92099
NG75	82716	28288	79323	10978	**102918**	52916	82167
L50	**6**	16	**6**	48	**6**	13	7
LG50	**6**	17	**6**	52	7	13	7
L75	15	36	**13**	102	**13**	26	14
LG75	14	40	14	113	**13**	26	15
# misassemblies	15	21	9	**3**	21	20	10
# misassembled contigs	14	15	6	**3**	12	16	8
Misassembled contigs length	924513	531956	716209	**17831**	1595689	384448	395034
# local misassemblies	68	23	6	**0**	64	16	129
# unaligned contigs	1 + 0 part	**0 + 0 part**	0 + 2 part	1 + 0 part	1 + 1 part	1 + 1 part	1 + 0 part
Unaligned length	602	**0**	399	551	603	1261	549
Genome fraction (%)	98.010	95.629	**98.147**	95.589	97.289	98.053	97.140
Duplication ratio	1.040	**1.001**	1.009	1.002	1.003	1.016	1.006
# N's per 100 kbp	424.33	9.85	1.50	**0.00**	56.11	**0.00**	547.10
# mismatches per 100 kbp	7.34	17.32	68.63	**2.98**	12.56	8.95	5.18
# indels per 100 kbp	5.11	7.83	5.68	**2.62**	3.67	5.21	4.39
# genes	3515 + 54 part	3380 + 117 part	**3573 + 46 part**	3272 + 159 part	3480 + 50 part	3536 + 47 part	3400 + 128 part
# predicted genes (unique)	3556	3540	**3594**	3571	3510	3555	3555
# predicted genes ($\geq$ 0 bp)	**3672**	3542	3624	3572	3512	3637	3555

# predicted genes ($\geq$ 300 bp)	**3290**	3172	3227	3177	3174	3238	3203
# predicted genes ($\geq$ 1500 bp)	**590**	535	565	544	574	570	560
# predicted genes ($\geq$ 3000 bp)	**55**	54	54	49	54	**55**	52
Largest alignment	522201	256726	555664	108154	**557947**	246251	521619
NA50	157127	63201	**236373**	24152	181109	95858	171505
NGA50	157127	62912	**236373**	23429	181109	94762	171505
NA75	67355	33039	79323	12813	91819	52911	**91938**
NGA75	71310	28288	76901	10978	**80998**	52911	80883
LA50	7	18	**6**	48	7	13	7
LGA50	7	19	**6**	52	7	14	7
LA75	18	39	**14**	102	**14**	27	**14**
LGA75	17	43	**15**	113	**15**	27	**15**

APÊNDICE B
PARÂMETROS

FGAP (v1.7) - Padrão:
minScore: 25; maxEValue: 1e-07; minIdentity: 70; contigEndLength: 300; edgeTrimLength: 0; maxRemoveLength: 500; maxInsertLength: 500; positiveGap: 1; zeroGap: 0; negativeGap: 0; gapChar: N; blastAlignParam: 1,1,1,-3,15; blastMaxResults: 200;

FGAP (v1.7) - Teste Controlado:
minScore: 25; maxEValue: 1e-07; minIdentity: 70; contigEndLength: 300; edgeTrimLength: 0; maxRemoveLength: 500; maxInsertLength: 500; positiveGap: 1; zeroGap: 1; negativeGap: 1; gapChar: N; blastAlignParam: 1,1,1,-3,15; blastMaxResults: 200;

FGAP (v1.7) - Cromossomo humano 14:
minScore: 250; maxEValue: 1e-07; minIdentity: 70; contigEndLength: 3000; edgeTrimLength: 0; maxRemoveLength: 500; maxInsertLength: 500; positiveGap: 1; zeroGap: 0; negativeGap: 0; gapChar: N; blastAlignParam: 1,1,1,-3,15; blastMaxResults: 20;

GAPCLOSER (v1.12):
avg. insert size: 200; max. read length: 101;

GAPFILLER (v1.11):
insert size: 200; min. error: 0.25; orientation: FR;

IMAGE (v2.33):
iterations: 10; k-mer: 81; velvet insert length: 200;

QUAST (v2.3):
-R: reference file; -G: genes file; --gene-finding; --scaffolds (Somente para comparação de programas);

Para todos os outros parâmetros não citados acima foram utilizados os valores padrão de cada programa

