Documentation

1 - Obtaining Coala

Coala can be downloaded directly from our site.

2 - System requirements

Coala was developed in https://www.java.com/en/ and tested in the **Linux** and **Mac OS X** environments.

Additionally, in order to perform the clustering of the accepted vectors and produce graphical plots, Coala requires the installation of https://www.r-project.org/ and of the R packages **dynamicTreeCut** and **ade4**.

3 - Using Coala

The general command line for using Coala is:

```
java -jar Coala.jar -input <nexusfile> [options]
```

We recommend the use of the Java option -Xmx for increasing the maximum memory available for Coala. For example, to set the maximum amount of memory to 3 Gigabytes, use the following command line:

```
java -Xmx3g -jar Coala.jar -input <nexusfile> [options]
```

Here we list all options that are offered by the software:

-a1 <value>

Defines the value of the constant α_1 which is used to compute the metrics "LEAVES AND MAAC" or "EVENTS AND MAAC". *Default value = 0.5 (for "LEAVES AND MAAC") or 0.7 (for "EVENTS AND MAAC")*.

-a2 <value>

Defines the value of the constant α_2 which is used to compute the metrics "LEAVES AND MAAC" or "EVENTS AND MAAC". *Default value = 0.5 (for "LEAVES AND MAAC") or 0.3 (for "EVENTS AND MAAC")*.

-ac <value>

Defines the value of $\alpha_{\text{co-speciation}}$ for the Dirichlet Process on the first round of the ABC-SMC process (sampling of the space of probability vectors). *Default value = 1.0*.

-ad <value>

Defines the value of $\alpha_{\text{duplication}}$ for the Dirichlet Process on the first round of the ABC-SMC process (sampling of the space of probability vectors). *Default value = 1.0*.

-al <value>

Defines the value of $\alpha_{\text{host-switch}}$ for the Dirichlet Process on the first round of the ABC-SMC process (sampling of the space of probability vectors).

Default value = 1.0.

-as <value>

Defines the value of α_{loss} for the Dirichlet Process on the first round of the ABC-SMC process (sampling of the space of probability vectors).
Default value = 1.0.

-cluster

If this option is present, the list of accepted vectors produced in the last round of the ABC-SMC process will be clustered using a hierarchical clustering procedure implemented by the R package **dynamicTreeCut**.

-continue

If this option is present, the software tries to continue a job that was started previously and interrupted in the middle. Coala looks for the output files which are present in the output directory and starts the process from the last executed round. If no output file is found, Coala starts the process from the beginning. Notice that, to use this option, the software must be configured with the same options of the job that was previously interrupted.

-ct <number>

Defines the cyclicity test implementation that will be used to avoid time inconsistent scenarios during the generation of the trees:

1. Stolzer *et al.* 2012 [1]
2. Donati *et al.* 2014 [2]
3. Tofigh *et al.* 2011 [3]

Default cyclicity test = 2.

-discard <factor>

Defines the multiplier factor that is used to compute the maximum number of vectors which can be discarded during the generation of the quantile populations (rounds greater than 1). The maximum number of discarded vectors **D** is defined as: **D = factor × Q**, where **Q = N × threshold_first_round**. Notice that **D** must be an integer greater than 1 (**D** > 1).
Default factor = 10.

-h

If this option is present, Coala prints a help describing all available options and exits.

-input <nexusfile>

Defines the path for the nexus file which contains the pair of host and parasite trees and their associations.

-M <value>

Defines the number of trees which are going to be produced for each probability vector.
Default value = 1000.

-maxtree <factor>

Defines the multiplier factor that is used to compute the maximum number of trees which are going to be simulated in order to obtain the required number of trees **M** (trees that are more than 2 times bigger than the "real" parasite are discarded). The maximum number of trees **X** is defined as: **X = factor × M**. Notice that the factor must be greater than 1 (factor > 1).
Default factor = 5.

-metric <number>

Defines the metric that will be used in the comparison between real and simulated trees. Currently, Coala offers three options:

1. MAAC
2. LEAVES AND MAAC
3. EVENTS AND MAAC

Default metric = 3.

-N <value>

Defines the number of probability vectors which are going to be sampled in the first round of the ABC-SMC process. *Default value = 2000*.

-p <value>

Defines a perturbation limit that is going to be applied to each element p_i of a probability vector v in the refinement phases of the ABC-SMC process (rounds 2, 3, ...). During the perturbation routine, each probability p_i receives an increment delta_i that is uniformly sampled from the interval [-value,+value]. After that, the new vector v' is normalised such that the sum of all elements is equal to 1.

