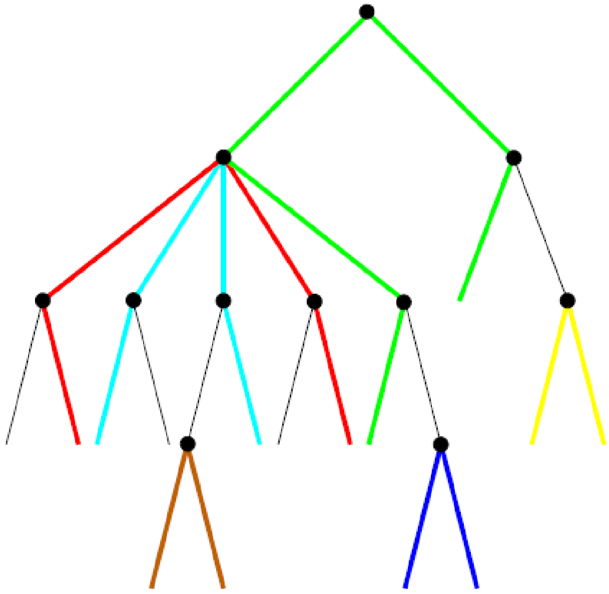
The TARGETING software - An efficient solver for the Maximal Pairing Problem on arbitrary trees

***Christian Arnold*[1,2*]*, and Peter F. Stadler*[1,3,4,5,6]**

1 Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics,
University of Leipzig, Härtelstraße 16-18, D-04107 Leipzig, Germany
2 Harvard University, Department of Human Evolutionary Biology, Peabody Museum, 11 Divinity Avenue,
Cambridge MA 02138, USA
3 Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, D-04103 Leipzig, Germany
4 Fraunhofer Institute for Cell Therapy and Immunology, Perlickstraße 1, D-04103 Leipzig, Germany
5 Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA
6 Institute for Theoretical Chemistry, University of Vienna, Währingerstraße 17, A-1090 Wien, Austria

***to whom correspondence should be addressed (achristian@bioinf.uni-leipzig.de)**



Documentation

Last updated: April 2010

This document provides documentation for the TARGETING software. The newest version of the software is available at the following URL:

http://bioinf.uni-leipzig.de/Software/Targeting

If you have questions or comments, feel free to contact me, Christian Arnold (achristian@bioinf.uni-leipzig.de). I will be happy to answer any questions related to this project or the implementation.

If you use this software, please cite the following reference:

Arnold, **C. and Stadler, PF. 2010. Polynomial algorithms for the Maximal Pairing Problem: efficient phylogenetic targeting on arbitrary trees. *Algorithms for Molecular Biology*. In review.**

Content

1. Description

Abstract for the following publication:

Arnold, C. and Stadler, PF. 2010. Polynomial algorithms for the Maximal Pairing Problem: efficient phylogenetic targeting on arbitrary trees. *Algorithms for Molecular Biology*. In review.

Background

The Maximal Pairing Problem (MPP) is the prototype of a class of combinatorial optimization problems that are of considerable interest in bioinformatics: Given an arbitrary phylogenetic tree T and weights ω_{xy} for the paths between any two pairs of leaves (x, y), what is the collection of edge-disjoint paths between pairs of leaves that maximizes the total weight? Special cases of the MPP for binary trees and equal weights have been described previously. Algorithms to solve the general MPP are still missing, however.

Results

We describe a relatively simple dynamic programming algorithm for the special case of binary trees. The general case of multifurcating trees can then be treated by interleaving solutions to certain auxiliary *Maximum Weighted Matching* problems with an extension of this dynamic programming approach, resulting in an overall polynomial-time solution of complexity O(n^4 *log n*) w.r.t. the number n of leaves. For binary trees, we furthermore discuss several constrained variants of the MPP as well as a partition function approach to the probabilistic version of the MPP.

Conclusions

The algorithms introduced here make it possible to solve the MPP also for large trees with high-degree vertices. This has practical relevance in the field of comparative phylogenetics and, for example, in the context of phylogenetic targeting, i.e., data collection with resource limitations.