APÊNDICE C
RESULTADOS COMPLEMENTARES

Comparação com outros programas - *E. coli*

Assembly	FGAP	FGAP+P	GAPCLOSER	GAPFILLER	IMAGE
# contigs (≥ 0 bp)	99	**75**	95	98	95
# contigs (≥ 1000 bp)	80	**73**	82	85	91
Total length (≥ 0 bp)	4554904	4558702	4557911	4554869	**4592469**
Total length (≥ 1000 bp)	4547032	4557786	4554321	4549678	**4591033**
# contigs	87	**73**	83	87	92
Largest contig	414012	**414013**	414012	414012	347930
Total length	4552598	4557786	4555296	4551260	**4591978**
Reference length	4641652	4641652	4641652	4641652	4641652
GC (%)	50.73	50.74	50.74	50.73	50.75
Reference GC (%)	50.79	50.79	50.79	50.79	50.79
N50	132608	**172148**	112396	132608	110882
NG50	132608	**148525**	112396	132608	110882
N75	57888	**61315**	57888	57972	55559
NG75	56977	**59718**	57222	57888	54764
L50	12	**10**	12	12	13
LG50	12	**11**	12	12	13
L75	25	**22**	26	25	28
LG75	26	**23**	27	26	29
# misassemblies	1	1	1	1	1
# misassembled contigs	1	1	1	1	1
Misassembled contigs length	185322	185318	185319	**124502**	124806
# local misassemblies	9	**2**	12	12	22
# unaligned contigs	0 + 0 part	0 + 0 part	0 + 0 part	0 + 0 part	0 + 0 part
Unaligned length	0	0	0	0	0
Genome fraction (%)	98.084	98.177	98.108	98.033	**98.672**
Duplication ratio	**1.000**	**1.000**	**1.000**	**1.000**	1.003
# N's per 100 kbp	0.00	0.00	0.00	0.00	0.00
# mismatches per 100 kbp	4.11	3.49	1.19	**1.16**	6.55
# indels per 100 kbp	0.90	13.14	**0.15**	0.53	1.03
# genes	4377 + 34 part	**4388 + 27 part**	4375 + 35 part	4367 + 35 part	4386 + 69 part
# predicted genes (unique)	4291	4306	4303	4294	**4355**
# predicted genes (≥ 0 bp)	4294	4308	4306	4296	**4365**
# predicted genes (≥ 300 bp)	3789	3792	3793	3785	**3819**
# predicted genes (≥ 1500 bp)	574	573	**576**	574	575
# predicted genes (≥ 3000 bp)	51	51	51	51	51
Largest alignment	414012	**414013**	414012	414012	347930
NA50	115083	**133062**	110227	132608	107145
NGA50	114132	**133062**	106887	132608	107145
NA75	57888	**61315**	57888	57972	54764
NGA75	56977	**59718**	57222	57888	54153
LA50	12	**11**	12	12	13
LGA50	13	**11**	13	12	13
LA75	26	**23**	27	25	29
LGA75	27	**24**	28	26	30

GAGE - Cromossomo Humano 14

Assembly	abyss2	allpathslg	bambus2	cabog	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	94950	**418**	1792	498	1476	30975	38477	61455
# contigs (≥ 1000 bp)	9916	**137**	569	471	722	3429	3905	998
Total length (≥ 0 bp)	101522258	87693509	78611652	86509787	90024415	94916662	105622875	**143608383**
Total length (≥ 1000 bp)	91108355	87623578	78106274	86502831	89644379	84487578	98011858	**138241267**
# contigs	12775	**174**	847	474	1056	9586	7264	1463
Largest contig	137146	**81656333**	2672038	2261791	4222531	552946	1841356	4621881
Total length	93147250	87651982	78281198	86505076	89877524	88725655	100423376	**138559818**
Reference length	107043718	107043718	107043718	107043718	107043718	107043718	107043718	107043718
GC (%)	40.83	40.79	40.19	40.81	40.70	40.33	40.75	39.95
Reference GC (%)	40.83	40.83	40.83	40.83	40.83	40.83	40.83	40.83
N50	17115	**81656333**	373044	401311	897636	82898	379059	856047
NG50	14284	**81656333**	234327	296749	762409	61723	358640	1062971
N75	8453	**81656333**	197251	203923	541078	36587	130244	428407
NG75	4771	**81656333**	-	71957	190828	8706	95415	735757
L50	1527	**1**	54	61	25	295	77	48
LG50	1970	**1**	103	91	36	423	86	31
L75	3444	**1**	128	137	57	694	185	102
LG75	5073	1	**-**	258	96	1366	229	62
# misassemblies	366	508	5009	395	5753	**339**	8000	13679
# misassembled contigs	301	**17**	311	144	505	174	632	232
Misassembled contigs length	**6470631**	87210821	74086699	53117470	86362452	17042845	89011460	106208910
# local misassemblies	**1088**	2447	11137	1906	6033	9363	8753	14800
# unaligned contigs	55 + 80 part	**0 + 4 part**	54 + 60 part	2 + 1 part	25 + 165 part	13 + 323 part	11 + 203 part	28 + 453 part
Unaligned length	183126	47499	193013	**29832**	340799	1115002	1271325	19619515
Genome fraction (%)	78.100	78.926	62.805	**80.242**	77.950	72.186	77.800	70.203
Duplication ratio	1.113	1.037	1.162	**1.008**	1.075	1.134	1.193	1.583
# N's per 100 kbp	870.87	3429.39	13159.89	**219.58**	4840.74	12594.50	9605.19	44975.49
# mismatches per 100 kbp	98.34	**67.86**	102.67	102.20	227.26	90.97	151.89	108.65
# indels per 100 kbp	**17.64**	22.59	21.95	24.49	36.59	18.29	25.30	28.14
# genes	949 + 651 part	**1167 + 397 part**	623 + 776 part	1157 + 417 part	1010 + 559 part	802 + 717 part	881 + 694 part	674 + 858 part

