Supplementary for “inGAP: an integrated next-generation genome analysis pipeline”

Ji Qi, Fangqing Zhao, Anne Buboltz and Stephan C. Schuster

Contents
1. Supplemental figures
   1.1. Supplemental Figure 1. Accuracy comparison between MAQ and inGAP on different data sets with simulated sequencing errors
   1.2. Supplemental Figure 2. An assembly of a 3.5Kb LINE/RTE element from 454 sequenced woolly mammoth genome sequences
   1.3. Supplemental Figure 3. An overview of graphical interface of inGAP
   1.4. Supplemental Figure 4. Whole genome comparison of 8 mitochondrial sequences of woolly mammoth
   1.5. Supplemental Figure 5. An example of contig order predicted by PGA method
2. Supplemental methods
   2.1. Mapping of high-throughput reads
   2.2. Detailed multiple alignment for partitioned windows and difference detection
   2.3. SNPs evaluation by a Bayesian method
   2.4. Comparing difference among multiple genomes
   2.5. Assisting genome assembly
   2.6. Implementation
3. Supplemental results
   3.1. Variant calling and comparison among multiple genomes
   3.2. Comparative genome assembly
4. Supplemental references
1 Supplemental figures

Supplemental Figure 1. Accuracy comparison between MAQ and inGAP on different data sets with simulated sequencing errors (1% for substitutions and 0.2% for indels). (A) Sensitivity rates of MAQ and inGAP on mapping simulated Illumina reads with different divergence from the reference. (B) PPV rates of MAQ and inGAP on different divergent Illumina reads; (C) Sensitivity rates of MAQ and inGAP on different coverage of simulated Illumina reads ranging from 5X to 100X. Green line is the indels predicted by inGAP; (D) PPV rates of MAQ and inGAP on different coverage of reads; (E) Sensitivity rates of inGAP to predict SNP and Indels on simulated 454 reads; (F) PPV rates of inGAP on simulated 454 reads.
Supplemental Figure 2. An assembly of a 3.5Kb LINE/RTE element from 454 sequenced woolly mammoth genome sequences, where the RTE-2_MD from the opossum genome was used a reference. A consensus sequence could be built from alignment of high divergent reads.
Supplemental Figure 3. An overview of graphical interface of inGAP. The user friendly interface provides information of reads coverage of reference on either local or global scale. Detailed alignments are also shown for reviewing or editing by users.
Supplemental Figure 4. Whole genome comparison of 8 mitochondrial sequences of woolly mammoth. Sequence differences were collected by inGAP for building a phylogenetic tree (upper left panel), then each difference is assigned to a node by “latest common ancestor” rule. Multiple alignment of 8 reference sequences close to selected difference (with red frame) is displayed in the lower panel.
Supplemental Figure 5. An example of contig order predicted by PGA method. The upper arrows indicate ordered contigs, their matches on a reference genome are displayed by lower squares.
2 Supplemental methods

As shown in figure 1, we use a three-step strategy for mapping and discovering SNPs, mapping of reads from Roche/454 or Illumina to a reference genome by pairwise aligner, assembling of mapped reads, and finally, investigating differences then evaluated by a Bayesian method.

2.1 Mapping of high-throughput reads

Reads from Illumina are mapped by BWA, a fast short reads aligner. SAM (Sequence Alignment/Map) format is introduced to save detailed sequence, quality and alignment informations. While reads from Roche/454 are much longer than that of Illumina and can be mapped either by BLASTN for high divergent mapping or by BLAT for close related mapping, these alignment results are also saved with SAM format but need fine adjusments in the assembly step. When using BLAST/BLAT as a reads aligner, inGAP provides thresholds of minimum matching percentage of reads and matching identity to filter alignment outputs. The local alignments identify all highly conserved orthologous and repeated matches. For each paired-end read, inGAP maps both ends respectively then chooses the pair of matching with the most proper distance. A paired-end read with only one pair(ends?) matched will be considered as a single-end read for further repeats checking. Repetitive reads are removed if they have identical matches on different loci of the genome.

2.2 Reads assembly and difference detection

inGAP scans the pairwise mapping results of reads obtained in the first step, to collect all possible insertion and deletions. Alignments of reads overlaped with these indels need further fine adjustments.

It is relatively simple to adjust alignment results of BWA by considering to add extra padded segment or not. Insertions and deletions are then reevaluated based on padded alignments, subsitutions are called after determination of indels.

Tabular format of BLAST/BLAT results do not provide detailed information for gaps. In order to bypassing this difficulty, MUSCLE is applied to build multiple alignments for conveniencing reads assembly. The reference genome need to be divided into small segments first by sliding and overlapping windows with given size and step. Reads overlaped with a given window are collected and are sent to MUSCLE for further assembly. inGAP looks through each window to check the differences between reads and reference.

After that the informative substitutions will be further evaluated by a Bayesian guided algorithms for SNP identification. An informative site can be homozygous, heterozygous or pseudo SNP depending on the coverage details and the quality of its supporting base pairs. In this step, the neighborhood quality standard is imported for quality control. All the penetrated informative differences must have good quality neighboring nucleotides on both sides.

