trimAl v1.2 - Other Benchmarks

1. trimAl v1.1 benchmark analyses.

In order to test the general applicability of trimAl 1.1, as well as to find an empirical base to set the heuristics for the automatic selection of parameters we performed a benchmark analysis. For this purpose we used a benchmark set that has been used previously to test the improvement in phylogenetic performance after an alignment trimming phase. This set comprises evolutionary simulations of protein sequences of various lengths (400 to 3200 positions), performed with ROSE along phylogenetic trees with 16 tips. These trees have three different topologies varying in their level of symmetry, and whose branch lengths were multiplied by 0.5, 1 and 2 respectively, totaling 9 different phylogenetic trees.

To measure the improvement in phylogenetic reconstruction after running trimAl in the alignments we applied a standard phylogenetic analysis pipeline to each simulated sequence set. This included multiple sequence alignment with MUSCLE, ClustalW or MAFFT and Neighbor Joining or Maximum Likelihood phylogenetic tree reconstruction using PhyML, in all possible combinations.

Before the phylogenetic analyses, multiple sequence alignments were trimmed using different parameter sets. The accuracy of the resulting trees was measured by comparing them with the original trees used to generate the sequence sets, and measuring the Robinson Foulds distance (merely topological distance) and the K-tree score (a distance measure that includes branch-lengths). The trees produced by the complete alignment were also compared with the original trees.

For the trimmed alignments, we observed an overall improvement of the phylogenetic accuracy both in terms of topology (Robinson Fould distance) and of combined topology and branch lengths (K-tree score). Similarly to what has been previously described for Gblocks the level of improvement in phylogenetic accuracy was higher in the case of the asymmetric trees and increased with the length of the branches in the seed tree and the length of the alignment. The improvement was also clearer in the case of alignments performed with ClustalW, rather than MUSCLE or MAFFT. A similar situation has been observed for Gblocks, this time when comparing ClustalW and MAFFT. We interpret this result as evidence for a higher quality of the alignments produced by methods such as MUSCLE and MAFFT that include refinement phases. We focused on the improvement of alignments based in these algorithms to benchmark our different heuristics. Panels summarizing the results obtained in the benchmark trees are included in the following figures.

8 figures (figures S1 to S8) divided into two subsets. The first subset groups results by Robinson Foulds Distance metric meanwhile the second subset groups the results by Ktree Distance metric. Each figure corresponds to a possible combination among the metric system, the tree topology, the phylogenetic tree reconstruction method, Neighbor Joining (NJ) or Maximum Likelihood (ML),and the programs, Muscle or Mafft, to construct the multiple sequence alignment. Panels within a figure represent the three different evolutionary divergence of the seed tree used to generate those alignments (0.5, 1 or 2).

In each panel, x-axis represents the average length of the sequences in the alignment, whereas the y-axis represents the system metric, the Robinson Foulds for the first four figures and the Ktree for the second four ones distances. The Robinson Foulds distance measures the topological difference between two given tree, therefore, lower values indicate a better performance of the alignment when reconstructing the tree while the Ktree distance measures the topological and branch length differences between two given tree, therefore, lower values indicate a better performance of the alignment when reconstructing the tree.

Finally, there are three lines that represents the performance of each method: untrimmed alignments (blue), old trimAl's relaxed method (orange) and old trimAl's strict one (turkey blue). Note that in trimAl 1.2, the relaxed method has been eliminated and the strict method correspond to gappyout. Strict method in trimAl 1.2 has no correspondence with any method in trimAl 1.1

2. Development of the heuristic method: Automated1 (trimAl v1.2).

In our previous benchmarks we detected that, when using Maximum Likelihood in the phylogenetic reconstruction, different automatic method would provide better results depending on the underlying scenario considered (see extended benchmark trimAl v1.2). To address this we measured the relationship of the alignment properties with the simulation scenario to set up a series of heuristic rules to automatically apply the most adequate method.

For this, we took two values for each alignment in our dataset. Firstly, we measured the identity score (see supplementary material in the online documentation) among the sequences in the alignment and compute the average of these values. Secondly, we computed the average of the identity score between the most related pairwise sequences for each sequence in the alignment.

With the information measured, we derived two figures (figures S9 and S10). Each figure corresponds to the identity score measures for all possible combinations among the number of sequences, the tree topology and the tree divergence from our extended dataset (see the online documentation). In the first of them, we measured the average identity score among sequences in the alignment and in the second one; we measured the average identity score between the most related pairwise sequences for each sequence in the alignment.

3. Extended Benchmark: trimAl v1.2. Results based on topology.

These figures are complementary to those shown in the supplementary material of trimAl v1.2 publication and show the complete results for all the methods applied.

4 figures (figures S11 to S14). Each figure corresponds to a possible combination between the phylogenetic tree reconstruction method (Neighbor Joining or Maximum Likelihood) and the tree topology (Asymmetric or Symmetric). Panels within a figure represent the nine different combinations of the number of sequences in an alignment (16, 32 and 64) and the evolutionary divergence of the seed tree used to generate those alignments (0.5, 1 or 2).

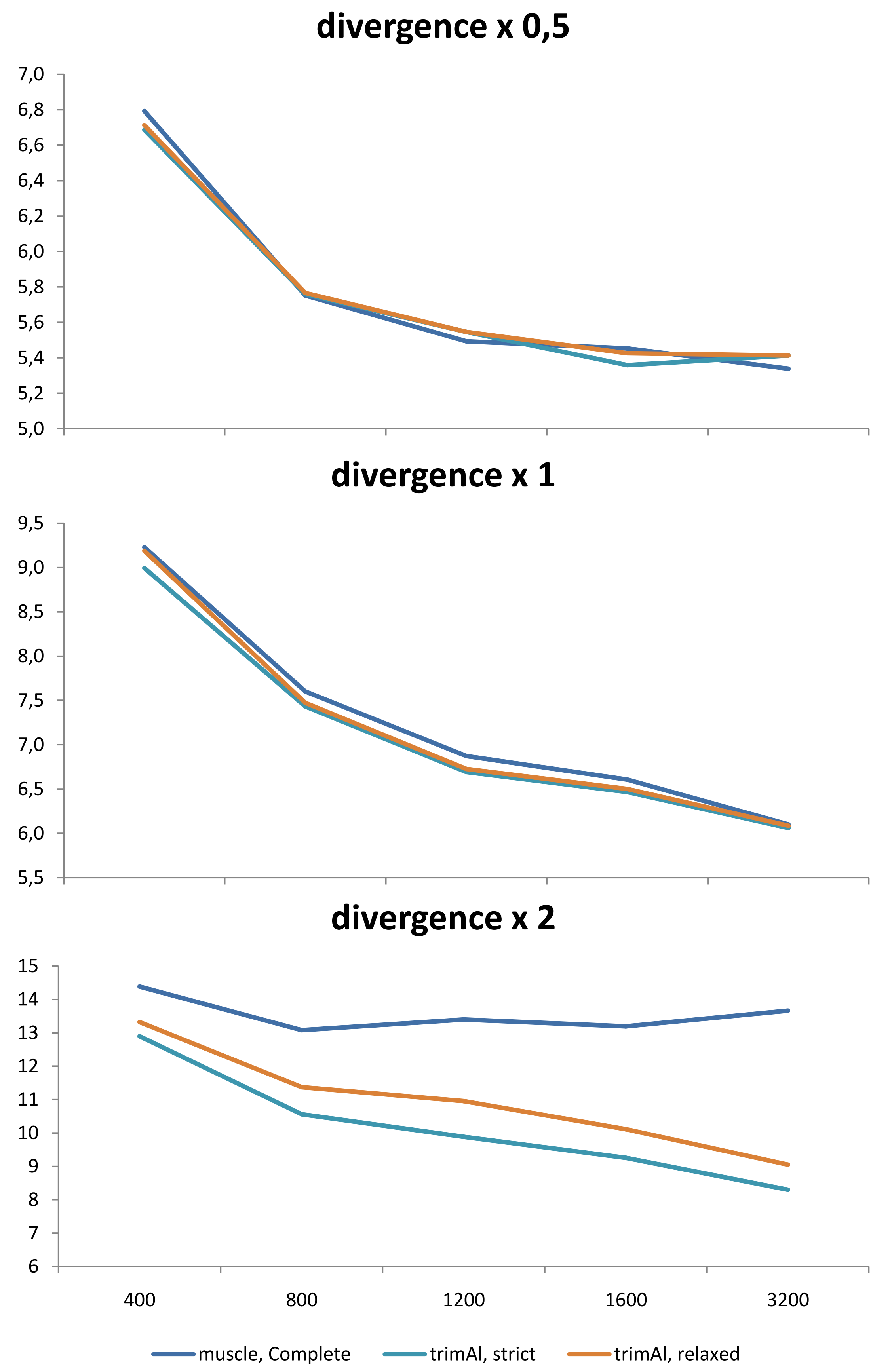
In each panel, x-axis represents the average length of the sequences in the alignment, whereas the y-axis represents the Robinson Foulds distance. This distance measures the topological difference between two given tree, therefore, lower values indicate a better performance of the alignment when reconstructing the tree. Finally, there are six lines that represents the performance of each method: untrimmed alignments (green), Gblocks trimming (blue), trimAl gappyout method (red), trimAl strict one (green), trimAl strictplus one (brown) and trimAl automated1 heuristic (pink).

4. Extended Benchmark: trimAl v1.2. Results based on branch-length.

Just as before, these figures are complementary to those shown in the supplementary material of trimAl v1.2 and show the complete results for all the methods applied.

This set of 4 figures (figures S15 to S18) describes the results in terms of Ktree score, which takes into account differences in branch lengths. These figures are organized as in the previous set. However, the y-axis represents the Ktree scores. This score measures the branch-length differences between two given trees, therefore, lower values indicate a better performance of the alignment when reconstructing trees.

