Phylogenetic pipeline

to infer a species/genome tree from a set of genomes

2020/2021, GP AGO
Project

Species

Species selected to this experiment were all viruses with their proteomes in Uniprot database, from *Coronaviridae* family. It's a family of enveloped, positive-strand RNA viruses which infect amphibians, birds, and mammals. The viral genome is 26–32 kilobases in length (resulting in approximately 10 proteins). The particles are typically decorated with spikes on their surface, which in electron micrographs of spherical particles create an image reminiscent of the solar corona (Figure 1).

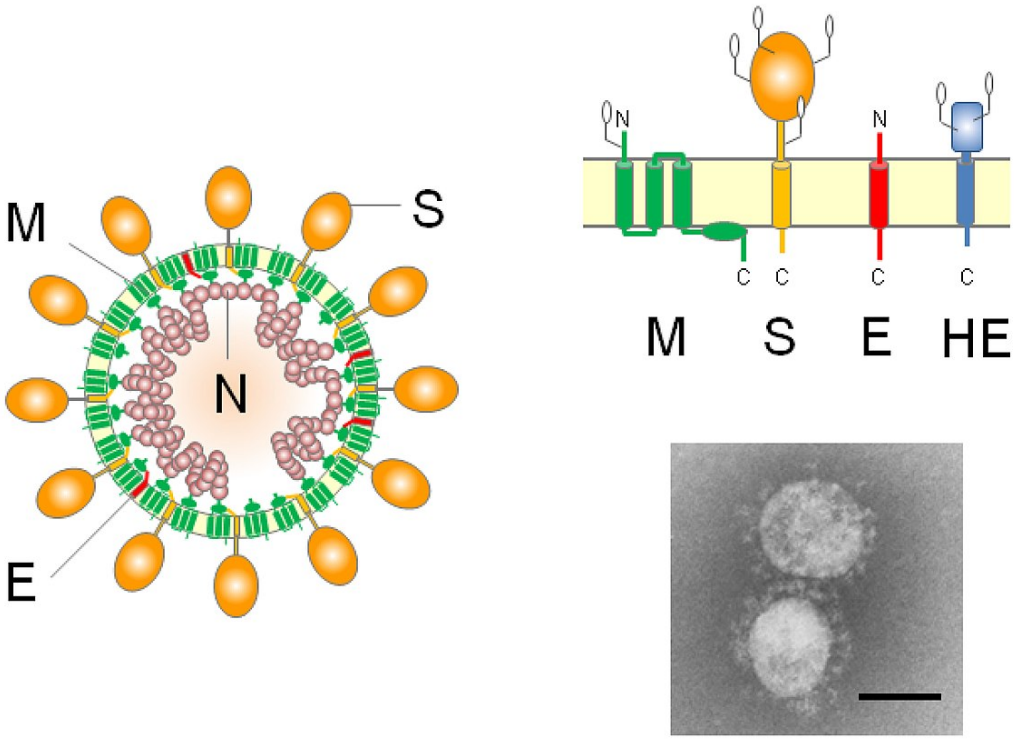

Figure 1. *Coronaviridae* viruses organization

Their main taxonomy is as follow (bolded most common):

Coronaviridae

> ***Orthocoronavirinae***

> ***Alphacoronavirus***

> *Colacovirus*

> *Decacovirus*

> *Duvinacovirus*

> *Luchacovirus*

> *Minacovirus*

> ***Minunacovirus***

> *Myotacovirus*

> *Nyctacovirus*

> *Pedacovirus*

> *Rhinacovirus*

> *Setracovirus*

> *Soracovirus*

> *Sunacovirus*

> ***Tegacovirus***

> ***Betacoronavirus***

> ***Embecovirus***

> *Hibecovirus*

> ***Merbecovirus***

> *Nobecovirus*

> ***Sarbecovirus***

> ***Gammacoronavirus***

> *Brangacovirus*

> *Cegacovirus*

> ***Igacovirus***

> ***Deltacoronavirus***

> *Andecovirus*

> ***Buldecovirus***

> *Herdecovirus*

Pipeline

For all species from *Coronaviridae* family one proteome per species (selected by score) was downloaded from Uniprot database. Then their proteomes were filtered if needed (in case that there were too small/too large). After that, all proteomes were merged and clustered to separated protein families. Then all those clusters were aligned within and saved as FASTA files. Maximum-likelihood, neighbor-joining and maximum-parsimony trees were built from those families. At the end, for each kind of built trees, consensus tree (if possible) and super tree was built.