We implemented the polynomial algorithms for the MPP in the program TARGETING. The TARGETING progam is written in C and uses Ed Rothberg's implementation [1] of the Gabow algorithm [2] to solve the *Maximum Weight Matching Problem* on general graphs. The software provides a user-friendly interface and can also solve the special weight variants (see Section "Variants" in the publication).

2. Installation

Requirements: C Compiler (tested with gcc) and GNU Make (should be installed on all Linux and MAC systems by default)

Installation of the software is easy and fast. Simply follow the following instructions.

- Copy source file in target directory
- Unzip the *TARGETING_v1.0.tar.gz* archive
```
$ tar -xzvf TARGETING_v1.0.tar.gz
```
- Change to TARGETING_v1.0 directory
```
$ cd TARGETING_v1.0
```
- Compile TARGETING
```
$ make
```
- Compile the w-match software from Ed Rothberg
```
$ cd w-match
$ make
$ cd ..
```
- If the installation was successful, you should have an executable file called "MPP". You should now be able to run the TARGETING software using the provided example files by typing
```
$ ./MPP
```

3. Input

Required files:

A tree file with a phylogenetic tree in NEWICK format is required and must be specified.

Supported formats and format specifications

The NEXUS format is not yet supported. The tree file must have only a single line with the NEWICK tree. If you get an error while parsing the file, make sure that there is no line break after the tree definition. The NEWICK string must furthermore end with ";" (as specified in the NEWICK format). Edge weights, internal branch labels and other kind of labelling will be ignored during the parsing procedure. It may be the case, though, that you get a parsing error. In that case, ensure that the NEWICK string has no such features. For example, the following NEWICK string (this is the example file that is provided) is valid and will be parsed without any problems:

```
(((raccoon:19.19959,bear:6.80041):0.84600,((sea_lion:11.99700,seal:12.00300
):7.52973,((monkey:100.85930,cat:47.14069):20.59201,weasel:18.87953):2.0946
0):3.87382):1,dog:25.46154);
```

The following tree, however, will not work, because it spans multiple lines and the last character is not ";":

```
(((raccoon:19.19959,bear:6.80041):0.84600,((sea_lion:11.99700,
seal:12.00300):7.52973,((monkey:100.85930,cat:47.14069):20.59201,
weasel:18.87953):2.09460):3.87382):1,dog:25.46154)
```

Representation of polytomies

Another important issue is the representation of polytomies. Poytomies (or multifurcations) must be represented as a so-called *hard polytomy*. That is, the polytomy must not be specified as a series of 0-branches (*soft polytomy*). Compare the following examples:

```
1. (((A:1.0,B:1.0):0.0,C:1.0):0.0,D:1.0):1.0;
2. (A:1.0,B:1.0,C:1.0,D:1.0):1.0;
```

The first tree has a series of 0-branches to define the polytomy. The TARGETING software will interpret this simply as a fully bifurcating tree, thus ignoring that in fact, polytomies are present in the tree. This may change the optimal path-system substantially. The second tree, however, uses a representation that is interpreted as a true polytomy by the TARGETING software.

Example file

An example tree file is provided in the main folder. The file is called "tree.tre".

Optional files:

Optionally, a scores file can be specified, that overwrites the default weights for particular pairs of species.

Supported formats and format specifications

The scores file may have multiple lines. On each line, there must be three columns, as follows:

1. The name of the first species that forms the pair
2. The name of the second species that forms the pair
3. The weight of that particular species pair

If a particular species name in the scores file is not found in the tree (i.e., if this name is not in the NEWICK tree file), the TARGETING software throws an error and exits. Thus, ensure that all species names occur in the tree file as well. If a particular pair is not listed in the scores file, the default weight as specified by the *mode* parameter will be used. Thus, it is possible to modify only a particular set of pairs instead of all possible ones.

The weight can be any arbitrary number. We nevertheless recommend using weights with not more than 2 positions after the decimal point, because the maximal weighted matching program that the TARGETING software uses can handle only integer weights. For the maximal weighted matching problems, we convert non-integer weights by multiplying them with a multiple of 10 (depending on the value of the --*prec* parameter, see next section). For example, if the --*prec* parameter is set to 2 (the default value), all weights are multiplied with 10^2 = 100 and then rounded, which converts for example 1.56 to 156 or 1.115 to 112. The latter example illustrates that the --*prec* parameter is important if weights with more than 2 positions after the decimal point are used. In such a case, increase the --*prec* parameter accordingly.