Robinson Foulds Distance - Asymmetric trees. NJ Method. Muscle

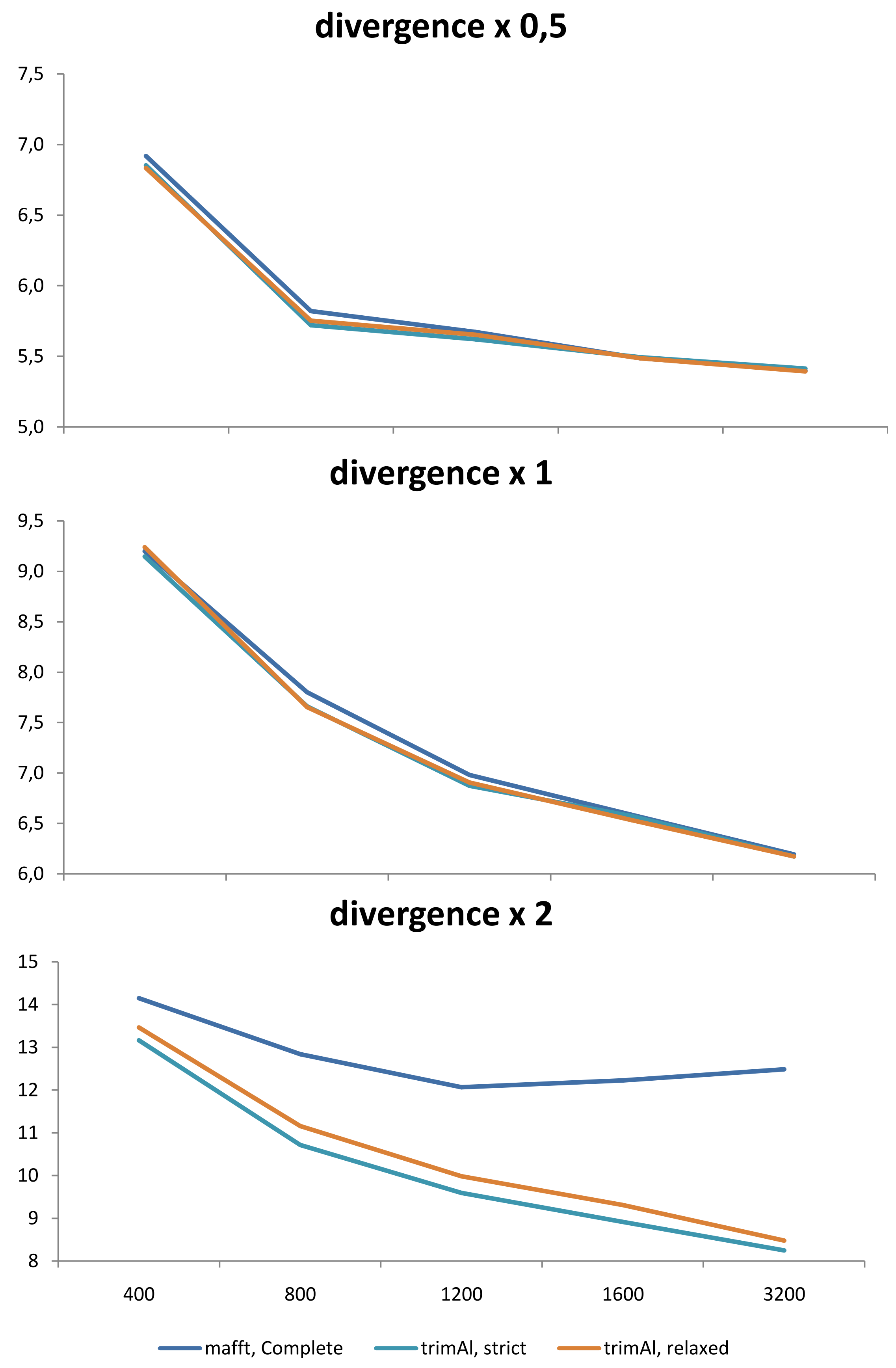


divergence x 0,5

divergence x 1

divergence x 2

Robinson Foulds Distance - Asymmetric trees. NJ Method. Mafft

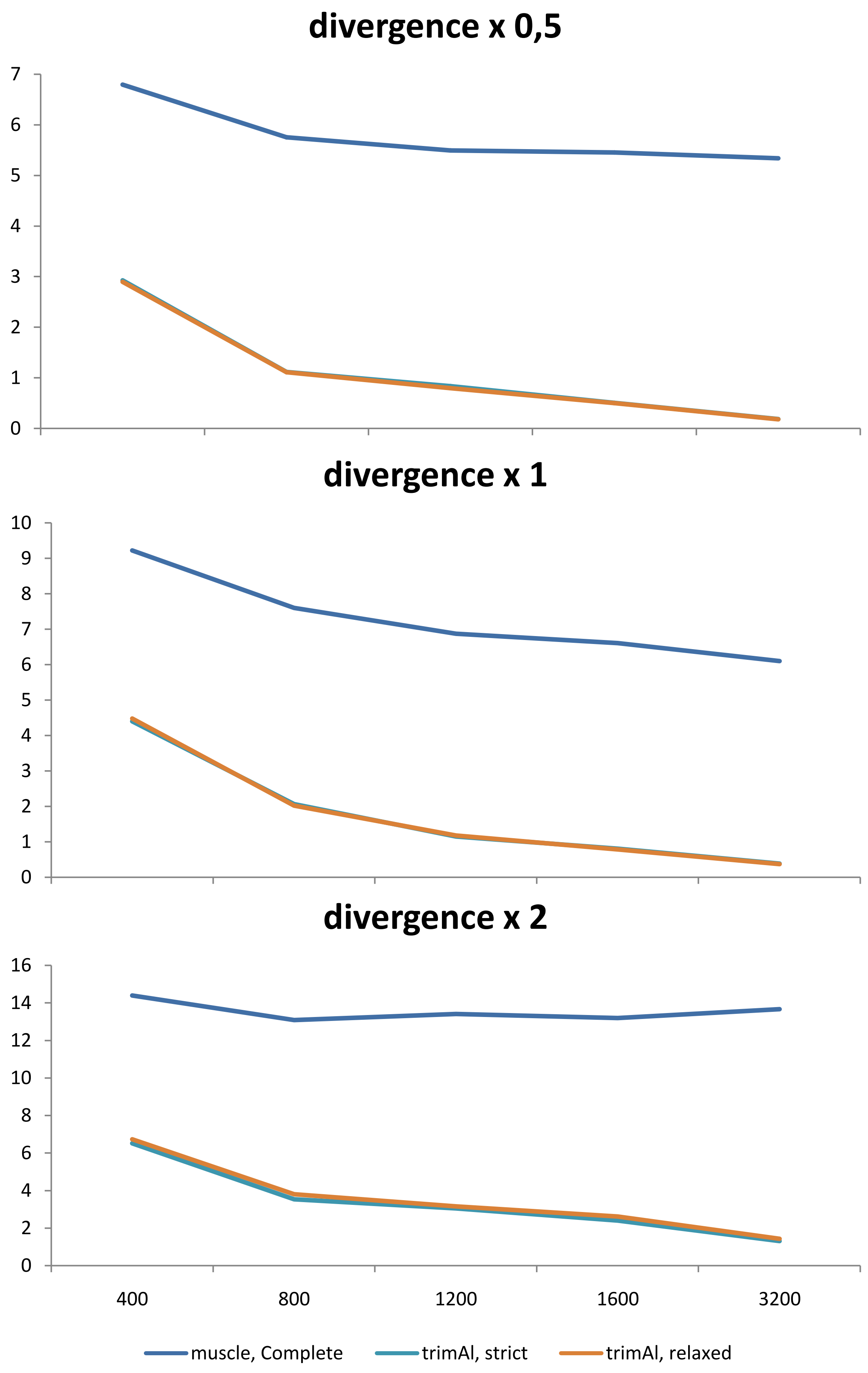


divergence x 0,5

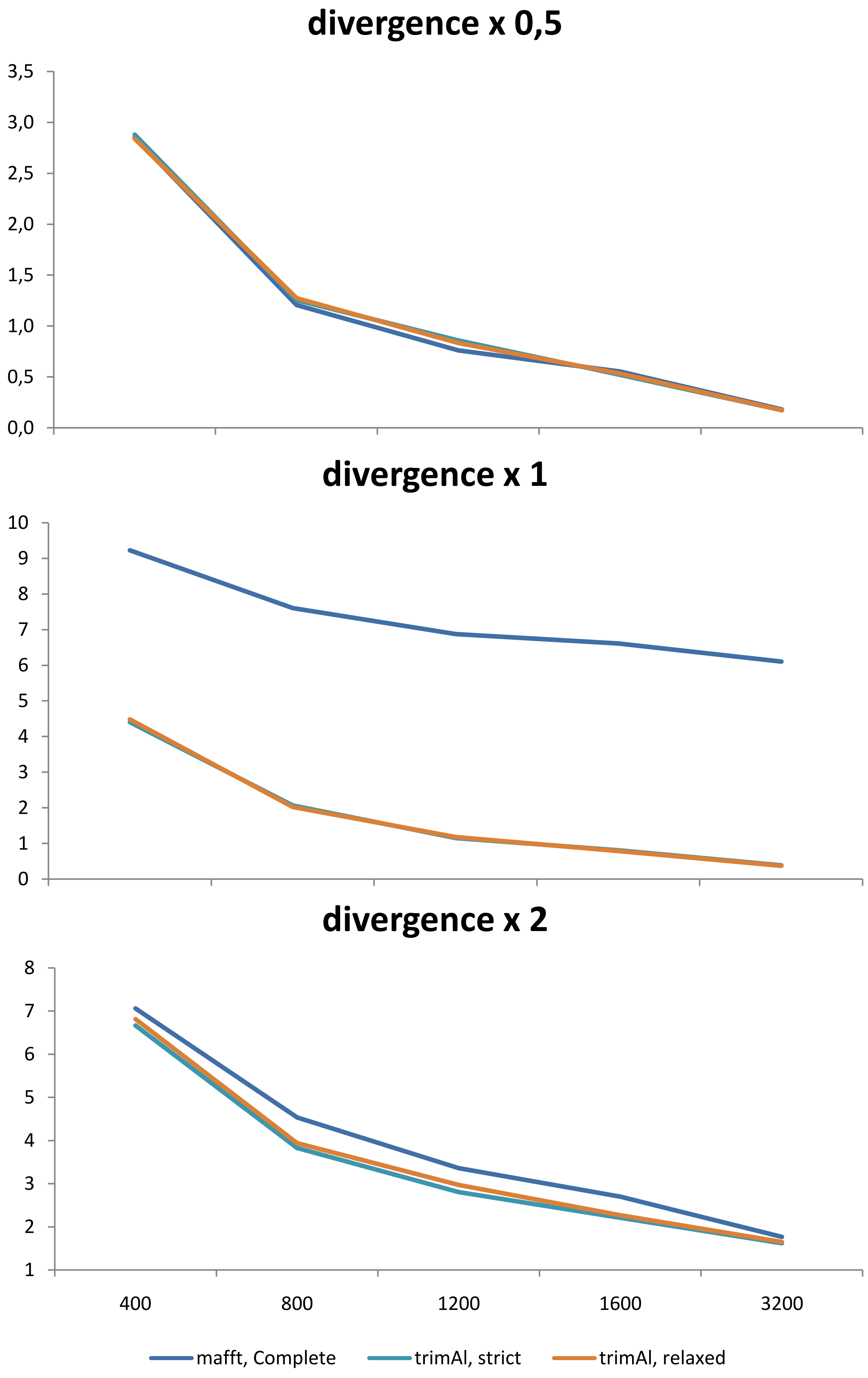
divergence x 1