Pipeline splits into two separated stages after clustering, one stage is performed on all sequences and species for all clusters, second stage builds trees filtering clusters before to guarantee one-to-one correspondence for at least 1/5 species (configurable through arguments).

In results, first pipeline is performed on:

clusters **68/174**, species **282/282**, records **2691/2934**

Second pipeline is performed on:

clusters **4/174**, species **78/282**, records **312/2934**

When correspondence is one-to-one, only 4 clusters are selected. They contain main coronaviruses proteins, such as: the biggest cluster (protein family) is made for 3C-like protease which is the main protease in coronaviruses and corresponds to nonstructural protein 5 (nsp5). It is a multifunctional protein: it contains the activities necessary for the transcription of negative stranded RNA, leader RNA, subgenomic mRNAs and progeny virion RNA as well as proteinases responsible for the cleavage of the polyprotein into functional products.

Execution time (bottlenecks)

All steps which requires a lot of execution time are paralleled (if possible). All times between steps in pipeline are measured and show to stdout and dump to log file. For main steps in pipeline run used to obtain results (without correspondence guaranteed), times are:

1. Downloading proteomes: 14 min
2. Clustering sequences: 17 s
3. **[Paralleled]** Aligning sequences: 7 min
4. **[Paralleled]** Building NJ trees: 45 s
5. **[Paralleled]** Building ML and MP trees: 7 h 15 min
6. Build consensus trees: -
7. **[Paralleled]** Build super trees: 6 h 10 min

For one-to-one correspondence first 2 steps are the same:

1. Downloading proteomes: 14 min
2. Clustering sequences: 17 s
3. **[Paralleled]** Aligning sequences: 2 min 15 s
4. **[Paralleled]** Building NJ trees: 3 s
5. **[Paralleled]** Building ML and MP trees: 53 min
6. Build consensus trees: 3 s
7. **[Paralleled]** Build super trees: 21 min

After first look, bootlenecks (even with parallel computation) are clearly identifiable for both stages (marked with red background): building ML trees (MP trees are built simultaneously, but from my tests are not very time demanding) and building super trees (it's worth mentioning that heuristics algorithm was run with parameters resulting in pretty exhaustive search).

Tools and methods

Some steps in pipeline were performed using external tools:

Clustering

Clustering were performed using *mmseqs2* software, which is very fast and easy-to-use tool when it comes to clustering, with following parameters:

mmseqs2 *easy-cluster <merged fasta file>*

Aligning

Aligning fasta files with protein families were performed using *muscle* software through *biopython* package, which based on input and output file, returns command line for performing aligning. Software were used without any additional parameters:

muscle *-in <merged fasta file> -out <aligned fasta file>*

Building NJ trees

To build neighbor-joining trees, very fast (claiming to be the fastest) tool was used - *ninja*. It was also run without any additional arguments:

ninja *—in <aligned fasta file> —out <neighbor-joining tree>*

Building ML and MP trees

For building maximum likelihood and parsimony trees, standalone version of RAxML was used. Software were built from sources with enabled support for AVX registers and pthreads library. Substitution model is estimated from data GTR model:

raxml -s <aligned fasta file> -w <output dir> -m PROTGAMMAGTR

Consensus trees

Consensus trees were obtained using *biopython Phylo* package. Used method is *majority_consensus* without cutoff, so trees are relaxed binary consensus trees:

