

PanGraphRNA User Manual

(version 1.0)

- PanGraphRNA is an efficient, flexible and web-based Galaxy platform that can be easily used to construct graph pangenomes from genetic variations at individual, subpopulation, and population levels. It can assist researchers to select appropriate graph pangenomes using various performance metrics for both real and simulation experiments.
- Currently, PanGraphRNA is composed of four functional modules: **Graph Pangenome Preparation Module, Construction Module, Evaluation Module, and Application Moudule.**
- PanGraphRNA was powered with an advanced packaging technology, which enables compatibility and portability.
- PanGraphRNA project is hosted on <https://github.com/cma2015/PanGraphRNA>
- PanGraphRNA docker image is available at <https://hub.docker.com/r/malab/pangraphrna>

Graph Pangenome Construction Module and Alignment

This module implements a fast, memory-efficient toolkit HISAT2 to construct graph pangenomes at the individual, subpopulation, or population level. Subsequently, it performs read-genome alignment.

| Tools | Description | Input | Output | Time (test data) | Reference |
|---|--|--|--|---------------------|------------------------|
| Individual Level Graph Pangenome | Construct individual level graph pangenome and perform read-genome alignment | Reference genome in FASTQ format and variation information in VCF format | HISAT2 alignment report in TXT format and alignment result in BAM format | ~10 mins | HISAT2 |

| Tools | Description | Input | Output | Time (test data) | Reference |
|--|---|--|--|------------------|------------------------|
| Subpopulation Level Graph Pangenome | Construct subpopulation level graph pangenome and perform read-genome alignment | Reference genome in FASTQ format and variation information in VCF format | HISAT2 alignment report in TXT format and alignment result in BAM format | ~10 mins | HISAT2 |
| Population Level Graph Pangenome | Construct population level graph pangenome and perform read-genome alignment | Reference genome in FASTQ format and variation information in VCF format | HISAT2 alignment report in TXT format and alignment result in BAM format | ~10 mins | HISAT2 |

For large genomes like maize, graph construction and indexing require memory proportional to genome size and variant density. Incorporating known splicing sites during indexing constrains read boundaries is a strategy of reducing spurious alignments from genome complexity and improving mapping efficiency. In polyploid genomes, reporting additional multimapping loci may help evaluate alignment ambiguity caused by high subgenomic sequence similarity and its impact on downstream analyses. For highly heterozygous genomes, careful variant organization is critical: anchoring the graph on the most representative haplotype as the primary backbone and progressively integrating other variants enhances allelic diversity representation.

Individual Level Graph Pangenome

In this function, an ultrafast and memory-efficient tool **HISAT2** (Kim, D., et al., 2019) is integrated for constructing individual level graph pangenomes and aligning sequencing reads. See <https://daehwankimlab.github.io/hisat2/manual> for details.

Input

- **Input reference genome file:** Input reference genome file for primary path of graph pangenome in FASTA format
- **Input VCF file:** Input VCF file containing variant information to be integrated into the primary path of graph pangenome in VCF format
- **Input FASTQ file:** Cleaned single-end or paired-end RNA-seq reads in FASTQ format

Parameters

- **Accession name:** Input accession name available in the VCF to specify the variant data (Default: 628)
- **Threads:** The number of threads used for parallel computation (Default: 10)
- **VCF prefix:** The prefix of variant records (e.g., var_1125, "var" is the vcf prefix) (Default: var)

Output

- **HISAT2 alignment report in TXT format**
- **HISAT2 alignment result in BAM format**

Step 1: select a reference genome

Step 2: select a VCF file

Step 3: input an accession name

Step 4: choose single-end or paired-end

Step 5: select clean reads

Step 6: click here to run this function

Subpopulation Level Graph Pangenome

In this function, an ultrafast and memory-efficient tool **HISAT2** (Kim, D., et al., 2019) is integrated for constructing subpopulation level graph pangenomes and aligning sequencing reads. See <https://daehwankimlab.github.io/hisat2/manual> for details.

Input

- **Input reference genome file:** Input reference genome file for primary path of graph pangenome in FASTA format
- **Input VCF file:** Input VCF file containing variant information to be integrated into the primary path of graph pangenome in VCF format

- **Input accession name list:** Input accession name list (TXT file) available in the VCF file to specify the variant data
- **Input FASTQ file:** Cleaned single-end or paired-end RNA-seq reads in FASTQ format

Parameters

- **Threads:** The number of threads used for parallel computation (Default: 10)
- **VCF prefix:** The prefix of variant records (e.g., var_1125, "var" is the vcf prefix) (Default: var)

Output

- **HISAT2 alignment report in TXT format**
- **HISAT2 alignment result in BAM format**

Step 1: select a reference genome

Step 2: select a VCF file

Step 3: input an accession list

Step 4: choose single-end or paired-end

Step 5: select clean reads

Step 6: click here to run this function

Population Level Graph Pangenome

In this function, an ultrafast and memory-efficient tool **HISAT2** (Kim, D., et al., 2019) is integrated for constructing population level graph pangenomes and aligning sequencing reads. See <https://daehwankimlab.github.io/hisat2/manual> for details.

Input

- **Input reference genome file:** Input reference genome file for primary path of graph pangenome in FASTA format

- **Input VCF file:** Input VCF file containing variant information to be integrated into the primary path of graph pangenome in VCF format
- **Input FASTQ file:** Cleaned single-end or paired-end RNA-seq reads in FASTQ format

Parameters

- **Threads:** The number of threads used for parallel computation (Default: 10)
- **VCF prefix:** The prefix of variant records (e.g., var_1125, "var" is the vcf prefix) (Default: var)

Output

- **HISAT2 alignment report in TXT format**
- **HISAT2 alignment result in BAM format**

Step 1: select a reference genome

Step 2: select a VCF file

Step 3: choose single-end or paired-end

Step 4: select clean reads

Step 5: click here to run this function

The screenshot shows the AnVIL Population Level Graph Pangenome tool interface. The left sidebar lists various tools and modules. The main panel displays the 'Population Level Graph Pangenome (Galaxy Version 1.0.0)' workflow. The steps are outlined as follows:

- Step 1: select a reference genome** (highlighted in red box): Input reference genome file for primary path of graph pangenome (6: tair10.fasta).
- Step 2: select a VCF file** (highlighted in red box): Input VCF file containing variant information to be integrated into the primary path of graph pangenome (7: 1001all_test.vcf).
- Step 3: choose single-end or paired-end** (highlighted in red box): Single-end or paired-end reads? (single-end).
- Step 4: select clean reads** (highlighted in red box): Input FASTQ file (SE type) (5: SRR1234567.fastq_clean_reads.fastq).
- Step 5: click here to run this function** (highlighted in red box): Execute button.

The right side of the interface shows a history of datasets and files, including:

- 12 : hisat2_alignment_sub_population.bam (on SRR1234567.fastq_clean_reads.fastq)
- 11 : hisat2_alignment_sub_population_summary.txt (on SRR1234567.fastq_clean_reads.fastq)
- 10 : accession_list
- 9 : hisat2_alignment_individual_dual.bam (on SRR1234567.fastq_clean_reads.fastq)
- 8 : hisat2_alignment_individual_dual_summary.txt (on SRR1234567.fastq_clean_reads.fastq)
- 7 : 1001all_test.vcf
- 6 : tair10.fasta
- 5 : SRR1234567.fastq_clean_reads.fastq
- 4 : SRR1234567.fastq_clean_reads.fastq