# predicted genes (unique)	84442	**662123**	66644	82943	120379	76019	178480	597638
# predicted genes (≥ 0 bp)	91579	**664391**	66648	83179	121115	76034	185304	599157
# predicted genes (≥ 300 bp)	11114	10378	8863	**12978**	11990	8604	11861	7457
# predicted genes (≥ 1500 bp)	48	34	32	55	44	37	**61**	25
# predicted genes (≥ 3000 bp)	9	6	6	9	8	9	**13**	5
Largest alignment	115219	**1989599**	293688	1100906	490442	389756	202069	166158
NA50	16402	**501153**	36730	249288	52034	55988	20580	1396
NGA50	13646	**373982**	13409	178250	40995	37874	18335	4057
NA75	7894	**242845**	7574	123978	23738	13184	5286	-
NGA75	4326	**82051**	-	48410	5508	-	2574	-
LA50	1596	**53**	542	100	474	407	1290	11620
LGA50	2060	**75**	1179	148	660	605	1460	5052
LA75	3620	116	1700	225	1104	1148	3544	**-**
LGA75	5376	211	**-**	415	2109	**-**	4888	**-**

GAGE - *R. sphaeroides*

Assembly	abyss	abyss2	allpathslg	bambus2	cabog	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	2714	480	**38**	92	130	44	2096	312	382
# contigs (≥ 1000 bp)	1031	301	32	92	130	**25**	610	56	96
Total length (≥ 0 bp)	5159653	5324808	4609124	4430900	4259745	4495573	**5616620**	4626068	4612438
Total length (≥ 1000 bp)	4730268	**5261891**	4607169	4430900	4259745	4485944	4900580	4565382	4555796
# contigs	1352	344	**33**	92	130	36	1208	76	115
Largest contig	87865	139778	**3191959**	2439636	1352576	2973681	148655	1153848	771245
Total length	4968360	5294342	4608102	4430900	4259745	4492740	**5330314**	4578811	4569916
Reference length	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977
GC (%)	68.11	68.34	68.74	68.77	69.16	68.81	68.48	68.75	68.76
Reference GC (%)	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79
N50	8036	47317	**3191959**	2439636	245038	2973681	44262	660151	352465
NG50	8851	50751	**3191959**	2439636	65690	2973681	50779	660151	352465
N75	2710	22517	**913587**	200451	22145	535722	7952	298132	139520
NG75	3531	30232	**913587**	200451	18408	535722	19992	298132	139520
L50	161	40	**1**	**1**	3	**1**	38	3	5
LG50	139	33	**1**	**1**	4	**1**	30	3	5
L75	429	80	**2**	5	31	**2**	110	5	9

LG75	339	60	**2**	5	43	**2**	66	5	9
# misassemblies	85	15	15	11	18	29	**5**	13	34
# misassembled contigs	71	10	6	4	8	14	**3**	7	8
Misassembled contigs length	1016978	548826	4447210	2972895	2427414	3762528	**180578**	1927786	1841081
# local misassemblies	64	96	64	363	**62**	82	637	211	124
# unaligned contigs	2 + 115 part	0 + 3 part	**0 + 0 part**	0 + 1 part	**0 + 0 part**	0 + 2 part	0 + 19 part	**0 + 0 part**	1 + 6 part
Unaligned length	20607	8230	**0**	4716	**0**	1377	69226	**0**	27902
Genome fraction (%)	94.325	98.674	**99.326**	95.077	92.028	96.235	91.244	98.751	97.378
Duplication ratio	1.140	1.165	**1.008**	1.012	**1.008**	1.014	1.253	**1.008**	1.014
# N's per 100 kbp	2162.14	856.99	370.43	1199.89	441.29	621.78	20987.71	**183.98**	1727.97
# mismatches per 100 kbp	21.32	21.55	**3.63**	5.62	26.94	18.06	3.73	18.92	14.35
# indels per 100 kbp	3.09	3.30	3.81	3.41	4.70	7.31	**0.99**	6.75	3.60
# genes	3239 + 1147 part	4143 + 307 part	**4395 + 61 part**	3902 + 451 part	4006 + 141 part	4267 + 71 part	2873 + 1444 part	4229 + 217 part	4207 + 191 part
# predicted genes (unique)	5267	4639	4466	4408	4177	4348	**5364**	4528	4458
# predicted genes (≥ 0 bp)	**5743**	5350	4476	4408	4178	4364	5364	4551	4468
# predicted genes (≥ 300 bp)	**4589**	4552	3933	3826	3702	3841	4225	3975	3900
# predicted genes (≥ 1500 bp)	448	**583**	536	504	506	525	351	525	528
# predicted genes (≥ 3000 bp)	29	**56**	43	34	40	41	22	47	39
Largest alignment	87865	139778	**1444269**	1080678	676698	1208563	100859	1152145	652823
NA50	6219	44495	**930452**	343187	112370	541721	10383	539748	223836
NGA50	7152	48077	**930452**	343187	56486	535157	18123	539748	223836
NA75	2340	20650	**540950**	194209	21391	222521	741	257748	82687
NGA75	2923	28065	**540950**	157104	16359	222521	1401	223653	82687
LA50	196	41	**2**	4	7	3	80	3	6
LGA50	169	34	**2**	4	10	4	54	3	6
LA75	528	86	**4**	9	40	6	888	6	14
LGA75	425	64	**4**	10	54	6	356	7	14