Bio.Phylo.majority_consensus(*<list of input trees>***)**

Super trees

Super trees were built using *clann* command line tool. Scoring criterium used for obtaining best tree was “Most Similar Supertree Method”, which is default option which takes unrooted binary trees as input. Algorithmic procedure is described in manual at page 15 (https://raw.githubusercontent.com/ChrisCreevey/clann/master/Docs/Clann3%20manual.pdf). Pipeline allows two options when it comes to building super trees:

1) **standard search** - uses heuristics search: “Nearest neighbor interchange” (Figure 2) which is not very computational demanding, with following parameters:

hs swap=**nni** maxswaps=100000 nreps=5 weight=equal

2) **super-search** - uses more complex heuristics search: “Subtree pruning and regrafting” (Figure 3) which is more time consuming, but may give better results, with following parameters:

hs swap=**spr** maxswaps=500000 nreps=3 weight=equal

Results were obtained using **super-search** pipeline option. Parameters were adjusted to balance execution time and results scores.

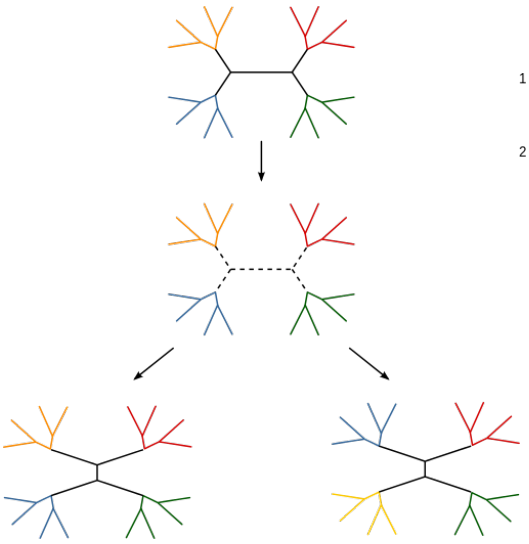

Figure 2. Nearest neighbor interchange (NNI)

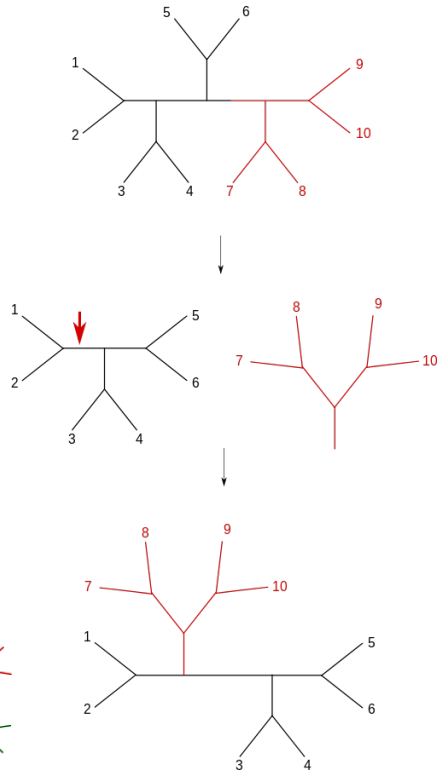

Figure 3. Subtree pruning and regrafting (SPR)

Usage

Pipeline is very easy to use, user needs to provide only *json* file with list of species to analyze, or species family name. All options are described while running pipeline with *-h* option.

Because nowadays a lot of bioinformatics software are lack of maintenance and requires specific software under specific locations etc., pipeline is built and ran using Docker, so it's platform independent and can be used on any environment which supports Docker without installing any additional software, setting special variables etc.

Usage options and examples are exhaustively described in *README.md* file attached to this report.