divergence x 2

Robinson Foulds Distance - Asymmetric trees. ML Method. Muscle



Robinson Foulds Distance - Asymmetric trees. ML Method. Mafft

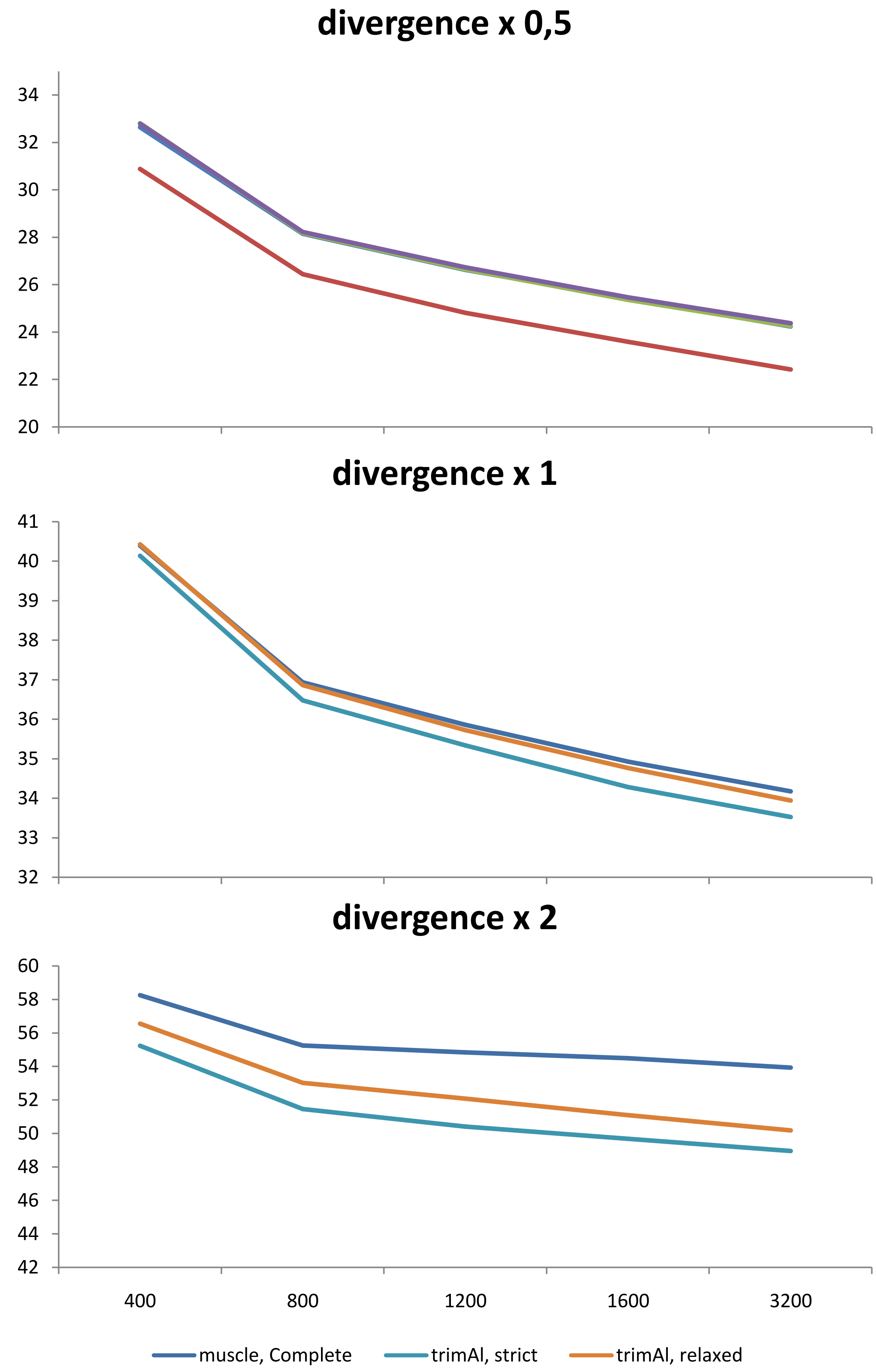


divergence x 0,5

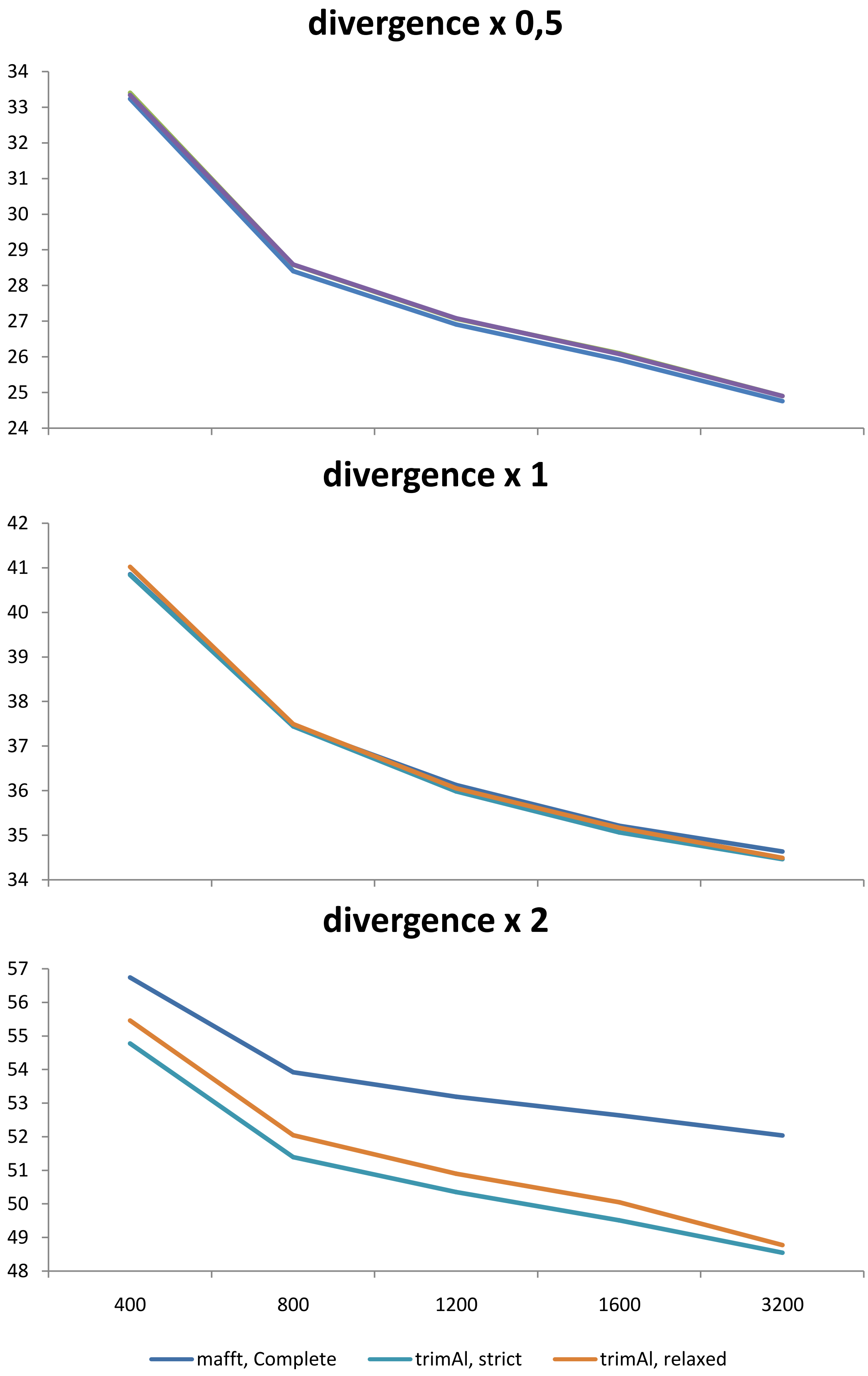
divergence x 1

divergence x 2

Ktreedist - Asymmetric trees. NJ Method. Muscle