Default value = 0.01.

-plot

If this option is present, Coala will produce plots with the results of each round.

-R <value>

Defines the number of rounds of the ABC-SMC process. *Default value = 3 (If a value different from the default value is chosen, the option -t must be specified.)*

-root <value>

Root mapping probability (real value in the interval [0.5,1.0]). This probability value is used in the recursive process that chooses the starting position during the simulation of parasite trees. Starting from the root of the host tree, we generate a random number and compare it to the given probability value. If the random number is smaller than the probability value, the root of the host tree is chosen as starting point. Otherwise, we choose one of the two subtrees of the host root node to continue the recursion. Notice that, to make a choice between the two subtrees, we attribute to each one a probability value which is proportional to the size of their leaf set.

Default value = 1.0

-t <value>

Defines a vector of tolerance values which are going to be used in each round of the ABC-SMC process. The list of tolerance values is composed by real numbers between 0 and 1, separated by commas (,). The size of this list must be equal to the number of rounds (option -R).

Default value = 0.10,0.25,0.25.

-threads <value>

Defines the number of threads that are going to be used to simulate parasite trees.

Default value = 1.

4 - Input File

Coala receives a NEXUS file as input. The input file must contain a pair of trees (one host tree and one parasite tree) and the association of their leaves. Notice that, in the current version of the software, one parasite leaf cannot be associated to more than one host leaf. The opposite is allowed: a host leaf can be associated to more than one parasite leaf.

Coala can read two types of Nexus files (**.nex**). A description of both can be found here: http://team.inria.fr/erable/files/2020/11/Input-File.pdf.

Please, verify that your file meets the format description.

5 - Software output

During its execution, Coala produces some output files to register intermediate and final results. Given an input file **file.nex**, the program will produce the following files:

file.nex.simul.round_X.csv

This file contains the list of probability vectors which were simulated during round **X**. It is a csv file which contains one line per probability vector and 5 columns: probability of co-speciation, probability of duplication, probability of host-switch, probability of loss, and observed distance.

file.nex.accep.round_X.csv

This file contains the list of probability vectors which were accepted by the ABC rejection method at round **X**. It is a csv file which contains one line per probability vector and 5 columns: probability of co-speciation, probability of duplication, probability of host-switch, probability of loss, and distance observed.

file.nex.plots.round_X.pdf

This PDF file contains a set of histograms which describe the results of round **X**. The plots show the distribution of the:

- Probabilities of each event type among all simulated probability vectors (first row);
- Probabilities of each event type among the accepted probability vectors (second row);
- Distances observed among all simulated probability vectors and among the accepted probability vectors (third row).

This file is produced only if the option -plot is specified in the command line.

file.nex.clust.round_X.Y.csv

This file contains a list of the probability vectors which were accepted during round **X** and were grouped together in the cluster **Y**. It is a csv file which has 6 columns: vector identifier, probability of co-speciation, probability of duplication, probability of host-switch, probability of loss, and observed distance. Additionally, the file contains statistics summaries (**Min**, **Q1**, **Med**, **Mean**, **Q3**, and **Max**) for each column and two proposals of representative probability vectors: one considering the average of each event probability (row **NMean**) and the other considering the median of each event probability (row **NMed**).

This file is produced only if the option -cluster is specified in the command line.

file.nex.clusters.round_X.pdf This file contains a plot which shows the projection of the clusters (of the list of accepted vectors during round **X**) on a plane.

This file is produced only if the options -cluster and -plot are specified in the command line.

Observation: All csv files use the TAB character (\t) as column separator.

6 - Example

Command line:

- java -jar Coala.jar -input file.nex -cluster -plot

Input file:

- file.nex:
    - Example of a Jane Nexus file:

```
#NEXUS
BEGIN HOST;
TREE HOST = ((A,B),((D,E),C));
ENDBLOCK;
BEGIN PARASITE;
TREE PARASITE = (((e,(c2,c1)),(d,b)),a);
ENDBLOCK;
BEGIN DISTRIBUTION;
RANGE
        e : E, c2 : C, c1 : C, d : D, b : B, a : A;
ENDBLOCK;
```