Note that approximated weights are only used if at least one weight in the scoring file is a floating point number. This conversion takes furthermore only place in the maximal weighted matching problems (which use the program of Ed Rothberg), internally, there is no approximation. Thus, it may be the case that due to the rounding, a wrong maximal weighted matching is returned. This should be very rare, but it is theoretically possible. By increasing the –*prec* parameter, however, this problem can be circumvented in most cases.

Example file

An example tree file is provided in the main folder. The file is called “scores.in”.

4. Syntax

To run the program, type

```
$ ./MWM [OPTIONS]
```

<u>OPTIONS</u>

--tree = *{path to tree file}*

<u>Description:</u> the path to the tree file in NEWICK format (see Section “Input” for details)

<u>Abbreviated form:</u> -t

<u>Default value:</u> “tree.tre”, this file should also be in the *“src”* directory

--scores = *{path to scores file}*

<u>Description:</u> the path to the scores file (see Section “Input” for details)

<u>Abbreviated form:</u> -s

<u>Default value:</u> no scores file, default weight of 1 for all pairs

--mode = {0|1|2}

<u>Description:</u> This option corresponds to some of the special weight variants as described in the publication. If no scores file is specified, mode=0 and mode=1 have the same effect: all pairwise weights between two species take the default value 1. This is equivalent to finding an optimal path-system that maximizes the number of pairs. This case may be of practical use under certain circumstances, as it maximizes the number of independent measurements, thus improving power of subsequent statistical tests. If mode=2, however, the software will find an optimal path-system that maximizes the number of edges that are covered. This is achieved by setting Ω_{xy} = d(x,y), where d(x,y) is the graph-theoretic distance, i.e., we interpret the edge lengths in the tree as unity.

If a scores file is specified, however, setting mode=1 or mode=2 forces the program to use the special weight variants, thus discarding the scores file altogether.

<u>Abbreviated form:</u> -m

<u>Default value:</u> 0

--output = {*path to output*}

Description: the path to the output file

Abbreviated form: -o

Default value: none (output is printed to stdout)

--prec = {*2-10*}

Description: the precision for the maximum weighted matching problems (see the end of section 3 for more details)

Abbreviated form: -p

Default value: 2

5. Output

The output is a list of pairs of species that have been paired in the optimal path-system. The first and second column list the two species that are paired (the names are drawn from the tree file, as specified by the "--*tree*" parameter), and the third column shows the weight of that particular pair, as specified by the "--*scores*" and "--*mode*" options, respectively.

The output is either written to a file, as specified in the option *"--output"*, or to *stdout* if no output file has been specified or if the output file cannot be opened.

The output for the example file with default weights is, for example, as follows:

```
Species 1   Species2    Weight
Weasel      dog         1.000000
Raccoon     bear        1.000000
sea_lion    seal        1.000000
monkey      cat         1.000000
```

6. Examples

If you have a tree file called *"tree.example"*, a scores file *"scores.example"*, and if you want to write the output to a file called *"output.example"*, simply type:

```
$ ./MPP –-tree tree.example --scores scores.example –-output output.example
```

You could also type:

```
$ ./MPP -t tree.example -s scores.example -o output.example
```

If you have no scores file, omit the –scores parameter, and default weights of 1 for all species pairs will be used. You can also enforce using default weights with *“--mode 1*”, even if a scores file is provided. This will, as stated above, maximize the number of pairs that are selected. If your goal is to maximize the number of edges, however, type *“--mode 2”*.

7. Copyright

The TARGETING software is published under the GNU Public License. For more details, see *http://gnu.org/licenses/gpl.html* or the file “GNU Public License.txt” in the “*src”* folder.

8. Bugs and Feature Requests

Please report any bug that you encounter as well as any feature request that you may have to Christian Arnold (achristian@bioinf.uni-leipzig.de).

9. References

[1] Rothenberg E: Solver for the Maximum Weight Matching Problem 1999. [http://elib.zib.de/pub/Packages/mathprog/matching/weighted/].

[2] Gabow H: Implementation of Algorithms for Maximum Matching on Nonbipartite Graphs. *PhD thesis*, Stanford University 1973.