GAGE - *S. aureus*

Assembly	abyss	abyss2	allpathslg	bambus2	msrca	sga	soapdenovo	velvet
# contigs (≥ 0 bp)	5012	125	19	**17**	**17**	546	175	173
# contigs (≥ 1000 bp)	198	50	11	16	**9**	132	45	19
Total length (≥ 0 bp)	**3893116**	3820555	2881908	2865039	2872548	3128004	2924291	2882462
Total length (≥ 1000 bp)	3687074	**3809154**	2880713	2864574	2868681	2927383	2888752	2860471
# contigs	206	52	**11**	16	13	299	64	26
Largest contig	130192	346556	1436580	1427663	**2412085**	286577	518710	992474
Total length	3692634	**3810689**	2880713	2864574	2871048	3050621	2903123	2865350
Reference length	2903081	2903081	2903081	2903081	2903081	2903081	2903081	2903081
GC (%)	32.60	32.60	32.66	32.57	32.70	32.51	32.57	32.58
Reference GC (%)	32.73	32.73	32.73	32.73	32.73	32.73	32.73	32.73
N50	27695	110519	1091761	1084573	**2412085**	149534	331598	763340
NG50	32467	170130	1091761	1084573	**2412085**	208306	331598	763340
N75	15495	63116	1091761	1084573	**2412085**	60400	172582	528635
NG75	22673	101547	1091761	1084573	**2412085**	67942	172582	528635
L50	40	10	2	2	**1**	7	4	2
LG50	27	6	2	2	**1**	6	4	2
L75	84	21	2	2	**1**	14	7	3
LG75	53	12	2	2	**1**	13	7	3
# misassemblies	8	15	**4**	9	31	5	32	32
# misassembled contigs	5	9	**2**	**2**	5	3	12	4
Misassembled contigs length	**159081**	857506	1616322	2512236	2805873	321870	2348912	2522482
# local misassemblies	32	**9**	30	81	36	239	51	45
# unaligned contigs	1 + 2 part	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	**0 + 0 part**	0 + 1 part	1 + 1 part
Unaligned length	7935	**0**	**0**	**0**	**0**	**0**	4055	1270
Genome fraction (%)	97.604	**99.329**	98.968	97.994	98.369	95.800	98.554	98.256
Duplication ratio	1.301	1.323	**1.003**	1.007	1.007	1.097	1.015	1.005
# N's per 100 kbp	1341.21	**50.86**	264.48	775.05	219.01	8672.56	161.58	343.41
# mismatches per 100 kbp	12.11	11.10	5.99	**4.43**	21.43	4.82	24.16	20.72
# indels per 100 kbp	**1.02**	1.70	1.04	7.21	3.82	1.62	2.90	2.77
# genes	2641 + 91 part	**2772 + 17 part**	2746 + 26 part	2694 + 59 part	2749 + 26 part	2302 + 410 part	2729 + 45 part	2727 + 41 part
# predicted genes (unique)	2809	2776	2712	2758	2696	**3012**	2758	2719
# predicted genes (≥ 0 bp)	3564	**3611**	2720	2759	2716	3013	2777	2722
# predicted genes (≥ 300 bp)	3050	**3119**	2357	2381	2346	2455	2397	2357
# predicted genes (≥ 1500 bp)	376	**389**	304	289	301	250	302	300
# predicted genes (≥ 3000 bp)	36	**40**	34	30	33	26	34	30
Largest alignment	130192	346556	**1084202**	859904	605521	261073	328221	436131
NA50	26776	101547	**897266**	394857	430211	99004	172575	158012
NGA50	31695	137733	**897266**	394857	430211	136317	172575	158012
NA75	14539	58150	**538658**	178486	110296	27799	106659	72786
NGA75	22214	83719	**538658**	178486	106410	35342	106659	72786
LA50	41	11	**2**	3	3	8	6	6
LGA50	28	7	**2**	3	3	7	6	6