    - Example of a CoRe-Pa Nexus file:

```
#NEXUS
BEGIN TAXA;
        DIMENSIONS NTAX = 11;
        TAXLABELS
                A
                B
                D
                E
                C
                e
                c2
                c1
                d
                b
                a
                ;
ENDBLOCK;

BEGIN TREES;
```

```
        TRANSLATE
                H0      H0,
                H1      H1,
                H2      A,
                H3      B,
                H4      H2,
                H5      H3,
                H6      D,
                H7      E,
                H8      C,
                P0      P0,
                P1      P1,
                P2      P2,
                P3      e,
                P4      P3,
                P5      c2,
                P6      c1,
                P7      P4,
                P8      d,
                P9      b,
                P10     a
                ;
        TREE HOST = ((H2,H3)H1,((H6,H7)H5,H8)H4)H0;
        TREE PARASITE =
(((P3,(P5,P6)P4)P2,(P8,P9)P7)P1,P10)P0;
ENDBLOCK;

BEGIN COPHYLOGENY;
[RANKS represents the ranks of the nodes in the tree]
[Syntax is: nodename timezone_from timezone_to]
        RANKS
                H0      0       0,
                H1      0       0,
                A       0       0,
                B       0       0,
                H2      0       0,
                H3      0       0,
                D       0       0,
                E       0       0,
                C       0       0,
                P0      0       0,
                P1      0       0,
                P2      0       0,
                e       0       0,
                P3      0       0,
                c2      0       0,
                c1      0       0,
                P4      0       0,
                d       0       0,
                b       0       0,
                a       0       0
                ;
[PHI represents the associations from the parasite leaf
nodes to the host leaf nodes]
[Syntax is: parasite_leaf_name host_leaf_name]
        PHI
                e       E,
                c2      C,
                c1      C,
                d       D,
                b       B,
                a       A
                ;
[RECONSTRUCTIONEVENTS represents the events which occurred
in the reconstruction. It includes 'COSPECIATION',
'DUPLICATION', 'EXTINCTION', 'SORTING' and 'HOSTSWITCH']
[Syntax is: operation cost]
        RECONSTRUCTIONEVENTS
                COSPECIATION    -1,
                DUPLICATION     -1,
                SORTING -1,
                HOSTSWITCH      -1
                ;
[RECONSTRUCTION represents the associations from the
parasite nodes to the host nodes as they occured in the
reconstruction/simulation]
[Syntax is: parasite_leaf_name host_leaf_name]
        RECONSTRUCTION
                ;
[POSITIONS represents the x and y position of the node]
[Syntax is: nodename x_position y_position]
        POSITIONS
                H0      10      100,
                H1      85      25,
                A       110     0,
                B       110     50,
                H2      60      150,
                H3      85      125,
                D       110     100,
                E       110     150,
                C       110     200,
                P0      355     125,
                P1      330     150,
                P2      280     200,
                e       230     250,
                P3      255     175,
                c2      230     200,
                c1      230     150,
                P4      255     75,
                d       230     100,
                b       230     50,
                a       230     0
                ;
[COSTS represents the cost table for the operations
'COSPECIATION', 'DUPLICATION', 'EXTINCTION', 'SORTING' and
'HOSTSWITCH']
[Syntax is: operation cost]
        COSTS
                COSPECIATION    0,
                DUPLICATION     2,
                SORTING 1,
                HOSTSWITCH      3
                ;
[OPTIONS represents the options for calculating the
reconstruction]
[Syntax is: option value]
        OPTIONS
                RANK    0,
                LEAFSPECIACIONCOST      0,
                CHECKCHRONOLOGY 0,
                AUTOMATICCOSTS  0,
                SORTING 1,
               PROBABILITYCOSTS       0,
               HOSTSWITCH     1,
               AUTOMATICMETHOD2,
               RANDOMSEED     auto,
               DUPLICATION    1,
               ROOTMAPPING    0,
               FULLHOSTSWITCH 0,
               RANDOMCYCLES   5000
               ;
ENDBLOCK;
```

Output files:

- The output files may be recovered here: http://team.inria.fr/erable/files/2020/11/Output-Files.tar .gz.

References

1. M. Stolzer, H. Lai, M. Xu, D. Sathaye, B. Vernot and D. Durand. **Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees**. *Bioinformatics*, 28(18):i409-i415, 2012.

2. B. Donati, C. Baudet, B. Sinaimeri, P. Crescenzi, and M.-F. Sagot. **EUCALYPT: Efficient tree reconciliation enumerator**. *Algorithms for Molecular Biology*, 2014, *(in press)*.

3. A. Tofigh, M. Hallett and J. Lagergren. **Simultaneous identification of duplications and lateral gene transfers**. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 8(2):517-535, 2011.