Results

For each method: NJ, ML, MP consensus and super trees were prepared. First stage of pipeline, does not guarantee one-to-one correspondence, thus consensus trees for viruses could not be done. When it comes to super trees, they were compared to each other using Robinson-Foulds distance with following results:

	super-NJ	super-ML	super-MP
super-NJ	0.0	-	-
super-ML	0.896	0.0	-
super-MP	0.946	0.946	0.0

Table 1. Super trees (from all available 282 Coronaviridae species) RF distances between

For trees built with guaranteed one-to-one correspondence, consensus trees were also made:

	super-NJ	super-ML	super-MP	cons-NJ	cons-ML	cons-MP
super-NJ	0.0	-	-	-	-	-
super-ML	0.787	0.0	-	-	-	-
super-MP	0.720	0.627	0.0	-	-	-
cons-NJ	0.733	0.613	0.560	0.0	-	-
cons-ML	0.823	0.701	0.646	0.673	0.0	-
cons-MP	0.760	0.640	0.560	0.520	0.592	0.0

Table 2. Super and consensus trees (from one-to-one Coronaviridae set - 78 species) RF distances between

As we can see, different methods give very different results. It may also be cause of pretty big dataset, because as we can see, for all species (Table 1), differences are much higher between super trees, than for species smaller subset.

After building trees, file with Coronaviridae taxonomy was downloaded, parsed and used to annotate species.

One-to-one correspondence

It turned out, that all selected organism to one-to-one correspondence trees, were *Betacoronaviruses*. In 4 selected clusters there were 4 protein families: **membrane protein**, **3C-like proteinase**, **RNA synthesis protein** and **spike protein**. Those proteins are very common in Coronaviridae, but selected family for probably membrane protein, spike protein, or mix of them (3C-like proteinase and RNA synthesis proteins should be in vast majority of Coronaviridae) must have been specific for *Betacoronaviruses*.

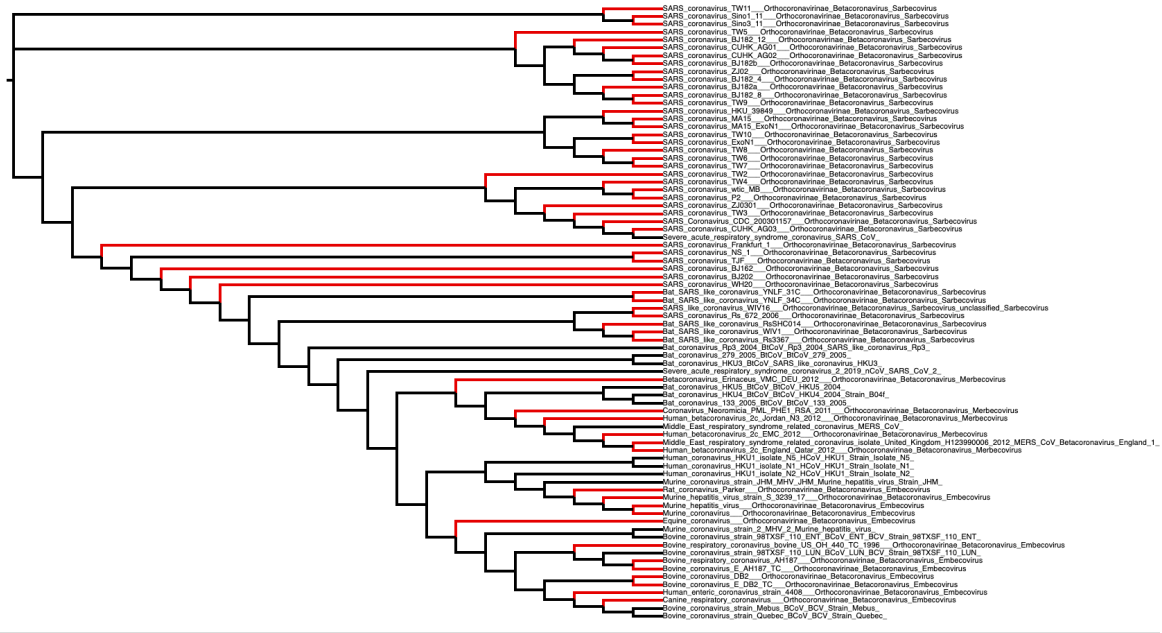

Figure 4. ML super tree (one-to-one correspondence), Betacoronaviruses marked with red color, organisms with black edges were not classified as Betacoronavirus because of lack of taxonomy annotation

In all trees of both types: consensus and super, for every method we can observed three main subspecies (*Embecovirus*, *Merbecovirus*, *Sarbecovirus*), grouped in different clades from *Betacoronavirus* family (rest are unannotated viruses):