LA75	88	23	**3**	5	7	22	12	12
LGA75	55	14	**3**	5	8	18	12	12

GAGE-B - *M. abscessus*

Assembly	abyss	cabog	msrca	soapdenovo	velvet
# contigs (≥ 0 bp)	98	109	**59**	136	108
# contigs (≥ 1000 bp)	59	109	**50**	61	53
Total length (≥ 0 bp)	5135230	5116571	5146138	**5150758**	5140553
Total length (≥ 1000 bp)	5125657	5116571	**5140500**	5133370	5126785
# contigs	64	109	**56**	66	59
Largest contig	**965989**	269214	375149	474277	621695
Total length	5129375	5116571	**5144859**	5137192	5130890
Reference length	5090491	5090491	5090491	5090491	5090491
GC (%)	64.10	64.10	64.13	64.09	64.11
Reference GC (%)	64.15	64.15	64.15	64.15	64.15
N50	147856	94359	246830	150256	**247959**
NG50	147856	94359	246830	150256	**247959**
N75	113132	50959	**122899**	95162	95160
NG75	113132	50959	**136061**	95162	95160
L50	**8**	19	9	10	**8**
LG50	**8**	19	9	10	**8**
L75	18	37	**17**	20	**17**
LG75	18	37	**16**	20	17
# misassemblies	11	10	11	**8**	9
# misassembled contigs	7	10	10	**6**	8
Misassembled contigs length	1763974	**956472**	1758713	1159760	1692662
# local misassemblies	22	11	**3**	19	23
# unaligned contigs	0 + 1 part	0 + 5 part	**0 + 0 part**	1 + 1 part	**0 + 0 part**
Unaligned length	11899	65564	**0**	17284	**0**
Genome fraction (%)	99.142	98.897	**99.367**	99.201	99.214
Duplication ratio	1.014	**1.004**	1.017	1.014	1.016
# N's per 100 kbp	36.83	4.22	**2.33**	9.25	37.38
# mismatches per 100 kbp	10.42	8.47	50.41	**1.73**	3.60
# indels per 100 kbp	1.33	5.89	4.25	**0.73**	0.81
# genes	4891 + 74 part	4837 + 126 part	**4913 + 54 part**	4888 + 74 part	4886 + 72 part
# predicted genes (unique)	5024	**5146**	5058	5022	5013
# predicted genes (≥ 0 bp)	5027	**5146**	5060	5022	5013
# predicted genes (≥ 300 bp)	4589	**4645**	4620	4579	4579
# predicted genes (≥ 1500 bp)	594	565	595	**597**	594
# predicted genes (≥ 3000 bp)	72	69	71	**73**	**73**
Largest alignment	**663825**	238341	309535	474277	522731
NA50	127410	89623	**187809**	147925	148854
NGA50	127410	89623	**187809**	147925	148854
NA75	79618	42892	**88132**	76845	76432
NGA75	79618	42892	**88132**	79786	79678

LA50	**10**	21	**10**	12	**10**
LGA50	**10**	21	**10**	12	**10**
LA75	22	40	**19**	24	22
LGA75	22	40	**19**	23	21

GAGE-B - *R. sphaeroides*

Assembly	abyss	cabog	soapdenovo	spades	velvet
# contigs (≥ 0 bp)	1513	320	7229	**299**	435
# contigs (≥ 1000 bp)	459	320	474	**104**	251
Total length (≥ 0 bp)	4635967	4219538	**5223055**	4668594	4540601
Total length (≥ 1000 bp)	4510446	4219538	4395981	**4611734**	4485161
# contigs	517	320	555	**129**	283
Largest contig	66408	132066	57198	**511523**	119364
Total length	4552916	4219538	4456547	**4630429**	4509083
Reference length	4602977	4602977	4602977	4602977	4602977
GC (%)	68.73	68.95	68.88	68.83	68.79
Reference GC (%)	68.79	68.79	68.79	68.79	68.79
N50	13523	23582	15950	**127911**	35976
NG50	13460	21177	15520	**127911**	33380
N75	8674	12157	8188	**73118**	19195
NG75	8502	9830	7371	**74016**	17601
L50	97	57	90	**10**	39
LG50	99	65	94	**10**	41
L75	204	117	187	**22**	83
LG75	208	143	202	**21**	87
# misassemblies	25	6	**2**	6	20
# misassembled contigs	21	5	**2**	6	11
Misassembled contigs length	135786	110664	**13073**	188233	59132
# local misassemblies	**9**	35	119	**9**	35
# unaligned contigs	**0 + 1 part**	0 + 4 part	1 + 2 part	0 + 2 part	**0 + 1 part**
Unaligned length	**27**	146	2085	103	43
Genome fraction (%)	97.992	90.894	96.517	**99.313**	97.857
Duplication ratio	1.009	1.009	1.003	1.013	**1.001**
# N's per 100 kbp	26.38	62.09	34.71	**7.58**	77.69
# mismatches per 100 kbp	10.40	21.15	24.40	**8.86**	9.90
# indels per 100 kbp	0.73	4.92	3.98	1.31	**0.53**
# genes	4033 + 367 part	3793 + 341 part	3818 + 591 part	**4374 + 79 part**	4155 + 276 part
# predicted genes (unique)	4717	4343	**4799**	4510	4586
# predicted genes (≥ 0 bp)	4739	4368	**4799**	4573	4587
# predicted genes (≥ 300 bp)	**4050**	3738	4027	4022	3957
# predicted genes (≥ 1500 bp)	505	483	475	**550**	513
# predicted genes (≥ 3000 bp)	34	33	35	**43**	38
Largest alignment	66408	132066	57186	**511523**	119327
NA50	13475	23406	15950	**127911**	35827
NGA50	13460	20570	15517	**127911**	33380