Ktreedist - Asymmetric trees. NJ Method. Mafft

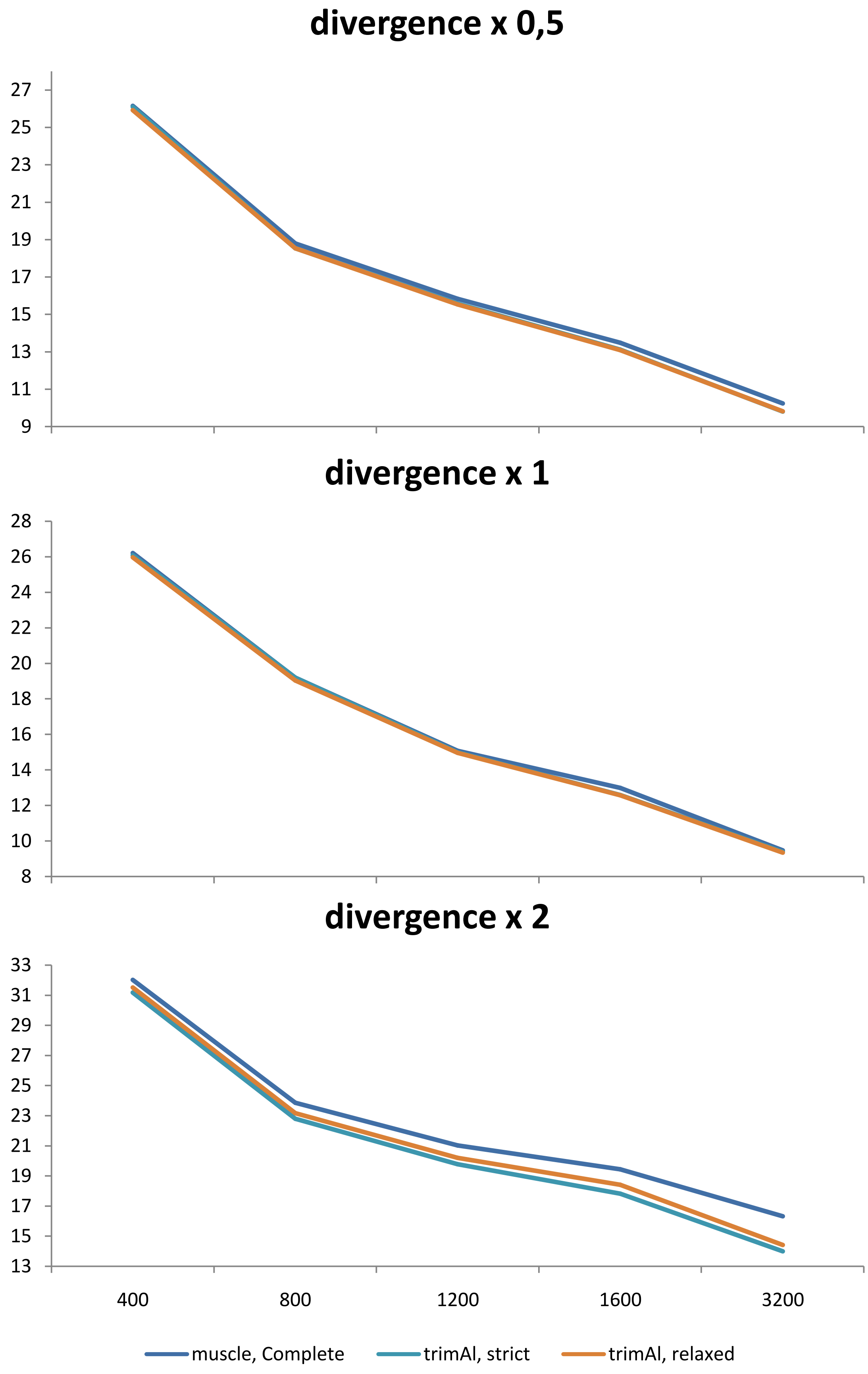


divergence x 0,5

divergence x 1

divergence x 2

Ktreedist - Asymmetric trees. ML Method. Muscle.

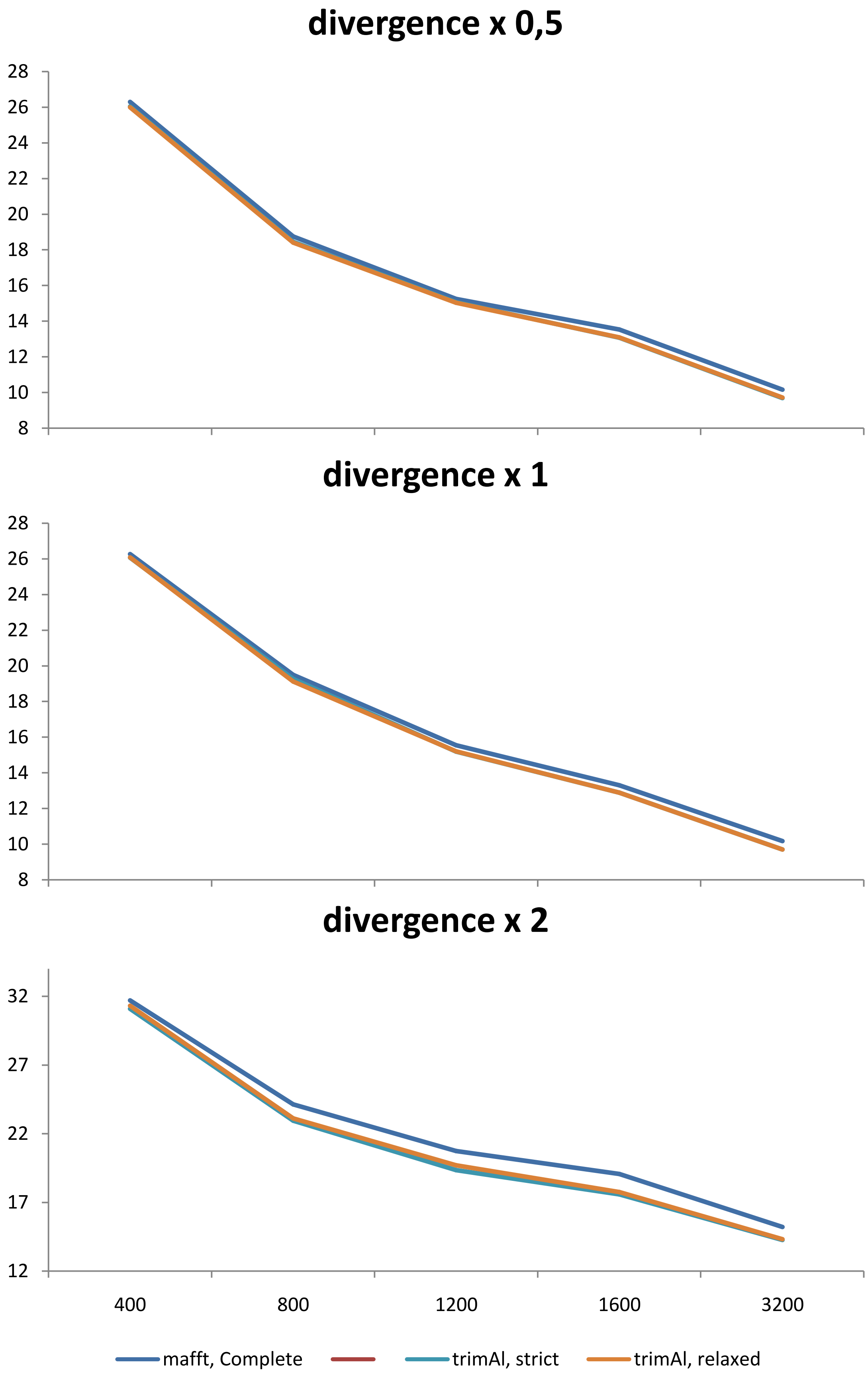


divergence x 0,5

divergence x 1

divergence x 2

Ktreedist - Asymmetric trees. ML Method. Mafft.



