

# RAxML-NG - Randomized Axelerated Maximum Likelihood – Next Generation

<https://github.com/amkozlov/raxml-ng>

**Linux binary:** [https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng\\_v1.2.1\\_linux\\_x86\\_64.zip](https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng_v1.2.1_linux_x86_64.zip)

**macOS Binary:** [https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng\\_v1.2.1\\_macos\\_x86\\_64.zip](https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng_v1.2.1_macos_x86_64.zip)

**Apple M1 binary:** [https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng\\_v1.2.1\\_macos\\_M1.zip](https://github.com/amkozlov/raxml-ng/releases/download/1.2.1/raxml-ng_v1.2.1_macos_M1.zip)

**Windows:** use Linux binary via [Windows Subsystem for Linux \(WSL\)](#)

## 1. Alignment check

```
./raxml-ng --check --msa test.fas --model GTR+G --prefix T1
```

## 2. ML tree estimation (10 random and 10 parsimony starting trees)

```
./raxml-ng --msa test.fas --model GTR+G --prefix T2 --threads 2 --seed 2
```

## 3. bootstrapping (Felsenstein bootstrap – fbp)

```
./raxml-ng --bootstrap --msa test.fas --model GTR+G --prefix T3 --seed 2 --threads 1
```

## 4. mapping the BS support values onto the best-scoring/best-known ML tree

```
./raxml-ng --support --tree T2.raxml.bestTree --bs-trees T3.raxml.bootstraps --prefix T4 --threads 1
```

## 5. ML tree + bootstrapping (both fbp and tbe, i.e., Transfer Bootstrap Expectation) in one command

```
./raxml-ng --all --msa test.fas --model GTR+G --prefix T5 --seed 2 --threads 1 --bs-metric fbp,tbe
```

## 6. ML tree estimation with partitions

First create a file with partition definition and the model you want to assign for that partition, e.g., `partitions.txt`:

GTR+I+G4, ITS = 1-500

TVM+I+G4, matK = 501-1100

Then run the analysis:

```
./raxml-ng --msa test.fas --model partitions.txt --prefix T6 --threads 2  
--seed 2
```