NA75	8674	12145	7924	**68531**	18462
NGA75	8441	9626	7370	**73118**	17601
LA50	98	58	90	**10**	39
LGA50	99	67	94	**10**	41
LA75	204	119	188	**22**	83
LGA75	209	146	202	**21**	87

GAGE-B - *V. cholerae*

Assembly	abyss	cabog	msrca	soapdenovo	velvet
# contigs ($\geq$ 0 bp)	249	109	**102**	317	182
# contigs ($\geq$ 1000 bp)	87	108	77	**58**	65
Total length ($\geq$ 0 bp)	**4123899**	3856176	3994767	3971680	3945252
Total length ($\geq$ 1000 bp)	**4097075**	3856013	3980309	3922985	3911260
# contigs	102	108	88	**75**	85
Largest contig	523142	256726	555664	**574497**	522790
Total length	**4106570**	3856013	3989312	3935154	3924191
Reference length	4033464	4033464	4033464	4033464	4033464
GC (%)	47.56	47.53	47.50	47.52	47.51
Reference GC (%)	47.49	47.49	47.49	47.49	47.49
N50	217824	67011	**246505**	200529	171642
NG50	217824	67009	**246505**	181109	171642
N75	80569	34758	85155	**102918**	91940
NG75	82896	28288	79323	**102918**	82013
L50	**6**	16	**6**	**6**	7
LG50	**6**	17	**6**	7	7
L75	15	36	**13**	**13**	14
LG75	14	40	14	**13**	15
# misassemblies	42	21	9	23	**8**
# misassembled contigs	22	15	**6**	12	**6**
Misassembled contigs length	2088175	531956	716299	1595594	**172334**
# local misassemblies	35	22	**6**	62	13
# unaligned contigs	1 + 0 part	**0 + 0 part**	0 + 2 part	1 + 1 part	1 + 0 part
Unaligned length	602	**0**	399	603	549
Genome fraction (%)	98.011	95.632	**98.150**	97.281	97.207
Duplication ratio	1.040	**1.001**	1.009	1.003	**1.001**
# N's per 100 kbp	124.00	7.26	**1.00**	43.63	25.05
# mismatches per 100 kbp	15.33	16.64	68.83	12.69	**4.23**
# indels per 100 kbp	4.05	7.03	5.71	**3.70**	4.00
# genes	3529 + 44 part	3381 + 116 part	**3573 + 46 part**	3480 + 49 part	3503 + 36 part
# predicted genes (unique)	3530	3536	**3594**	3510	3496
# predicted genes ($\geq$ 0 bp)	**3654**	3538	3624	3512	3496
# predicted genes ($\geq$ 300 bp)	**3276**	3169	3227	3174	3163
# predicted genes ($\geq$ 1500 bp)	**596**	537	565	574	577
# predicted genes ($\geq$ 3000 bp)	**56**	54	54	54	54
Largest alignment	297750	256726	555664	**557947**	522466

NA50	157151	63201	**236373**	181109	171642
NGA50	157151	62912	**236373**	181109	171642
NA75	66251	33039	79323	91819	**91940**
NGA75	66264	28288	76901	**80998**	80895
LA50	9	18	**6**	7	7
LGA50	9	19	**6**	7	7
LA75	21	39	**14**	**14**	**14**
LGA75	20	43	**15**	**15**	**15**