2.3 SNPs evaluation by a Bayesian method

A Bayesian algorithm is introduced into inGAP for SNPs identification. It starts from a universal-like Bayesian formula (Marth, et al., 1999) but applied a coarse-graining prior/posterior probabilities. Each informative site is a column in multiple alignment results, which contains inconsistence with the
reference. Given a column, which is covered by $n$ reads on this site, let $O_1, \cdots, O_n$ denotes “observed” $n$ nucleotides of $n$ reads while $E_1, \cdots, E_n$ as the “estimated” nucleotides. Each $E_i$ can equal to $O_i$ or not, depends on if there is a sequencing error.

Each $E_i$ can be one of A, T, C and G so there is total $4^n$ possible $E_1, \cdots, E_n$ permutations. The posterior probability of each $E_1, \cdots, E_n$ is calculated as follows when observed sequence $O_1, \cdots, O_n$ is given.

$$P(E_1, \cdots, E_n \mid O_1, \cdots, O_n) = \frac{\prod_i P(E_i \mid O_i) P_{\text{prior}}(E_1, \cdots, E_n)}{\sum_{\text{each} (E_1, \cdots, E_n)} \prod_i P(E_i \mid O_i) P_{\text{prior}}(E_1, \cdots, E_n)}$$

(1)

where the likelihood $P(E_i \mid O_i)$ is calculated based on quality score $Q_i$ of base,

$$P(E_i \mid O_i) = \begin{cases} 
1 - 10 \frac{Q_i}{10}, & \text{when } E_i = O_i, \\
1 \times 10^{-10}, & \text{when } E_i \neq O_i
\end{cases}$$

(2)

The prior probabilities $P_{\text{prior}}(E_1, \cdots, E_n)$ are hard to estimate because they are various among different data sets. An easy way of calculation $P_{\text{prior}}(E_1, \cdots, E_n)$ is counting from the current data set. We notice that each column $E_1, \cdots, E_n$ could only contain four types of nucleotides (indels will be considered separately) so that all $P_{\text{prior}}(E_1, \cdots, E_n)$ could be classified into four classes depending on the type number of appeared nucleotides in each $E_1, \cdots, E_n$. We say

$$P_{\text{prior}}(E_1, \cdots, E_n) = P_{\text{prior}}(E'_1, \cdots, E'_n)$$

if $E_1, \cdots, E_n$ and $E'_1, \cdots, E'_n$ and the same type number of nucleotides. So $P_{\text{prior}}(E_1, \cdots, E_n)$ could be estimated by

$$P_{\text{prior}}(E_1, \cdots, E_n) = \frac{P(\text{type} = k)}{\text{Permutation}(\text{type} = k)} = \frac{P(\text{type} = k)}{C_d^k \times C_{n-1}^{k-1}}$$

(3)

where $P(\text{type} = k)$ represents the probability of observing a column covered by $k$ types of
nucleotide along the mapping of reference and $P(\text{type} = k)$ satisfies $\sum_{k=1}^{4} P(\text{type} = k) = 1$. In this paper, $P(\text{type} = k)$ is counted from all informative sites obtained from results of multiple alignments and all $P_{\text{prop}}(E_1, \cdots, E_n)$ which have the same type number $k$ equally share the evaluated $P(\text{type} = k)$. Finally, solving the equation (1) by putting (2) and (3) yields

$$P(E_1, \cdots, E_n | O_1, \cdots, O_n) \propto P(E_i | O_i) \times \frac{P(\text{type} = k)}{\binom{4}{k} \times \binom{n-1}{k-1}}$$

(4)

inGAP provides users an option if the input reads are from haploid or diploid strains. For a haploid strain, only the most frequently appeared nucleotide (at least 60% of total coverage) contributes to the consensus and all the rest nucleotide types are considered as sequencing errors. For a diploid strain, the second most frequent type is also taken into account and the informative site can be evaluated as a homozygous or heterozygous SNP depending on the type number, which has the higher posterior probability. Multiple adjacent indels will be merged to check possible copy number difference if they locate in tandem repeated regions. Tandem Repeats Finder (Benson, 1999) is integrated into inGAP to identify tandem repeats across the whole reference. These estimated SNPs are then displayed in the inGAP graphic editor for checking or collecting statistic information. Pseudo SNPs yields from wrong alignment of reads will be removed from the editor.

2.4 Comparing difference among multiple genomes

inGAP can compare multiple genomes and find out the differences between them. First, global alignment are applied to each pair of the genomes by the software MUMMER (Kurtz, et al., 2004) and all the differences between them are recorded. After that, regions along both sides of each difference are searched for homologous matches on all references and are evaluated again by multiple alignments to obtain “core set” SNPs involving all genomes. A maximum likelihood based method, named PHYML (Guindon, et al., 2003), is used to reconstruct a phylogenetic tree for these references based on the “core set” SNPs. These SNPs are displayed by the inGAP editor for further evaluation. Each “core set” SNP is assigned to a node on the tree by the “latest common ancestor” rule.