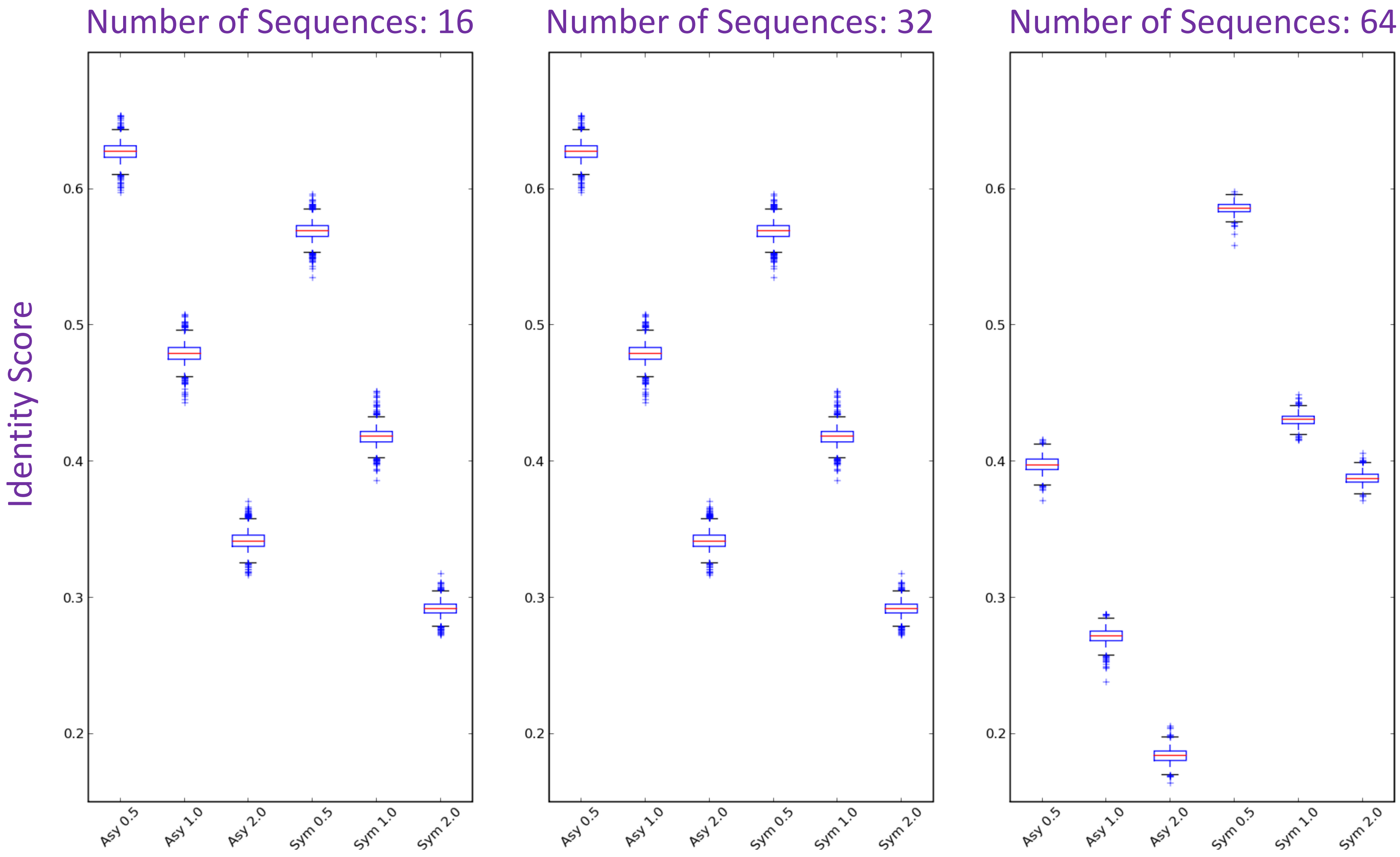
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Identity Score
0.6
0.5
0.4
0.3
0.2
Asy 0.5
Asy 1.0
Asy 2.0
Sym 0.5
Sym 1.0
Sym 2.0

Figure 9

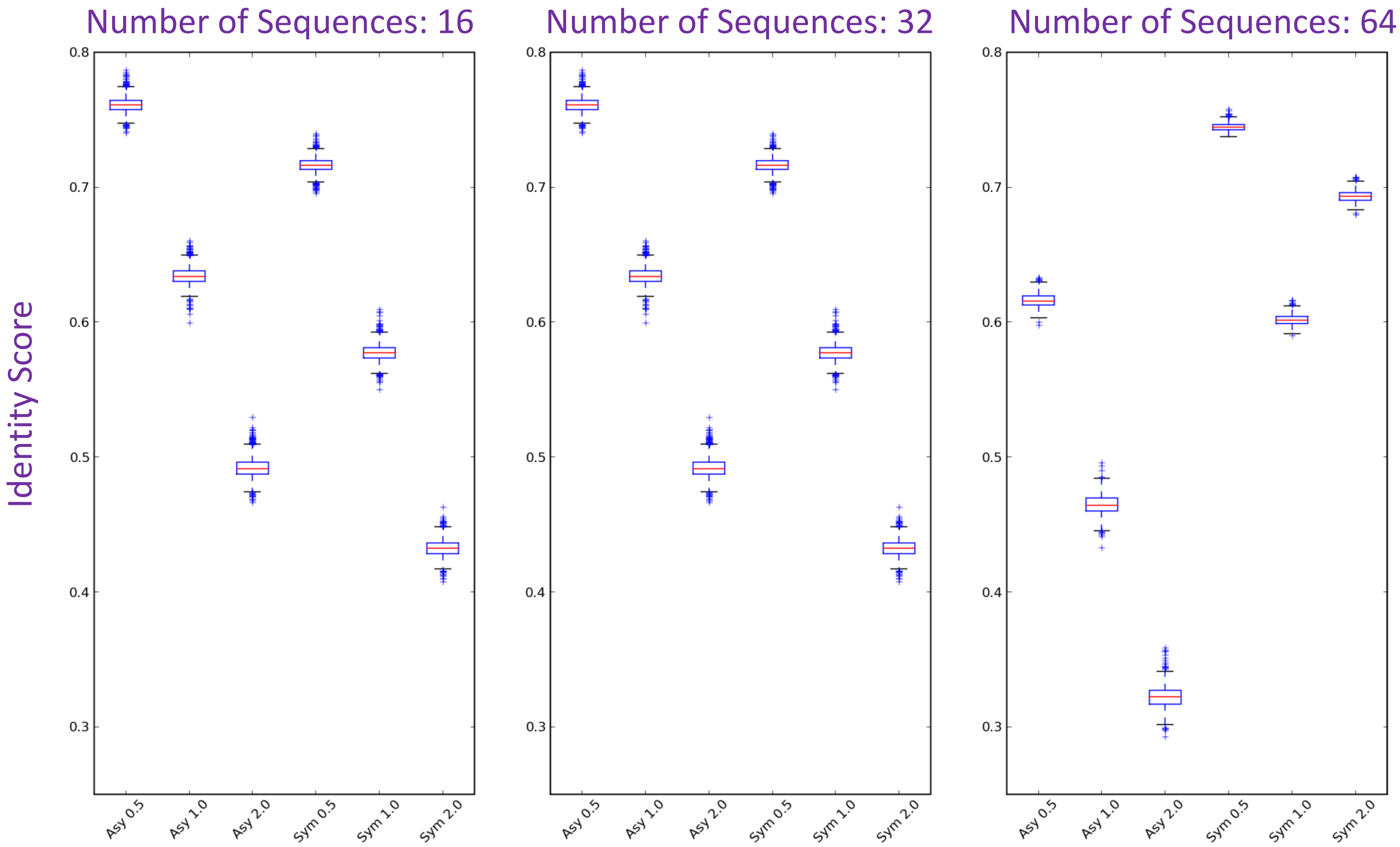

Identity Score
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
0.8
0.7
0.6
0.5
0.4
0.3
Asy 0.5
Asy 1.0
Asy 2.0
Sym 0.5
Sym 1.0
Sym 2.0

Figure 10

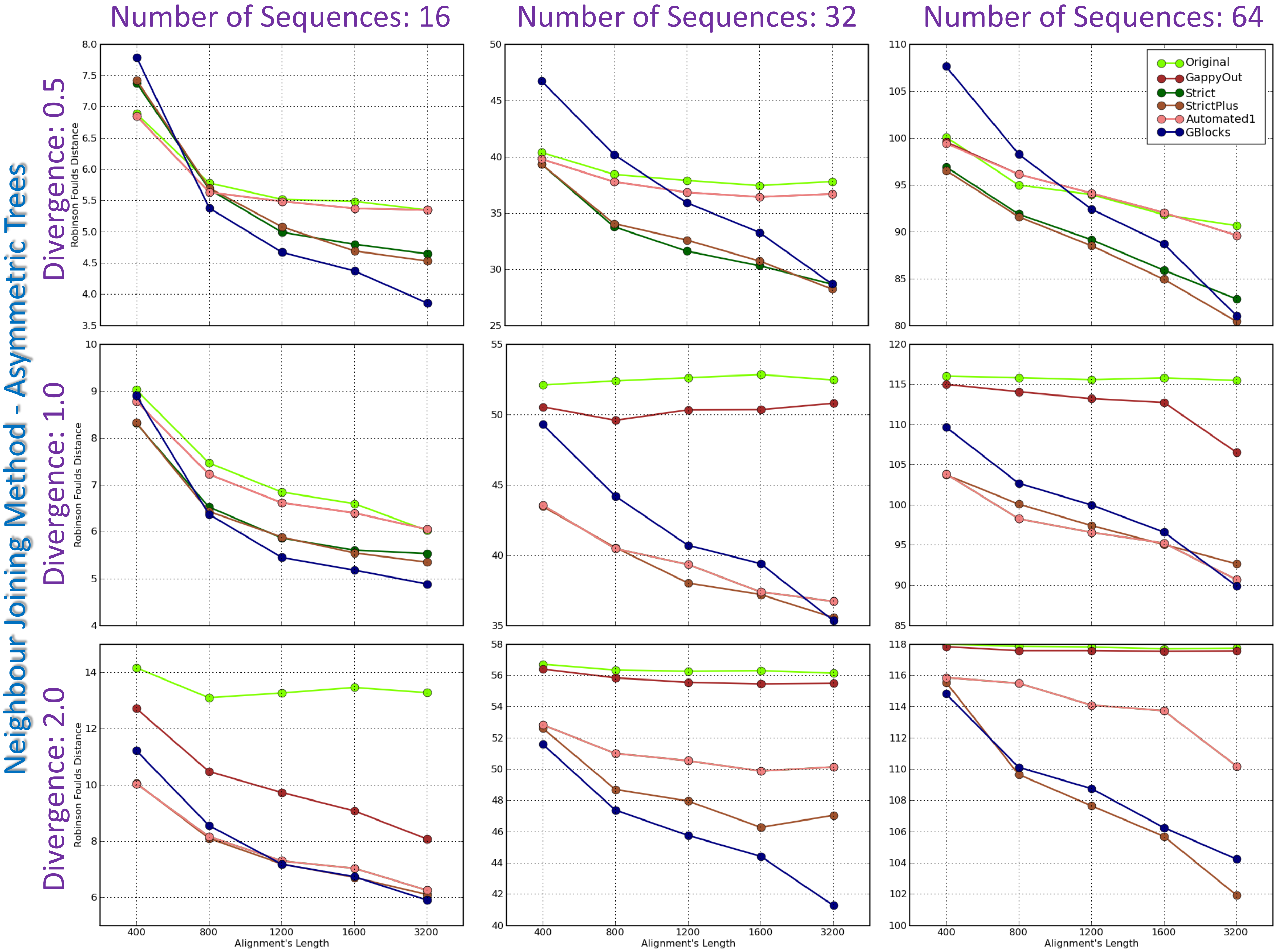

Neighbour Joining Method - Asymmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
Robinson Foulds Distance
Alignment's Length
400
800
1200
1600
3200
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
50
45
40
35
30
25
110
105
100
95
90
85
80
10
9
8
7
6
5
4
55
50
45
40
35
120
115
110
105
100
95
90
85
14
12
10
8
6
58
56
54
52
50
48
46
44
42
40
118
116
114
112
110
108
106
104
102
100
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks