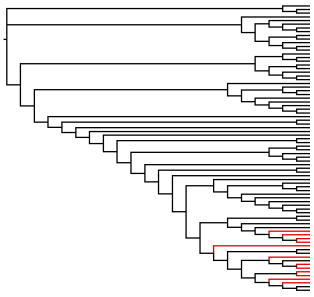

Figure 5. ML super tree: Embecovirus

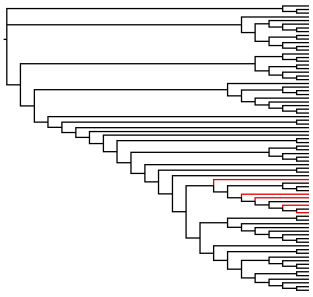

Figure 6. ML super tree: Merbecovirus

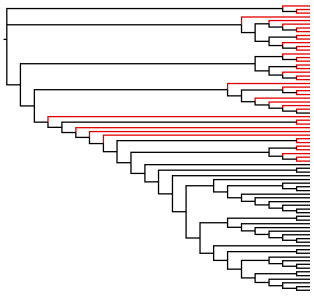

Figure 7. ML super tree: Sarbecovirus

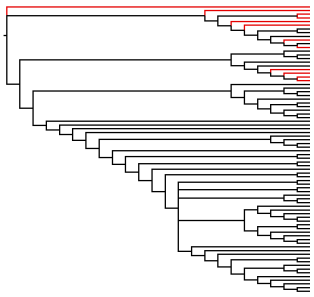

Figure 8. ML consensus tree: Embecovirus

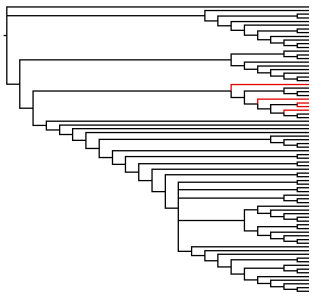

Figure 9. ML consensus tree: Merbecovirus

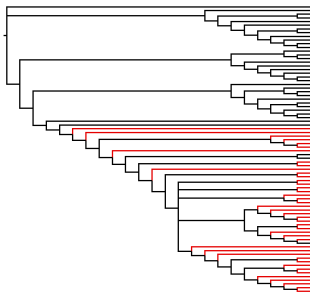

Figure 10. ML consensus tree: Sarbecovirus

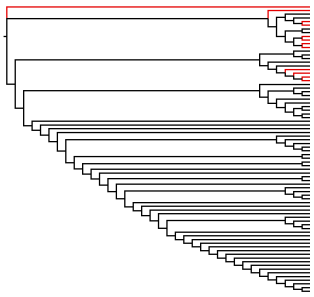

Figure 14. MP consensus tree: Embecovirus

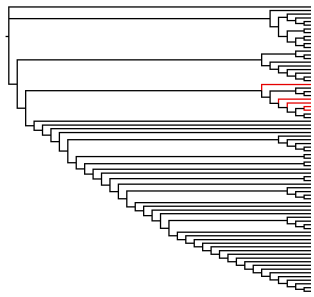

Figure 15. MP consensus tree: Merbecovirus

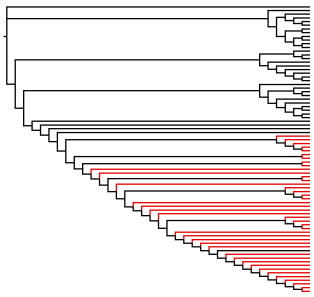

Figure 16. MP consensus tree: Sarbecovirus

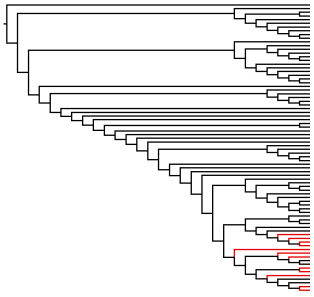

Figure 11. MP super tree: Embecovirus

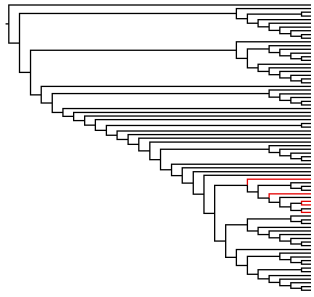

Figure 12. MP super tree: Merbecovirus

