T0 Paper

Authors: Coco Koedooder,Maxim Rubin-Blum, Yeala Shaked

October 2022

AIM: Analyse Trichodesmium and their associated bacteria from the Red Sea through metagenomes (and proteomes).

R Notebook: Provides reproducible analysis for the metagenomic analysis (and proteomics) that were conducted in the following manuscript:

1. **Metagenomic Pipeline:**

- ATLAS 2 Pipeline - assembly, binning and read coverage per bin

2. **Proteomics Pipeline:**
3. **Phylogenetic Analysis:**

- GTDB-tk taxonomy
- GTO-tree

3. **Proteomic Analysis:**
4. **Functional trait Analysis:**

- MAG annotation - Ghost-Koala (KEGG) and HMMScan (PFAM)
- Vitamin B12 biosynthesis and uptake
- Particle-related Traits
- Siderophore biosynthesis
- Phosphate-uptake
- Nitrogen Metabolism

Citation: Koedooder. (2023) Associated Bacteria of Trichodesmium populations from the Red Sea. https:

GitHub Repository: https:

NCBI BioProject: https:

Accepted for Publication: *Journal* Date

Metagenomic Pipeline: ATLAS2

Metagenomes were assembled and binned using the ATLAS (v2) pipeline (Kieser et al., 2020) which involved the following steps:

- Quality Control through BBTools
- Assembly via metaSPAdes - k-mer length 21, 33, 55, 99, 121 bp
- Binning using MetaBAT 2, MaxBin 2.0 and VAMB.
- Each bin was assessed for completeness and redundancy using CheckM2 v0.1.2
- A non-redundant set of bins is produced using DAS-Tool and dRep - 97.5% ANI cutoff
- Genes from each MAG are predicted using Prodigal.

1. MAG -XX
2. MAG R-XX
3. MAG T-XX.

```
conda activate atlasenv
cd /home/bioinf/Desktop/Data/Coco/FICUS_2 #FICUS

# important:
# do not have other fastq files present in folders within your folder
# ATLAS will attempt to use these as well when creating a tsv file

# important:
# do not skip fastq  (error message/ bug)

atlas init -d /media/bioinf/Data/ATLAS -w /media/bioinf/WD/coco/Morphotypes/reads/ATLAS2 --threads 54 --
# interleaved fastq is not necessary for MORPH Dataset

# make the necessary changes to your .yaml file

ATLAS Ficus - kmer 21,33,55,77,99
            - ANI 97
# run atlas
atlas run -w /home/bioinf/Desktop/Data/Coco/OCEAN/SRR -c /home/bioinf/Desktop/Data/Coco/OCEAN/SRR/config
```

Phylogenetic analysis (taxonomy and phylogenetic tree)

MAGs were taxonomically characterized using the genome taxonomy database tool kit GTDB-tk.

```
#GTDB-tk
conda activate gtd
gtdbtk classify_wf --cpus 24 --extension fasta --genome_dir PATHWAY --out_dir PATHWAY/gtdb
```

The MAGs were further placed in a phylogenetic tree using GTo-Tree which uses a concatenated list of 74 conserved single-copy HMM markers for bacteria.The tree was visualized using FigTree (v1.4.4). The genome of Fuseobacterium nucleatum (PRJNA1419) were downloaded and added to the analysis as an outgroup to re-root the tree (see Coleman et al., 2021). The tree table was visualised using IQ-Tree.

```
#GTO-tree

# Make a list (metadatafile - with all fasta names - doublecheck for strange sampels)
# The sequences MUST be in fasta format
# Save from excel to txt file for correct format
# GTO-tree aligns the FASTA files using a selection of HMMS (e.g 74 HMMs for Bacteria)

conda activate gtotree

gtt-hmms # to know the HMM groups possible

GToTree -f GTO.txt -H Bacteria -j 8 -o Tree #74 genes

conda activate gblocks

#o. file name

Aligned_SCGs.faa

#b. get blocks
# Original alignment: 11237 positions
# Gblocks alignment:  106 positions (0%) in 2 selected block(s)
# Too much - will continue with unfilted alignment

# Construct phylogenetic tree using IQtree2 (without gblocks)
/Users/dustbin/iqtree-2.1.3-MacOSX/bin/iqtree2 -s /Users/dustbin/Desktop/TO_paper/1_Phylogeny/tree/Alig
# Best-fit Model: Q.pfam+R7

iqtree -s /Users/dustbin/Desktop/TO_paper/4_Proteomics/old2new/combi/Tree/Aligned_SCGs.faa --alrt 1000
```

MAG Annotation

To annotate MAGs using pfam and kegg databases - one needs to first concatenate the MAGs into one large text file which is then annotated. Often we will need to rename the files, so its easier to subsequently pull out hits and genes of interest.

```
## Rename MAG fasta files
for f in MAG*.fasta; do mv "$f" "${f/%.fasta/_OLD.fasta}"; done
## concatenate into one file
cat *.faa > concatenated.faa
```

hmmscan (pfam)

```
hmmscan --tblout OLD_hmm.tsv --cut_ga --cpu 8  /Users/dustbin/Bioinf/pfam/Pfam-A.hmm concatenated.faa
```

ghostkoala (kegg)

Upload the concatenated file to **ghostkoala**:https://www.kegg.jp/ghostkoala/):

Metabolic

METABOLIC offers a standardized approach for the annotation of protein function for large datasets, where established databases such as KEGG, Pfam, and SEED/RAST are often highly detailed and often times overwhelming to users.