GAGE e GAGE-B - *R. sphaeroides*

Assembly	abyss	abyss2	allpathslg	bambus2	cabog	msrca	sga	soapdenovo	velvet
# contigs ($\geq$ 0 bp)	2714	480	**38**	92	130	44	2096	312	382
# contigs ($\geq$ 1000 bp)	1031	301	32	92	130	**25**	610	56	96
Total length ($\geq$ 0 bp)	5160938	5335586	4611507	4431990	4260256	4498120	**5620012**	4627022	4626498
Total length ($\geq$ 1000 bp)	4731556	**5272654**	4609552	4431990	4260256	4488491	4903972	4566336	4569856
# contigs	1352	344	**33**	92	130	36	1208	76	115
Largest contig	87865	140095	**3193788**	2440428	1352974	2975379	149166	1153879	772916
Total length	4969645	5305105	4610485	4431990	4260256	4495287	**5333706**	4579765	4583976
Reference length	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977	4602977
GC (%)	68.11	68.35	68.75	68.77	69.17	68.82	68.48	68.76	68.78
Reference GC (%)	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79	68.79
N50	8036	47402	**3193788**	2440428	245161	2975379	44262	660139	353939
NG50	8851	50751	**3193788**	2440428	65690	2975379	50779	660139	353939
N75	2711	22624	**914141**	200712	22517	535857	7991	298086	140416
NG75	3531	30232	**914141**	200712	18408	535857	19992	298086	140416
L50	161	40	**1**	**1**	3	**1**	38	3	5
LG50	139	33	**1**	**1**	4	**1**	30	3	5
L75	429	80	**2**	5	30	**2**	109	5	9
LG75	339	60	**2**	5	43	**2**	66	5	9
# misassemblies	87	14	13	9	18	29	**3**	13	34
# misassembled contigs	73	9	6	4	8	14	**2**	7	8
Misassembled contigs length	1028339	440248	4449593	2973687	2428053	3764940	**31923**	1928432	1846519
# local misassemblies	58	**40**	45	353	46	67	622	202	50
# unaligned contigs	2 + 113 part	0 + 3 part	**0 + 0 part**	0 + 1 part	**0 + 0 part**	0 + 2 part	0 + 19 part	**0 + 0 part**	1 + 6 part
Unaligned length	19184	8230	**0**	4716	**0**	1377	69226	**0**	27911
Genome fraction (%)	94.395	98.831	**99.416**	95.111	92.056	96.283	91.286	98.795	97.537

Duplication ratio	1.140	1.166	1.008	1.011	**1.007**	1.014	1.253	1.008	1.015
# N's per 100 kbp	2159.55	866.77	297.78	1183.78	367.37	577.14	20993.13	**156.45**	1721.50
# mismatches per 100 kbp	21.72	21.12	**2.49**	4.91	26.03	16.79	3.81	18.63	8.39
# indels per 100 kbp	2.93	2.13	2.19	3.20	3.71	5.96	**0.75**	6.62	1.72
# genes	3246 + 1144 part	4193 + 257 part	**4416 + 40 part**	3908 + 445 part	4016 + 131 part	4280 + 59 part	2887 + 1430 part	4236 + 210 part	4274 + 125 part
# predicted genes (unique)	5267	4628	4449	4405	4169	4337	**5366**	4523	4453
# predicted genes (≥ 0 bp)	**5742**	5338	4459	4405	4170	4353	5366	4546	4463
# predicted genes (≥ 300 bp)	**4589**	4551	3930	3824	3702	3838	4231	3973	3905
# predicted genes (≥ 1500 bp)	449	**592**	541	504	510	531	351	527	533
# predicted genes (≥ 3000 bp)	29	**55**	43	34	40	41	22	47	40
Largest alignment	87865	140095	**1446303**	1113629	677550	1208843	115729	1152926	656844
NA50	6283	44495	**931796**	343187	112370	541505	10383	539958	224550
NGA50	7152	48167	**931796**	343187	56486	535333	18123	539958	224550
NA75	2340	20741	**803116**	194532	21391	222892	741	257702	82794
NGA75	2923	28572	**803116**	157104	16359	222892	1412	223942	82794
LA50	196	41	**2**	4	7	3	79	3	6
LGA50	169	34	**2**	4	10	4	53	3	6
LA75	528	85	**4**	9	40	6	887	6	14
LGA75	425	63	**4**	10	54	6	353	7	14

ANEXO A
MÉTRICAS QUAST

3.1.1 Summary report

 # contigs (≥ x bp) is total number of contigs of length ≥ x bp. Not affected by the `--min-contig` parameter (see section 2.4).

Total length (≥ x bp) is the total number of bases in contigs of length ≥ x bp. Not affected by the `--min-contig` parameter (see section 2.4).

All remaining metrics are computed only the contigs that exceed the threshold specified by specified by the `--min-contig` option (see section 2.4, default is 500).

 # contigs is the total number of contigs in the assembly.

Largest contig is the length of the longest contig in the assembly.

Total length is the total number of bases in the assembly.

Reference length is the total number of bases in the reference.

GC (%) is the total number of G and C nucleotides in the assembly, divided by the total length of the assembly.

Reference GC (%) is the percentage of G and C nucleotides in the reference.

N50 is the length for which the collection of all contigs of that length or longer covers at least half an assembly.

NG50 is the length for which the collection of all contigs of that length or longer covers at least half a reference genome.
This metric is computed only if a reference genome is provided.