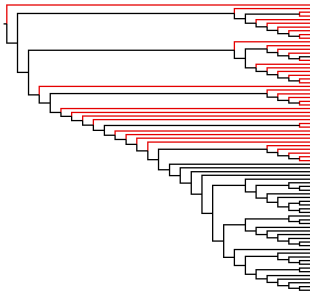

Figure 13. MP super tree: Sarbecovirus

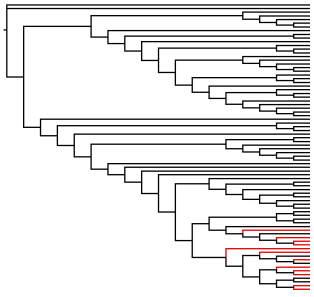

Figure 17. NJ super tree: Embecovirus

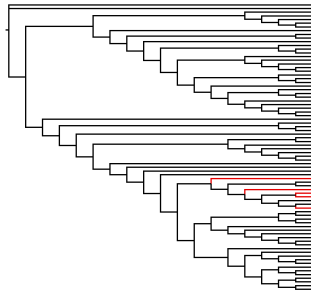

Figure 18. NJ super tree: Merbecovirus

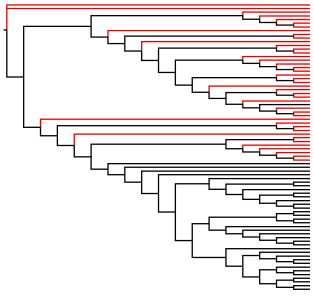

Figure 19. NJ super tree: Sarbecovirus

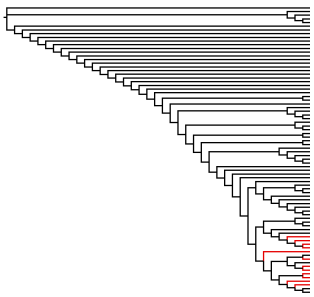

Figure 20. NJ consensus tree: Embecovirus

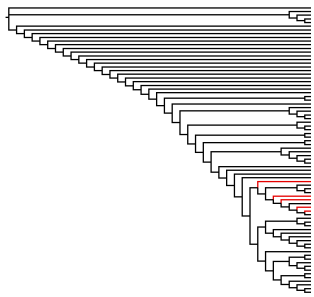

Figure 21. NJ consensus tree: Merbecovirus

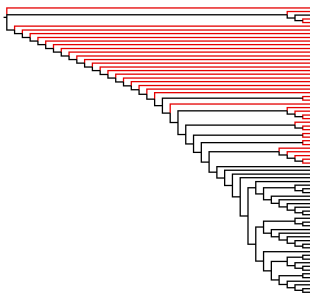

Figure 22. NJ consensus tree: Sarbecovirus

From figures above we could also super trees have much more multi-organisms separated clades than consensus trees.

When it comes to super trees built from all *Coronaviridae*, without guaranteed one-to-one correspondence, we can find there representatives not only from *Betacoronavirus* family, but also from *Alphacoronavirus*, *Gammacoronavirus* and *Deltacoronavirus*.