Figure 11

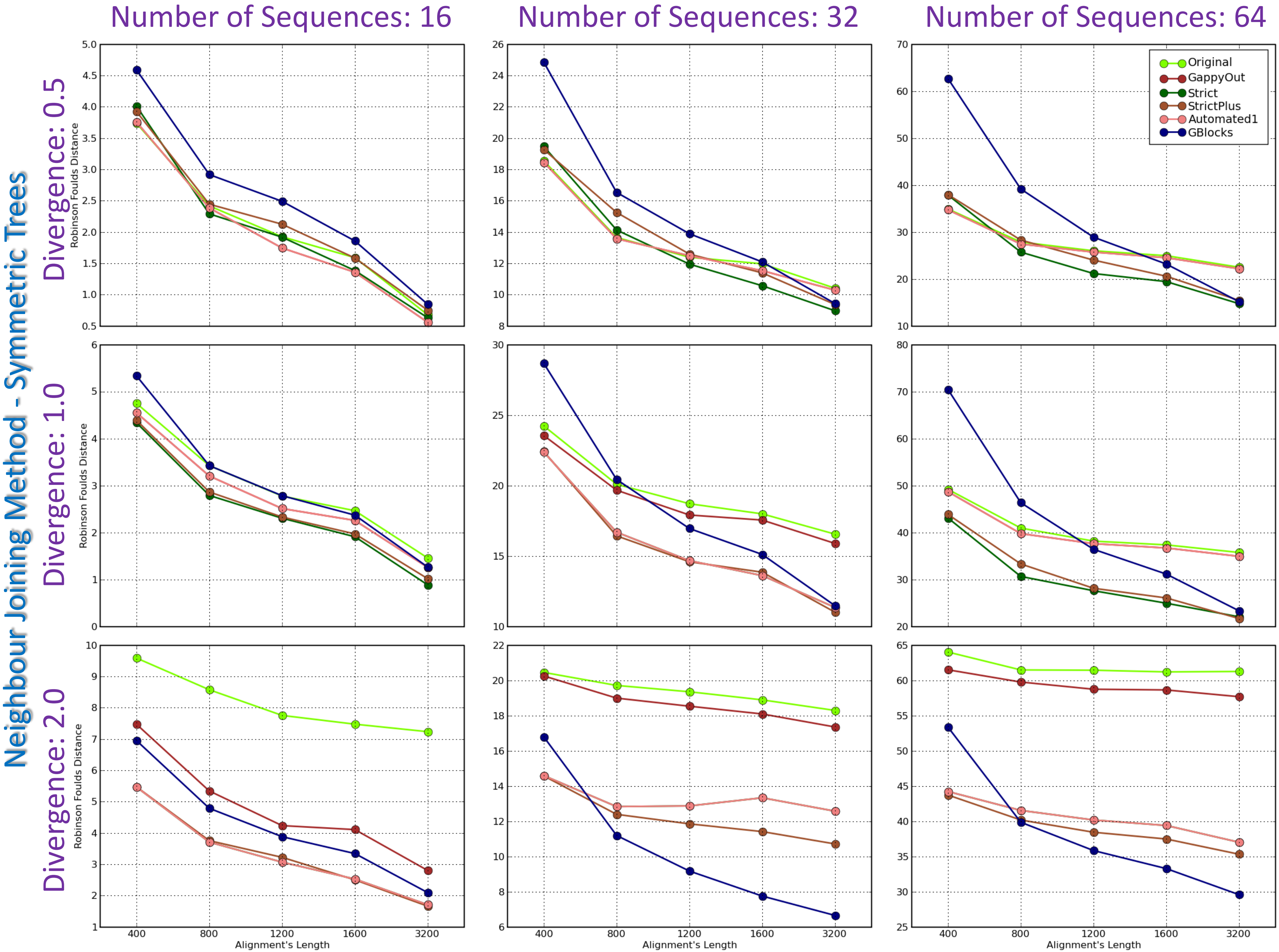

Neighbour Joining Method - Symmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
Robinson Foulds Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks

Figure 12

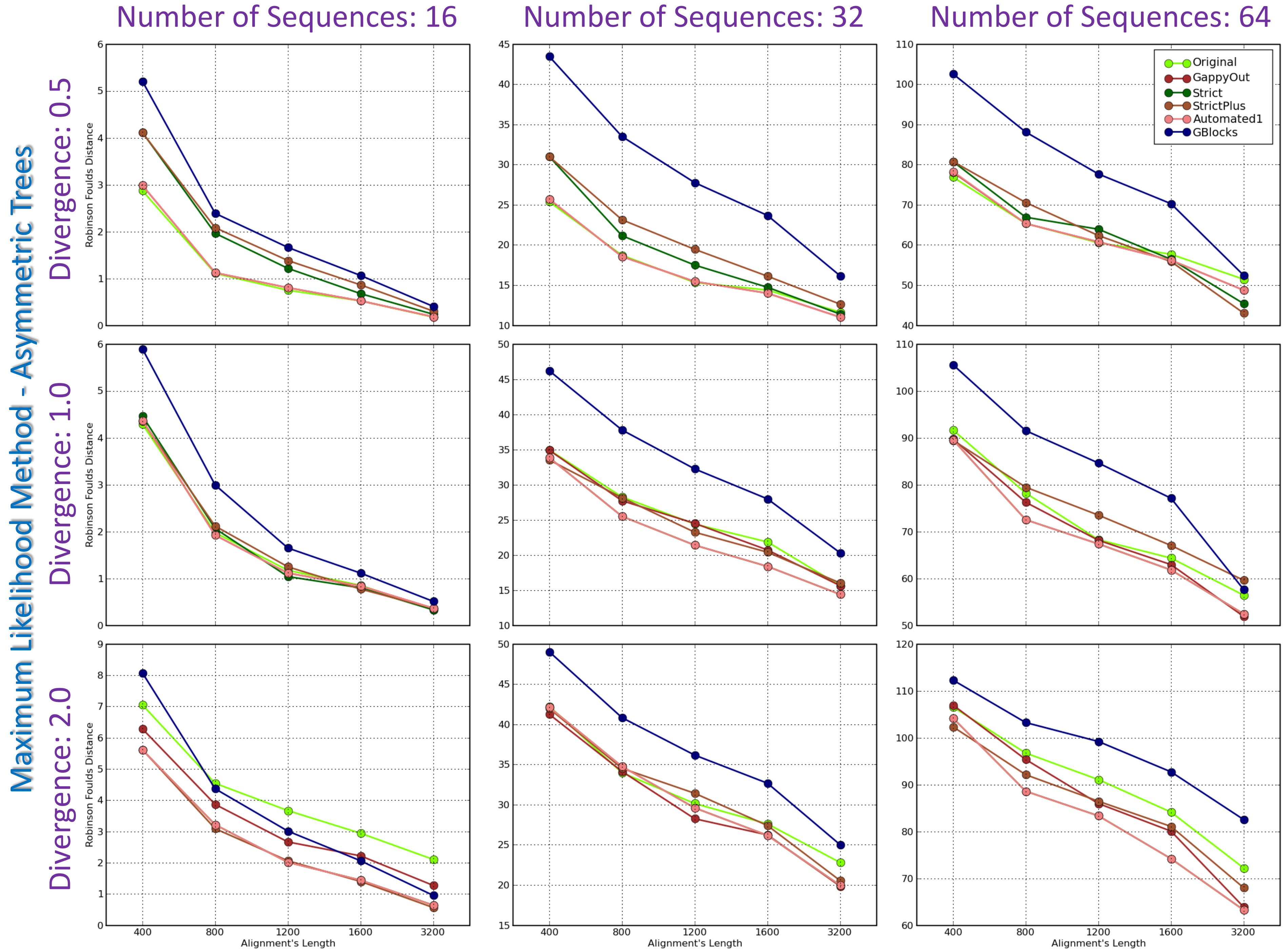

Maximum Likelihood Method - Asymmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
Robinson Foulds Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks
0
1
2
3
4
5
6
7
8
9
10
15
20
25
30
35
40
45
50
60
70
80
90
100
110
120