Metabolic-G Metabolic-C

```
#metabolic-g
#script to print pathway and filename into a text file (that can then be modified)

printf '%s\n' "$PWD"/* > ../filenames.txt

# metabolic-c
# genome coverage to make element cycling pathways
# Place T01 QC-fastq.gz reads in folder - FW and RV
# Run scripts on the raw reads to get abundances

conda activate metabolic

perl METABOLIC-C.pl -t 40 -m-cutoff 0.75  -in-gn  -r omic_reads_parameters.txt -o METABOLIC_out

# make a txt file with the path of the FW and RV reads separated by a ','
# m-cutoff pathway completeness higher than 0.75%
# taxonomy gives additional output - summarising presence according to the 'order' level
# 2nd_run allows you to repeat for metaT data from your genomics data or play with parameters such as y

perl /home/bioinf/bin/metabolic/METABOLIC/METABOLIC-C.pl -t 32 -m-cutoff 0.75  -in-gn /home/bioinf/Deskt

perl /home/bioinf/bin/metabolic/METABOLIC/METABOLIC-C.2nd_run.pl -t 32 -in-gn /media/bioinf/Data12New/Me
```

Presence/ Absence table of MAGs for a list of KEGG IDs of interest

R script that reformats output files of GhostKoala and Excel-Table list MAGs and KEGG IDs of interest

```
#Format a annotation table into a presence and absence table

library(reshape) #melt/transpose data
library(stringr) #get subset
library(dplyr)

# upload files
#KOs of interest file
#MAG names
#MAGs <- read.delim("Project_Julia.csv", header=TRUE, sep=",", dec=".") #data table with (240) KOs of i
MAGs <- read.delim("Project_Ficus.csv", header=TRUE, sep=",", dec=".") #KO annotations of all MAGs (wit

#KO annotation files
Koala <- read.delim("GhostKOALA_FICUS.csv", header=TRUE, sep=",", dec=".") #KO annotations of all MAGs

#reformat data for R - by transposing (melting) with columns 1 and 2 untouched
MAGs_long <- melt(MAGs, id = c(1, 2),
                  measured = c(3, ncol(MAGs)))
```

```
#rename column to KO
names(MAGs_long)[3] <- "KO"

#remove 'empty' value column
MAGs_long <- MAGs_long[, -4]

#make a new column "MAG ID dataset" - where column 1 and column 2 are pasted together, separated by "_"
#this column will be used to match names with the Koala data-table
MAGs_long$MAG_ID_dataset <- paste(MAGs_long$MAG_ID, MAGs_long$Dataset, sep="_")
#transposed data - results in 11424 rows

#In the Koala data-table - make a new column - that selects the first 11 characters
#Now both tables have a matching column that can be used as common identifier
Koala$MAG_ID <- str_sub(Koala$MAG,1,5)

#Make a new table from KOALA that only keeps unique identifiers
Koala_unique <- unique(Koala[, c(2, 3)])
#Koala goes from 194619 rows down to 72625

#Make a column called "presence" where those that have a KOALA ID match - get a 1 (present)
Koala_unique$presence <- 1
Koala_unique$KO <- as.factor(Koala_unique$KO)

#Now we match the KOs with the KO IDs of interest present in MAGs_long (column 3 - KO ID number)
Koala_unique_goodKOs <- unique(inner_join(MAGs_long[3], Koala_unique))
#2648 matches of the 72625

#Now we add a column of presence/ absence
MAGs_final <- full_join(MAGs_long, Koala_unique_goodKOs)
#to change NAs to 0s (see script below)
#MAGs_final$presence <- ifelse(is.na(MAGs_final$presence), 0, MAGs_final$presence)

#now we get back the transposed data to its original format (where each KO represents a column)
#we should get back the 241 KOs of interest in each column
MAGs_final_transposed <- cast(MAGs_final,
                              MAG_ID + Dataset ~ KO,
                              value = "presence")

#write table to csv
write.csv(MAGs_final_transposed, "List_FICUS_FINAL.csv")
```

Siderophores

NRPS Siderophores

FeGenie

FeGenie was used to assess the NRPS-type siderophore biosynthesis pathways

```
#### FeGenie
```

```
conda activate fegenie
pwd
FeGenie.py -bin_dir /Users/dustbin/Desktop/T0_paper/2_Iron_Siderophore/FeGenie -bin_ext fasta -out //Use
```

NIS Siderophores

NIS domains were assessed through the presence of an IucA/IucC domain.

```
# NIS-Type (IUCA/IUCC domain)
# PF04183
# confirm with ANTISMASH for IucC/IucA and RAST

grep -c PF04183 /Users/dustbin/Desktop/T0_paper/2_Iron_Uptake/FICUS_hmm.tsv #7
    grep PF04183 /Users/dustbin/Desktop/T0_paper/2_Iron_Uptake/FICUS_hmm.tsv | cut -f 3,5,18 > /Users/du
```

NIS-genes were then clustered into different types as described by Carroll and Moore (2018).

MAFFT

IQtree2

```
## MAFFT ############
https://www.ebi.ac.uk/Tools/msa/mafft/
NIS_all_aligned.fasta

# Construct phylogenetic tree using IQtree2 (without gblocks)

/Users/dustbin/iqtree-2.1.3-MacOSX/bin/iqtree2 -s /Users/dustbin/Desktop/T0_paper/2_Iron_Siderophore/NIS

# Best-fit Model: Q.pfam+F+R4 chosen according to BIC
# Branch selection using Shimodira-Hasegawa
# 1000 Bootstraps
```

Proteomics