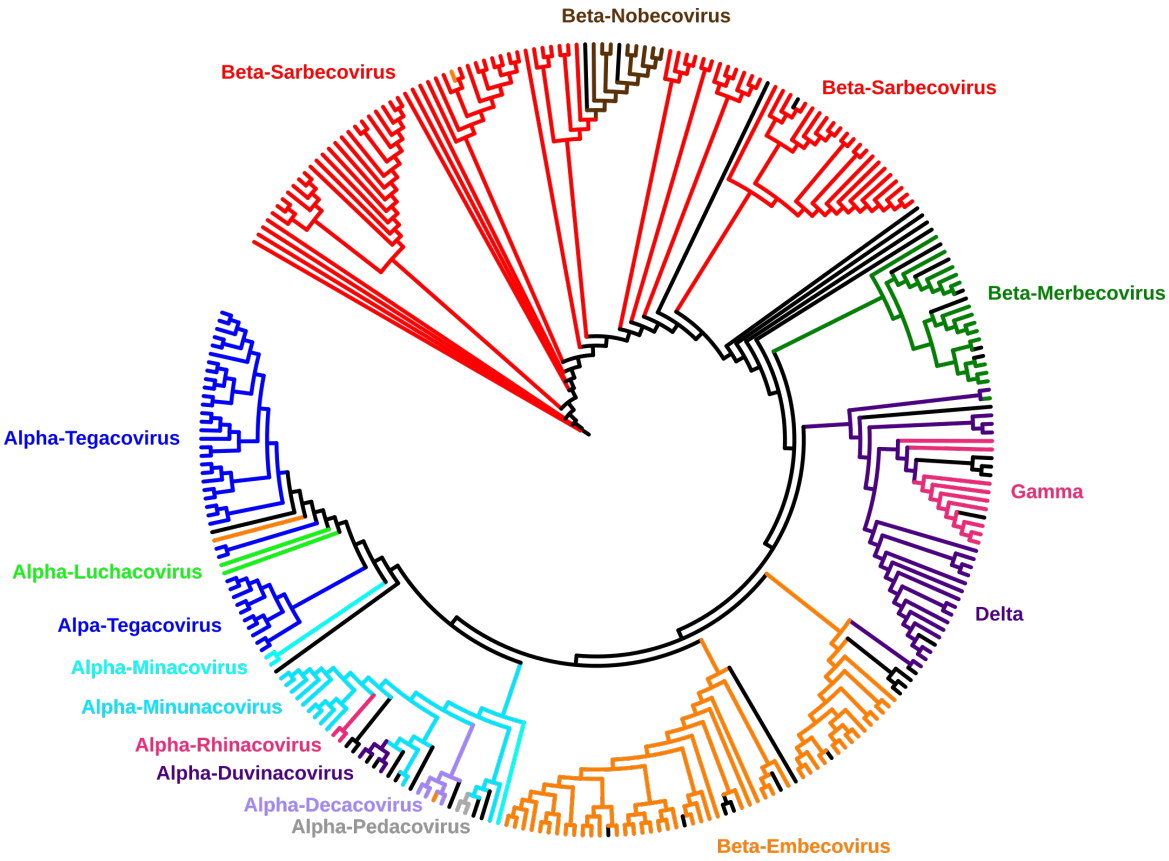

Figure 14. ML super tree from all available Coronaviridae

From tree above, we can observe that nearly all species were grouped correctly in separated clusters. Interesting is that *Gamma* clade is recognized as sub-clade in *Delta* family and *Delta* family as sub-clade in *Beta* family. *Gamma* and *Delta* are shown without granularity, because there were only individual organisms that belong to lower taxonomy families.

We can also compare single method, i.e. neighbor-joining between super trees and super trees with duplications allowed (paralogs), keeping in mind that viruses don't have many duplications because of their small and not so complex proteomes. With duplications allowed there were 50 single copy trees and 18 multicopy trees (without duplications: 68 trees with single copy).

