Aug 30, 2020 Version 3

allele.variability V.3

DOI
dx.doi.org/10.17504/protocols.io.bkhqkt5w

Sara Beier[1], Sara Beier[2]

[1]Lebniz Institute for Baltic Sea Research; [2]Leibniz Institute for Baltic Sea Research

Sara Beier
University of Clermont Auvergne

DOI: https://dx.doi.org/10.17504/protocols.io.bkhqkt5w

Protocol Citation: Sara Beier, Sara Beier 2020. allele.variability. **protocols.io**
https://dx.doi.org/10.17504/protocols.io.bkhqkt5w

Protocol status: Working
We use this protocol and it's working

Created: August 30, 2020

Last Modified: August 30, 2020

Protocol Integer ID: 41232

Keywords: aquatic metagenome time series, temporal variability of the respective gene, predictability of microbial community dynamic, specific patterns for the temporal allele variability, microbial community dynamic, temporal dynamics of gene, gene with highest allele variability, temporal allele variability, microbial community assembly, aquatic prokaryote, gene, highest allele variability, temporal variability of functional unit, increased temporal variability, copy gene, higher temporal variability, effect of the subcellular location, extracellular environment, respective gene, subcellular location, taxonomic unit, gene multiplication, metacommunity size, increased variability, larger metacommunity, variability, maintenance of diversity

Abstract

A prerequisite to improve the predictability of microbial community dynamics is to understand their assembly mechanisms. To study factors that contribute to microbial community assembly, we examined temporal dynamics of genes in five aquatic metagenome time series, originating from marine off-shore or coastal sites and one lake, while focusing on a trait-based data evaluation.
We expected to find gene-specific patterns for the temporal allele variability depending on the metacommunity size of carrier-taxa and variability of the milieu and the substrates that the resulting enzymes are exposed to. In more detail we hypothesized that a larger metacommunity would cause increased temporal variability of functional units, as shown previously for taxonomic units. Furthermore, we hypothesized that multi-copy genes feature higher temporal variability then single-copy genes, because gene multiplication is often the consequence of increased variability in (subtil) changes of substrate quality and quantity. Finally, we hypothesized that direct exposure of proteins to the extracellular environment would result in increased temporal variability of the respective gene compared to intracellular proteins as they would be exposed to highly variable conditions. The first two hypotheses were confirmed in all, while an effect of the subcellular location of gene-products was only seen in three out of the five time series.
The gene with highest allele variability throughout all datasets was an iron transporter, which also represents a target for phage infections. This finding points to the general importance of iron transporter mediated phage infections on the assembly and maintenance of diversity of aquatic prokaryotes.

Troubleshooting

1 removal of nextera adaptors (cutadapt v1.8.3)

Command

Run for all raw data sequence revers and forward read-files from BATS and HOT

Removal of nextera adaptors (Cutadapt v1.8.3)

```
cutadapt -a CTGTCTCTTATA -o ca.file.forward.fastq.gz
file.forward.fastq.gz
cutadapt -a CTGTCTCTTATA -o ca.file.reverse.fastq.gz
file.reverse.fastq.gz
```

2

Command

Run on BATS and HOT output files from step 1 and on LMO rawdata sequence files

Quality trimming (Sickle v1.33)

```
sickle pe \
-f /data/sara/LTG/pacific/BAT1/cutadapt/ca.file.forward.fastq.gz \
-r /data/sara/LTG/pacific/BAT1/cutadapt/ca.file.reverse.fastq.gz \
-t sanger \
-o /data/sara/LTG/pacific/BAT1/sickle/qtrim.file.forward.fastq \
-p /data/sara/LTG/pacific/BAT1/sickle/qtrim.file.reverse.fastq \
-s /data/sara/gesifus.strains/sickle/qtrim.file.unpaired.fastq \
-q 20 -l 50
```

3

Command

Run for SOLA (output files step 2). The presonal referenece database (personal_default_rRNA_DBs.fna) contains the silve rRNA sequences (downloaded 2013)

rRNA removal (SortMeRna v1.9)

```
#interleave reads
merge-paired-reads.sh qtrim.file.forward.fastq
qtrim.file.reverse.fastq file.inter.fastq
#sortmerna
sortmerna --I file.inter.fastq --paired-in -n 2 --db
personal_default_rRNA_DBs.fna --other file.inter.protein -a 20
#unmerge protein data
unmerge-paired-reads.sh file.inter.protein.fastq
file.forward.protein.fastq file.reverse.protein.fastq
```