Figure 13

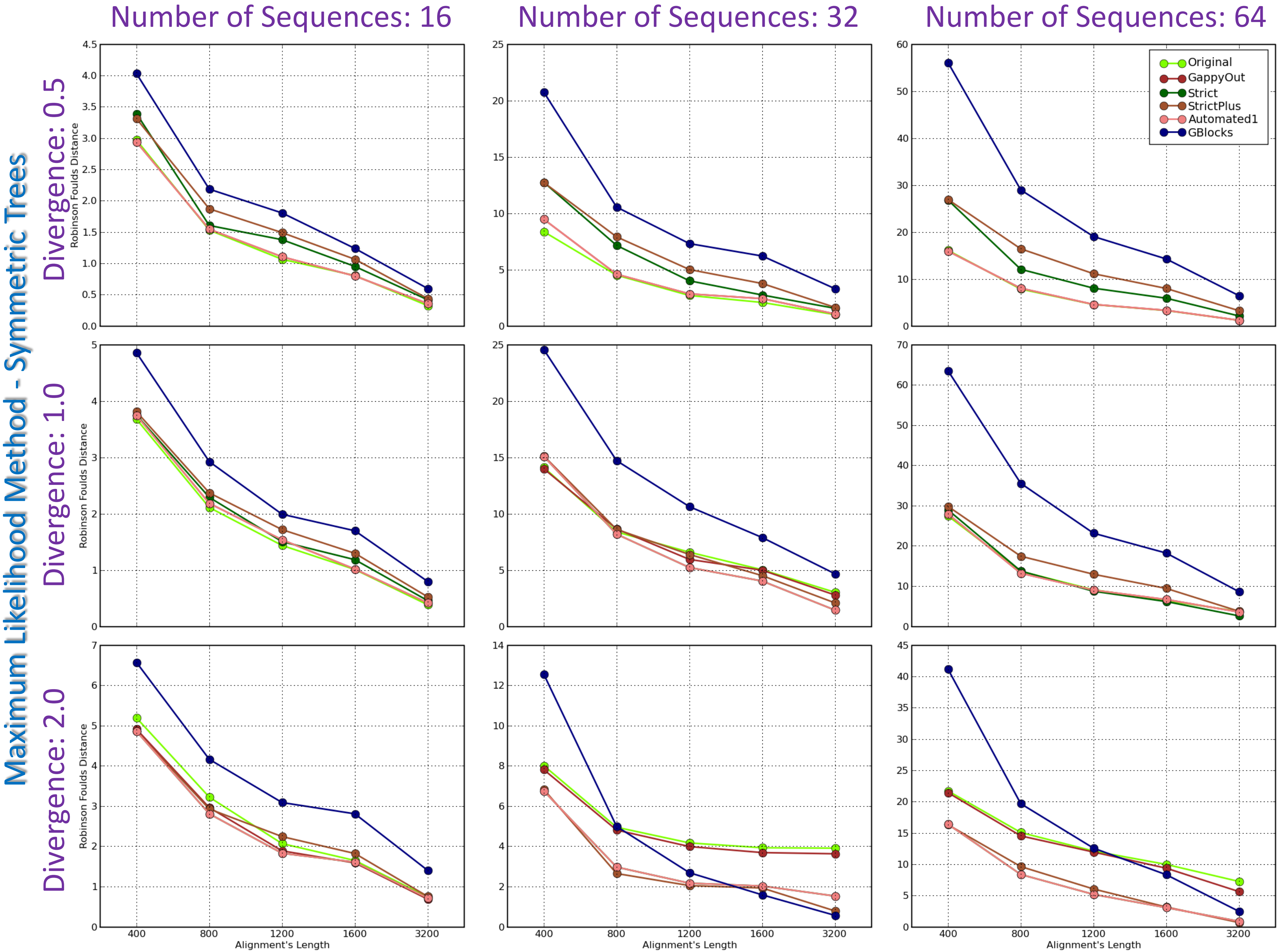

Maximum Likelihood Method - Symmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks
Robinson Foulds Distance
Alignment's Length
400
800
1200
1600
3200

Figure 14

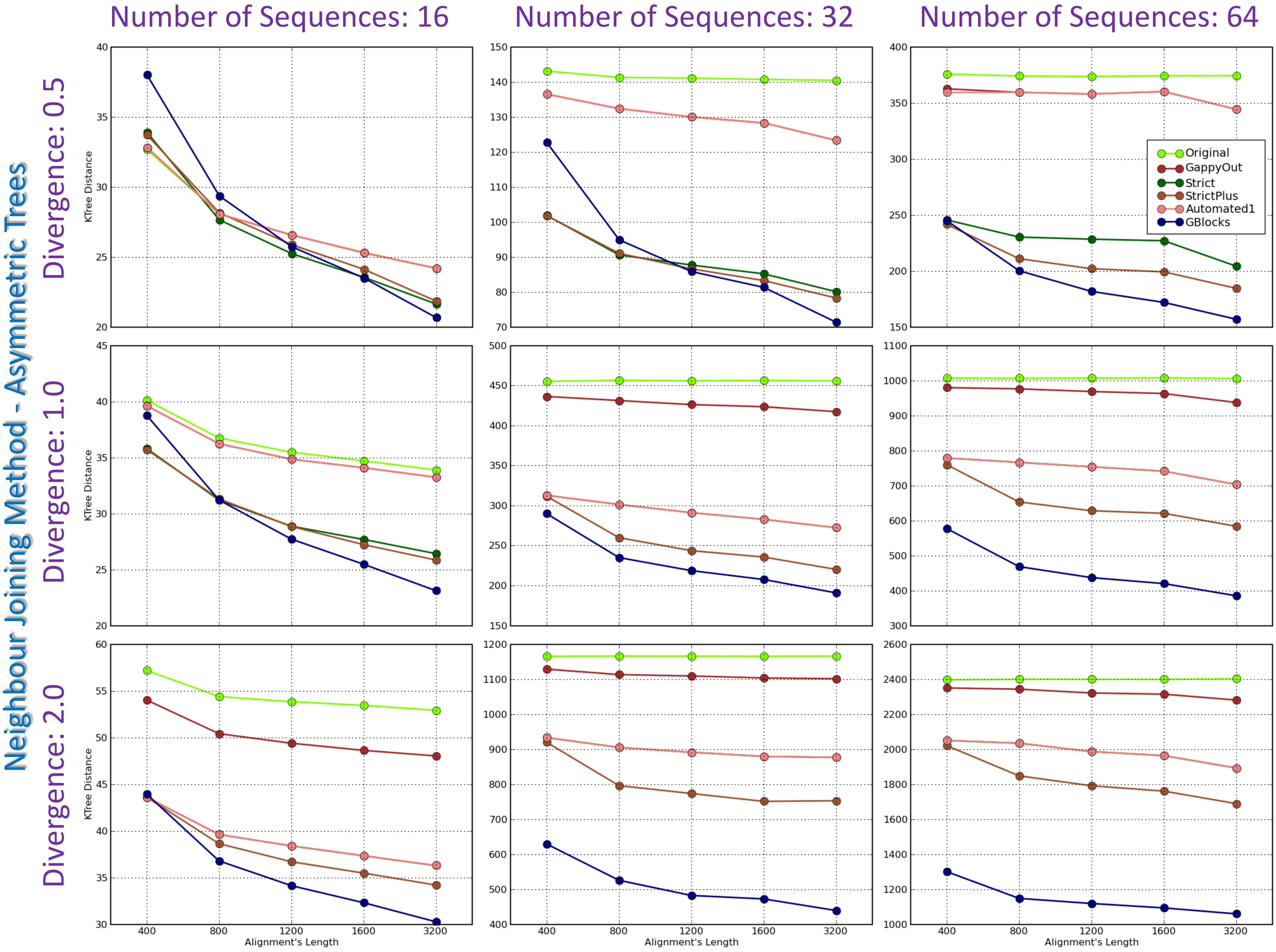

Neighbour Joining Method - Asymmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
KTree Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks
20
25
30
35
40
45
50
55
60
70
80
90
100
110
120
130
140
150
200
250
300
350
400
450
500
600
700
800
900
1000
1100
1200
1400
1600
1800
2000
2200
2400
2600