N75 and NG75 are defined similarly with 75 % instead of 50 %.

L50 (L75, LG50, LG75) is the number of contigs as long as N50 (N75, NG50, NG75)
In other words, L50, for example, is the minimal number of contigs that cover half the assembly.

 # misassemblies is the number of positions in the contigs that satisfy one of the following criteria:

- the left flanking sequence aligns over 1 kbp away from the right flanking sequence on the reference;
- flanking sequences overlap on more than 1 kbp;
- flanking sequences align to different strands or different chromosomes.

This metric requires a reference genome.

 # misassembled contigs is the number of contigs that contain misassembly events.

Misassembled contigs length is the total number of bases in misassembled contigs.

 # local misassemblies is the number of breakpoints that satisfy the following conditions:

1. Two or more distinct alignments cover the breakpoint.
2. The gap between left and right flanking sequences is less than 1 kbp.
3. The left and right flanking sequences both are on the same strand of the same chromosome of the reference genome.

unaligned contigs is the number of contigs that have no alignment to the reference sequence. The value "X + Y part" means X totally unaligned contigs plus Y partially unaligned contigs.

Unaligned length is the total length of all unaligned regions in the assembly (sum of lengths of fully unaligned contigs and unaligned parts of partially unaligned ones).

Genome fraction (%) is the percentage of alinged bases in the reference. A base in the reference is aligned if there is at least one contig with at least one alignment to this base. Contigs from repetitive regions may map to multiple places, and thus may be counted multiple times.

Duplication ratio is the total number of aligned bases in the assembly divided by the total number of aligned bases in the reference (see Genome fraction (%) for the 'aligned base' defenition). If the assembly contains many contigs that cover the same regions of the reference, its duplication ratio may be much larger than 1. This may occur due to overestimating repeat multiplicities and due to small overlaps between contigs, among other reasons.

N's per 100 kbp is the average number of uncalled bases (N's) per 100 000 assembly bases.

mismatches per 100 kbp is the average number of mismatches per 100 000 aligned bases. True SNPs and sequencing errors are not distinguished and are counted equally.

indels per 100 kbp is the average number of indels per 100 000 aligned bases. Several consecutive single nucleotide indels are counted as one indel.

genes is the number of genes in the assembly (complete and partial), based on a user-provided list of gene positions in the reference. A gene 'partially covered' if the assembly contains at least 100 bp of this gene but not the whole one.

This metric is computed only if a reference genome and an annotated list of gene positions are provided (see section 2.4).

operons is defined similarly to # genes, but an operon positions file required instead.

predicted genes is the number of genes in the assembly found by GeneMark.hmm, GlimmerHMM or MetaGeneMark. See the description of the `--gene-finding` option for details.

Largest alignment is the length of the largest continuous alignment in the assembly. A value can be smaller than a value of largest contig if the largest contig is misassembled.

NA50, NGA50, NA75, NGA75, LA50, LA75, LGA50, LGA75 ("A" stands for "aligned") are similar to the corresponding metrics without "A", but in this case aligned blocks instead of contigs are considered. Aligned blocks are obtained by breaking contigs in misassembly events and removing all analigned bases.

3.1.2 Misassemblies report

misassemblies is the same as # misassemblies from section 3.1.1. However, this report also contains a classification of all misassemblies into three groups: relocations, translocations, and inversions (see below).

Relocation is a misassembly where the left flanking sequence aligns over 1 kbp away from the right flanking sequence on the reference, or they overlap by more than 1 kbp, and both flanking sequences align on the same chromosome.

Translocation is a misassembly where the flanking sequences align on different chromosomes.

Inversion is a misassembly where the flanking sequences align on opposite strands of the same

chromosome.

misassembled contigs and misassembled contigs length are the same as the metrics from section 3.1.1 and are counted among all contigs with any type of a misassembly (relocation, translocation or inversion).

local misassemblies is the same as # local misassemblies from section 3.1.1.

mismatches is the number of mismatches in all aligned bases.

indels is the number of indels in all aligned bases.

short indels (≤ 5 bp) is the number of indels of length ≤ 5 bp.

long indels (> 5 bp) is the number of indels of length > 5 bp.

Indels length is the total number of bases contained in all indels.

3.1.3 Unaligned report

fully unaligned contigs is the number of contigs that have no alignment to the reference sequence.

Fully unaligned length is the total number of bases in all unaligned contigs.

partially unaligned contigs is the number of contigs that are not fully unaligned, but have fragments with no alignment to the reference sequence.

with misassembly is the number of partially unaligned contigs that have a misassembly in their aligned fragment. Note that such misassemblies are not counted in # misassemblies and other misassemblies statistics.

both parts are significant is the number of partially unaligned contigs that have both aligned and unaligned fragments longer than the value of `--min-contig`.

Partially unaligned length is the total number of unaligned bases in all partially unaligned contigs.

N's is the total number of uncalled bases (N's) in the assembly.