2.5 Assisting genome assembly

There are a lot of genome assembly softwares available to assemble high-throughput reads into contigs. People usually design primers to fill in the gaps between adjacent contigs to finalize the assembling of the whole genome sequence. PGA (Zhao, et al., 2008) is an efficient algorithm of assisting genome assembly by predicting orders of contigs. inGAP integrates PGA program and is able to automatically align contigs to reference genomes and to predict contig orders, then design primers with the program PRIMER3 (Rozen, et al., 2000) in the last step.
2.6 Implementation

inGAP is implemented in Java and integrates some sequence alignment tools. It provides user-friendly graphical interface for data importing, parameters setting and results reviewing. It can automatically partition large jobs into small parts and distribute them to multiple processors if available. Longer reference genomes, like human chromosomes, can be divided into small contigs according to the options set by user. Different sources of reads, Roche/454 or Illumina, are detected automatically and applied with different filtering parameters. InGAP can detect either homozygous or heterozygous SNPs upon user’s selection.

InGAP is available online at http://sites.google.com/site/nextgengenomics/ingap
3 Supplemental results

3.1 Variant calling and comparison among multiple genomes
In inGAP, we also provided some other functions, including SNP detection from multiple genome comparisons, and contig assembly. Intraspecific polymorphism among multiple genomes may help us link the genetic variation to their phenotypic effect. inGAP can identify strain-specific polymorphisms and also the phylogenetic distribution of these genetic variations among multiple genomes. As shown in Supplemental Figure 3, we used eight complete mitochondrial genomes of woolly mammoth to carry out a whole genome alignment and to detect shared and lineage specific polymorphic sites. The phylogeny on the left panel was constructed based on the identified segregating sites using the PHYML method (ref??). Node size displayed is proportional to the number of shared polymorphic sites by the descendents. The leaf node indicates the strain-specific polymorphism. inGAP provides an aligned sequence for each polymorphic site, which enables users to check the robustness of the sequence alignment and to access its flanking sequences. Moreover, a circular view of the pair-wise comparison is also presented, through which users can easily explore those polymorphisms shared by any two strains of genomes. As shown in Supplemental Figure 3, M21 and M25 shared more polymorphisms and were clustered together on the phylogeny, which well consist with previous study that M21 and M25 form a separate clade of woolly mammoth (Gilbert, et al., 2008).

3.2 Comparative genome assembly
Several de novo assembly tools have been recently reported to assemble dozens, even thousands of contigs from short reads (Butler, et al., 2008; Hernandez, et al., 2008; Warren, et al., 2007; Zerbino, et al., 2008). After initial assembly from reads to contigs has occurred, traditional sequencing approaches usually use linking information from paired-end reads of large-insert genomic libraries to create supercontigs. In the absence of libraries, however, sequences of related organisms can also provide scaffolding information. We have proposed a new pheromone trail based genetic algorithm (PGA) to layout contigs (Zhao, et al., 2008; Zhao, et al., 2008). PGA method can distinguish the optimal connection for each contig from mis-connections with global search heuristics. This method is proved successfully by real incomplete genome data sets produced by Sanger DNA sequencing and pyrosequencing. Here we incorporate this method into the inGAP package. Users can now easily order and orientate contigs, interactively visualize the comparison between contigs and their matches on reference genomes, and also design primers for subsequent gap closing (Supplemental Figure 4).
4 Supplemental references

Benson, G. (1999) Tandem repeats finder: a program to analyze DNA sequences, *Nucleic Acids Res*, 27, 573-580.

Butler, J., MacCallum, I., et al. (2008) ALLPATHS: de novo assembly of whole-genome shotgun microreads, *Genome Res*, 18, 810-820.

Gilbert, M.T., Drautz, D.I., et al. (2008) Intraspecific phylogenetic analysis of Siberian woolly mammoths using complete mitochondrial genomes, *Proc Natl Acad Sci U S A*, 105, 8327-8332.

Guindon, S. and Gascuel, O. (2003) A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood, *Syst Biol*, 52, 696-704.

Hernandez, D., Francois, P., et al. (2008) De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer, *Genome Res*, 18, 802-809.

Kurtz, S., Phillippy, A., et al. (2004) Versatile and open software for comparing large genomes, *Genome Biol*, 5, R12.

Rozen, S. and Skaletsky, H. (2000) Primer3 on the WWW for general users and for biologist programmers, *Methods Mol Biol*, 132, 365-386.

Warren, R.L., Sutton, G.G., et al. (2007) Assembling millions of short DNA sequences using SSAKE, *Bioinformatics*, 23, 500-501.

Zerbino, D.R. and Birney, E. (2008) Velvet: algorithms for de novo short read assembly using de Bruijn graphs, *Genome Res*, 18, 821-829.

Zhao, F., Li, T., et al. (2008) A new pheromone trail-based genetic algorithm for comparative genome assembly, *Nucleic Acids Res*, 36, 3455-3462.

Zhao, F., Zhao, F., et al. (2008) A new pheromone trail-based genetic algorithm for comparative genome assembly, *Nucleic Acids Res*, 36, 3455-3462.