Figure 15

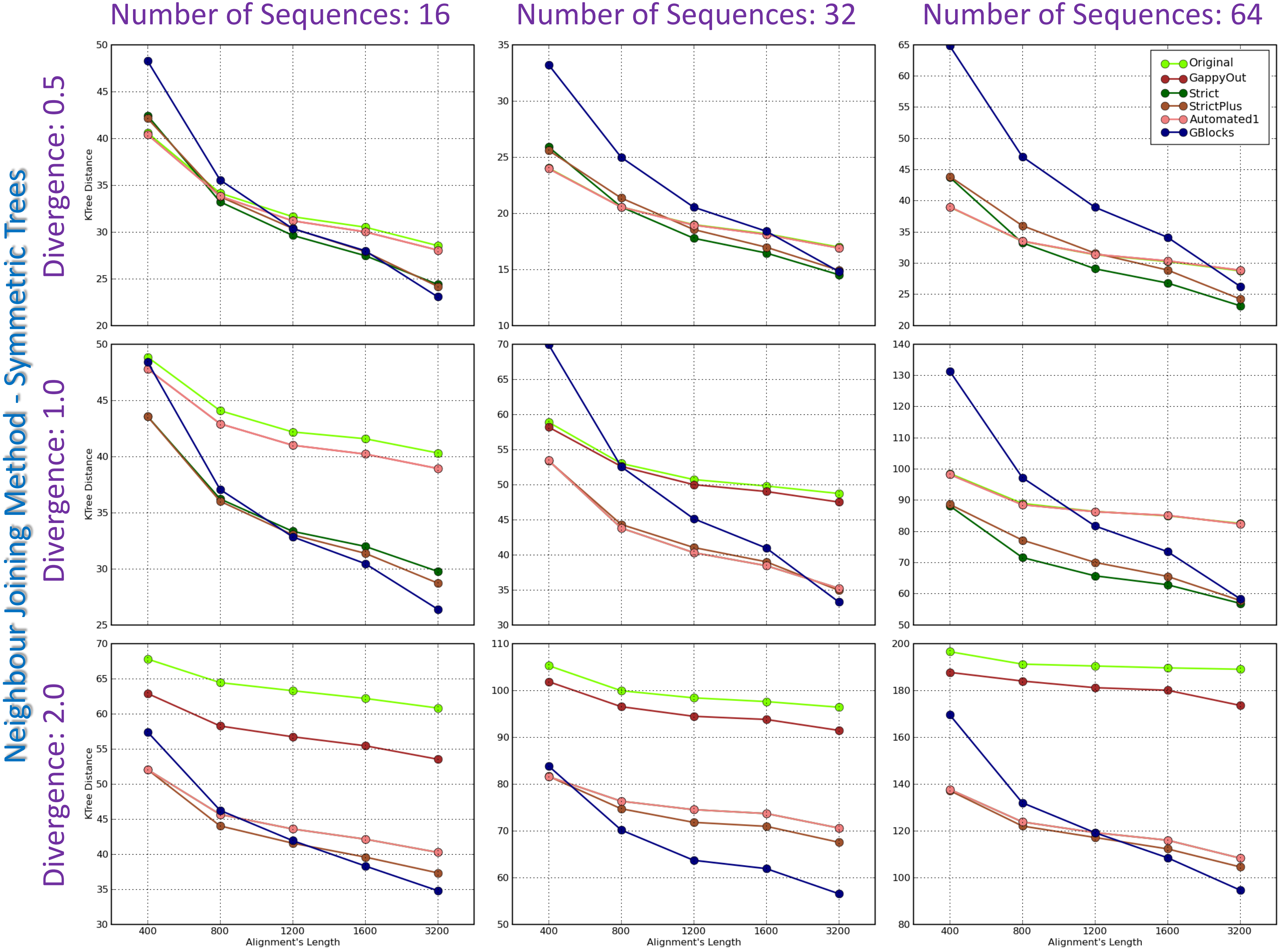

Neighbour Joining Method - Symmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
KTree Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks
20
25
30
35
40
45
50
10
15
55
60
65
70
80
90
100
110
120
130
140
160
180
200

Figure 16

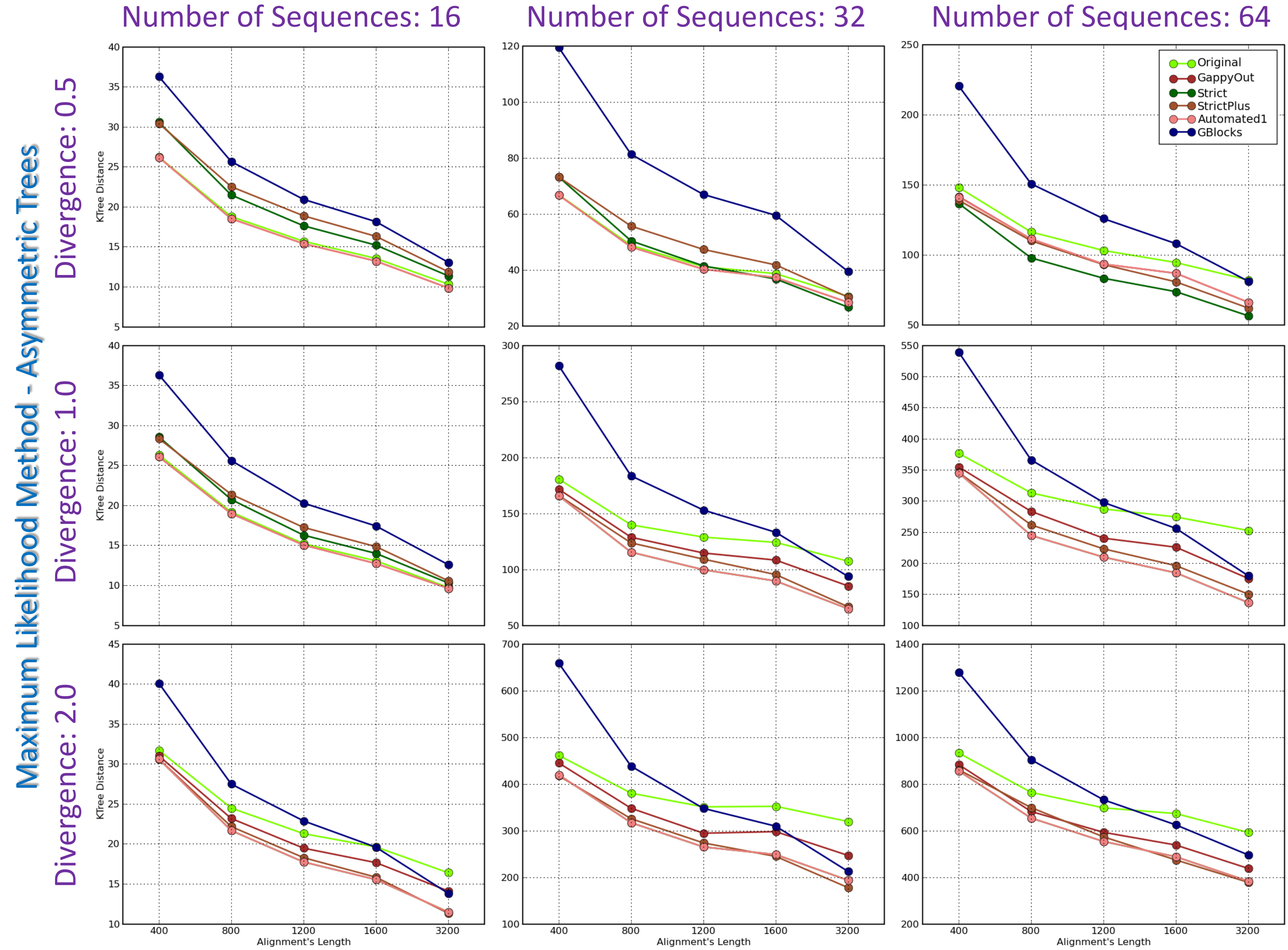

Maximum Likelihood Method - Asymmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
KTree Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks

Figure 17

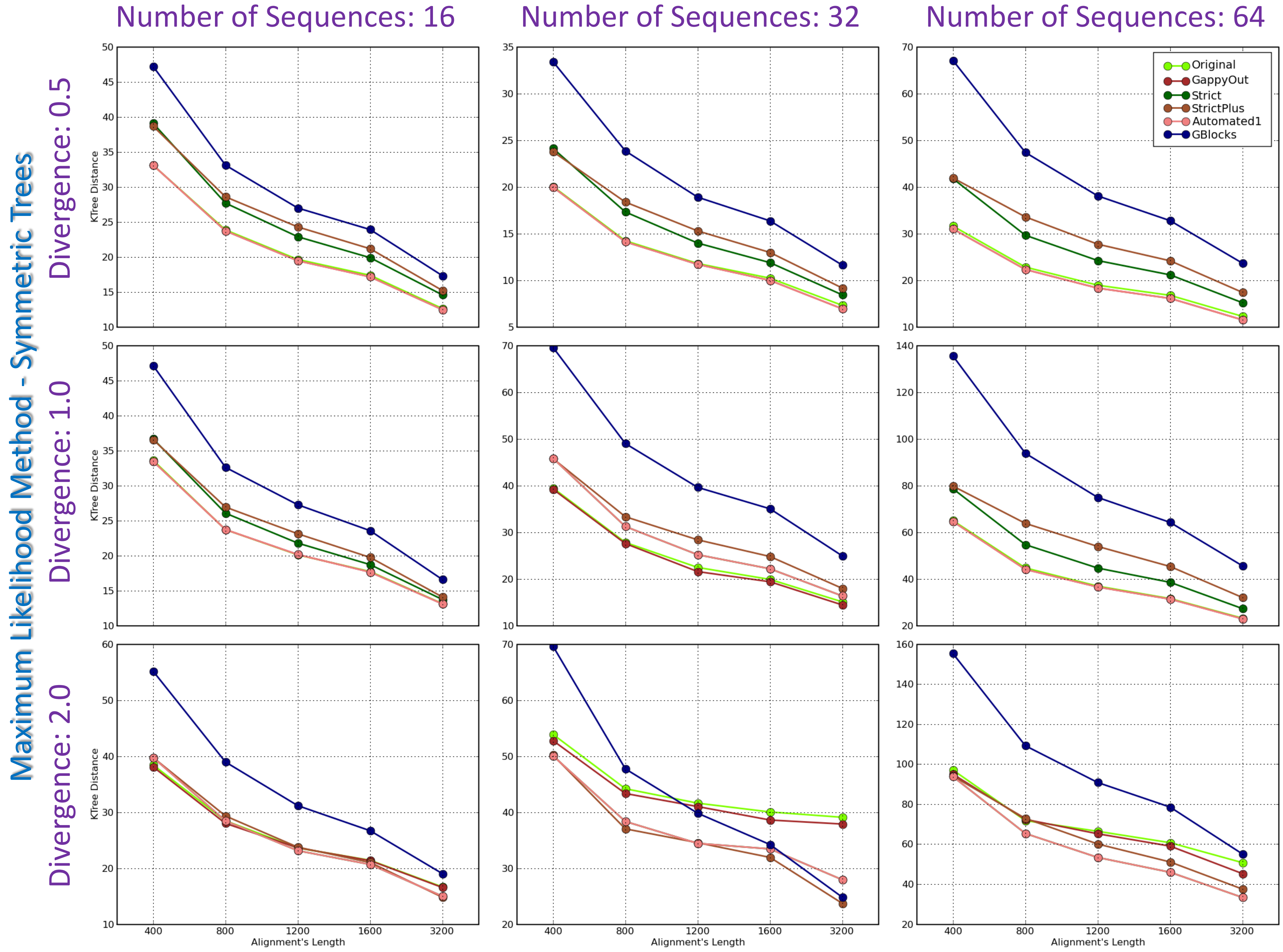

Maximum Likelihood Method - Symmetric Trees
Number of Sequences: 16
Number of Sequences: 32
Number of Sequences: 64
Divergence: 0.5
Divergence: 1.0
Divergence: 2.0
KTree Distance
Alignment's Length
400
800
1200
1600
3200
Original
GappyOut
Strict
StrictPlus
Automated1
GBlocks

Figure 18