4

Command

Run for BATS, HOT (using output from step 2, only data from 10 sample days considered in this study) and SOLA (using output from step 3, data fro all available sample days)

Assembly (IDBA-UD v1.1.3)

```
fq2fa --merge --filter qtrim.file.1.fastq qtrim.file.2.fastq
file.merged.fa #interleave paired reads
cat file*.merged.fa > all.merged.fa #concatenate all individual
merged read files of a time series into one file
idba_ud --mink=25 --maxk=99 --step=4 -l all.merged.fa -o IDBAoutput
#run assembly
```

5

Command

Run for BATS, HOT and SOLA using the outpute file from step 4

Gene calling (Prodigal v2.6.1)

```
prodigal -i IDBA.contig.fa -a prod.pep -d prod.fas -o prod.gff -f gff
```

6

Command

run for SOLA (predicted genes from step 5) and LMO (predicted genes from BARM assembly that were different from the public available data translated with table 11). The reference KEGG database (KEGG.faa) was downloaded on 15.05.2016.

Blast annotation (NCBI-BLAST v2.2.31+)

```
makeblastdb -in KEGG.faa -parse_seqids -dbtype prot #create database
for blastp
blastp -db KEGG.faa -query prod.pep -outfmt 6 -num_alignments 1 -
num_threads 16 -out blastout.tab #run blastp
```

7

Command

Diamond-Blast annotation (DIAMOND v0.8.22.84)

Diamond-Blast annotation (DIAMOND v0.8.22.84)

```
diamond makedb --in KEGG.faa -d KEGG #create diamond database
diamond blastp -d KEGG -q prod.pep  --more-sensitive -k 1 -o
diamond.tab  -p 26 #run diamond blast
```

8

Command

run for BATS, HOT, LMO (quality trimmed reads, step 3), SOLA (protein-coding reads, step 4) and MENDOTA (quality trimmed reads provided by collaborators). Assembled contigs were used as reference (BATS, HOT, SOLA: output from step 4; LMO: BARM assembly, MENDOTA: assembly provided by collaborators)

Mapping (Bowtie2 v2.2.9)

```
bowtie2-build contig.fa contigs #build indexed reference file
bowtie2 --very-sensitive-local --no-unal -x contigs -1
qtrim.file.forrward.fastq -2 qtrim.file.reverse.fastq -S file.sam
#for LMO, SOLA, BATS, HOT with paired-end reads
bowtie2 --very-sensitive-local --no-unal -x contigs -U
qtrim.file.fastq -S file.sam #for MENDOTA with single-end reads
```

9

Command

the prod.gff outputfile from step 5 was reformatted using a personal script (convert_prodigal_GFF_to_subread_featureCount_SAF.pl) to prod.saf concerning the requirements of the featureCounts software. For BATS, HOT, LMO, SOLA and MENDOTS sam-files produced during step 7 were used.

summarize mapped reads (Subread v1.4.6)

```
convert_prodigal_GFF_to_subread_featureCount_SAF.pl -g prod.gff -s
prod.saf #converts prodigal output into prod.saf
featureCounts -p -a prod.saf -T 30 -F SAF -o feature.tab *sam #for
LMO, SOLA, BATS, HOT with paired-end reads
featureCounts -a prod.saf -T 12 -F SAF -o feature.tab *sam #for
MENDOTA with single-end reads
```

10

Command

Run for prokaryotic amino acid sequences from the KEGG database (downloaded Nov 2015) representing each KEGG ortrholog (K*.prokaryotes.faa) with the alternative settings -n, -p or -a for sequences affiliating with gram negatives, gram positives or archaea, respectively

Subcellular location (PSORTb v.3.0)

```
psort -n/p/a K*.prokaryotes.faa -o terse |awk -F'\t' '{$1="XXX" FS
$1;}1' OFS='\t'|sed 1d > psort.K*.prok.tab #create psortoutput for
each KEGG ortholog
cat psort.K* >psort.all.prok.tab #concatenate results
```

11

Command

run for BATS, HOT, LMO, MENDOTA, SOLA: input files amino acid sequence files obtained after gene calling (prod.pep)

cluster CDS by 100% amino acid similarity (Cd-hit v4.6)

```
cd-hit -i prod.pep -o out.clust -c 1.00 -n 5 -M 16000 -T 22
```