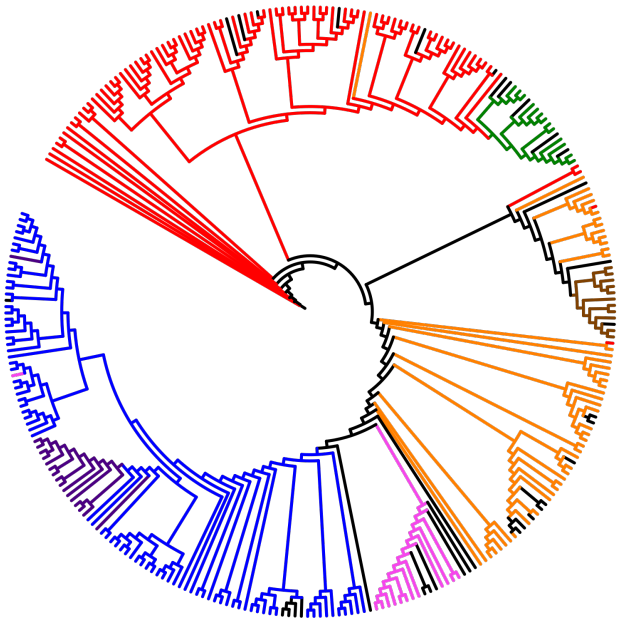

Figure 15. NJ tree from all species with duplications enabled (colors from Figure 14)

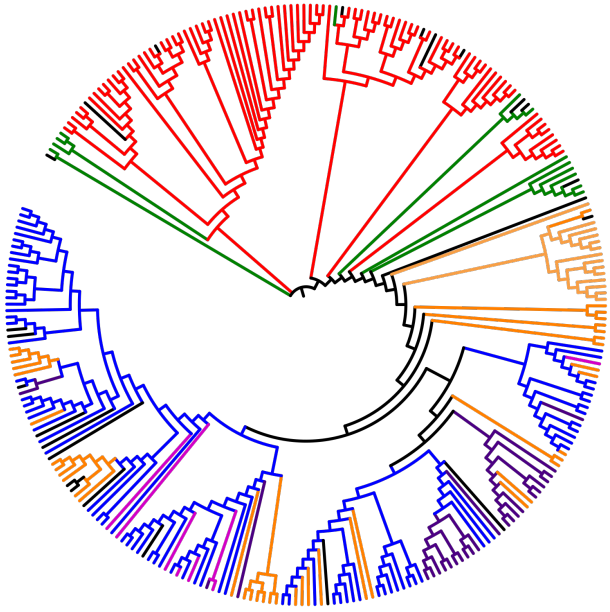

Figure 16. NJ tree from all species without duplications (colors from Figure 14)

By comparing two trees above, we can observed that with duplications (Figure 15), results are much better than without (Figure 16). NJ super tree with duplications seems to be very similar to ML super tree without duplications with only rearranged positions of *Gamma* and *Delta* families (and *Beta-Nobecovirus* within *Beta-Embecovirus* sub-family, instead of *Beta-Sarbecovirus*) and computing it much faster than ML super tree.

On the other side, NJ super tree without duplications seem to separating only *Betacoviruses* from *Alphacoviruses* and others mixed together.

Summary

It's very hard task to obtain reliable and accurate taxonomy tree for viruses, especially for RNA viruses such as *Coronaviridae*, because of their endurance to lethal mutations [1], high mutation rates [2] and many others mechanisms making their genomes highly differ.

Different methods resulted in highly different results, but nearly all of them made accurate clades for different types of Coronaviridae (Alpha, Beta, Gamma, Delta). When it comes to one-to-one correspondence, all methods properly recognized sub-taxonomy families of Betacoronavirus family.

NJ super tree without duplications was much less precise than its counterpart with duplications giving similar results as ML super tree without duplications.

Both methods: NJ (with and without duplications) and ML classified *Severe acute respiratory syndrome coronavirus 2 2019 nCoV SARS CoV 2* in the same clade as *Betacoronvirus -> Nobecovirus* and *Severe acute respiratory syndrome SARS CoV* as member of clade with *Betcoronavirus -> Sarbecovirus*.

Unfortunately, I didn't manage to compare my trees with Coronaviridae tree from timetree.org, because website is currently not working (Internal server error 500).

[1] Chen P, Shakhnovich EI. Lethal mutagenesis in viruses and bacteria. *Genetics*. 2009;183(2):639-650. doi:10.1534/genetics.109.106492

[2] R. Sanjuan, P. Domingo-Calap Machanisms of viral mutation. *Cell Mol Life Sci*. 2016; 73(23): 4433–4448. doi: 10.1007/s00018-016-2299-6