Supplementary Materials for "Multi-objective formulation of MSA for phylogeny estimation

(Do phylogeny-aware measures guide towards better phylogenetic tree?)"

Muhammad Ali Nayeem, Md. Shamsuzzoha Bayzid, Atif Hasan Rahman, Rifat Shahriyar, and M. Sohel Rahman*

Department of CSE, BUET, Dhaka 1205, Bangladesh

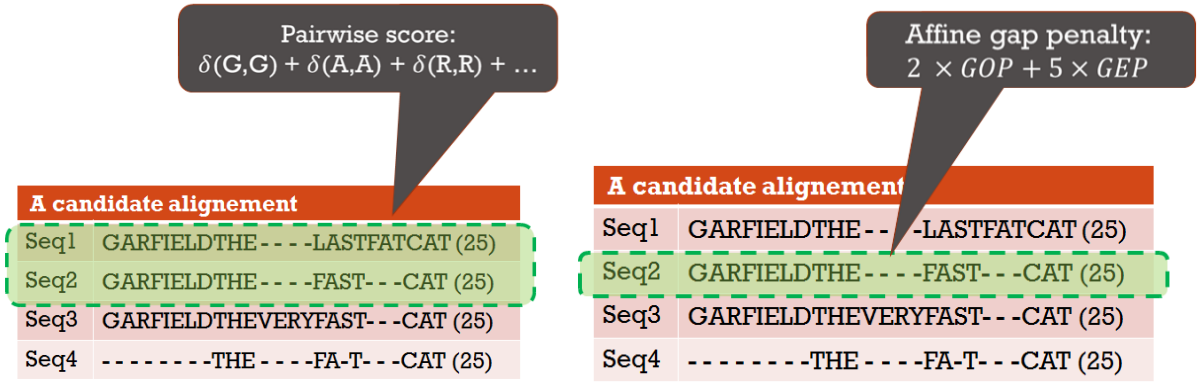
Contents

*Corresponding author. E-mail: msrahman@cse.buet.ac.bd

S1 Objective functions for MSA

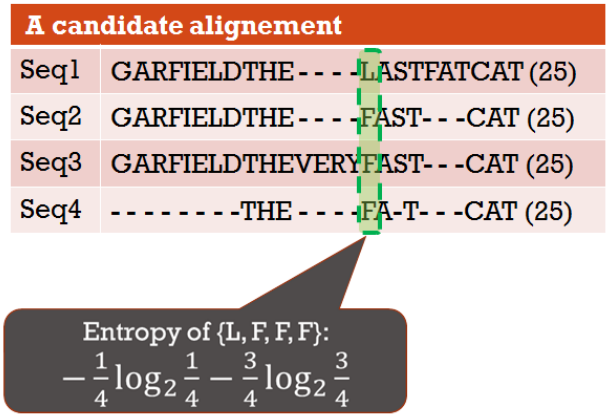
There are numerous objective functions defined for MSA in the literature. We identify the following widely used objective functions from the recent works and briefly discuss their feasibility in MSA:

- **Maximize sum of pairs score** [1, 2]: This is an extension of pairwise sequence alignment score. Pairwise score is calculated for each pair of aligned sequences. Then, we calculate the total score by summing pairwise scores of all possible pairs. In Figure S1a, the pairwise score is calculated by considering the elements of the same columns of two aligned sequences with the scoring or substitution matrix δ. There are some standard substitution matrices for biological sequences at `ftp://ftp.ncbi.nih.gov/blast/matrices/`.



(a) Pairwise score of two aligned sequences.

(b) Affine gap penalty of an aligned sequence.



(c) Entropy of a column of alignment.

Figure S1: Three objective functions for MSA. The example alignment is taken from [3].

- **Minimize entropy** [4]: Entropy is a measurement of dissimilarity in the same columns of different aligned sequences. When all the columns contain same element, the entropy is minimum 0. Again, the entropy is maximum 1 when every element is different. Total entropy is calculated by summing up entropy values of all columns. Figure S1c demonstrates the calculation of entropy for a single column. The problem with this function is that, while calculating entropy researchers treat gap as a separate character without proper justification.

Table S1: Alphabetic list of acronyms used in this study.

Acronym	Usage
23S.E	A biological rRNA dataset
23S.E.aa_ag	A biological rRNA dataset
BBXY0MN	MN^{th} BAliBASE instance under RVXY group
Clustal Ω	A state-of-the-art MSA method
Clustal W	A state-of-the-art MSA method
FN rate	False negative rate, measures quality of a phylogenetic tree w.r.t. the reference tree
FSA	A state-of-the-art MSA method
Gap	No. of gaps, an objective function that measures goodness of an MSA
GapCon	Concentration of gaps, an objective function that measures goodness of an MSA
Kalign	A state-of-the-art MSA method
MAFFT	A state-of-the-art MSA method
ML	Maximum likelihood approach for inferring a phylogenetic tree from an MSA
MSA	Multiple sequence alignment
MUSCLE	A state-of-the-art MSA method
NSGA-II	A multi-objective metaheuristics
NSGA-III	A multi-objective metaheuristics which an improved version of NSGA-II to handle more than three objective functions
PASTA	A state-of-the-art MSA method
PRANK	A state-of-the-art MSA method
ProbCons	A state-of-the-art MSA method
R0	A random replicate of 100-taxon simulated dataset
R14	A random replicate of 100-taxon simulated dataset
R19	A random replicate of 100-taxon simulated dataset
R4	A random replicate of 100-taxon simulated dataset
R9	A random replicate of 100-taxon simulated dataset
RetAlign	A state-of-the-art MSA method
RV11	One of the six groups of BAliBASE 3.0 benchmark
RV12	One of the six groups of BAliBASE 3.0 benchmark
RV20	One of the six groups of BAliBASE 3.0 benchmark
RV30	One of the six groups of BAliBASE 3.0 benchmark
RV40	One of the six groups of BAliBASE 3.0 benchmark
RV50	One of the six groups of BAliBASE 3.0 benchmark
SimG	Similarity based on gap columns, an objective function that measures goodness of an MSA
SimNG	similarity based on non-gap columns, an objective function that measures goodness of an MSA
SOP	Sum of pairs, an objective function that measures goodness of an MSA without using the reference alignment
SP score	Sum-of-pair score, measures quality of an MSA w.r.t. the reference alignment
T-Coffee	A state-of-the-art MSA methods
TC	No. of totally aligned columns, an objective function that measures goodness of an MSA without using the reference alignment
TC score	Total-column score, measures quality of an MSA w.r.t. the reference alignment
wSOP	Weighted sum of pairs, an objective function that measures goodness of an MSA

- **Minimize affine gap penalty** [1, 5, 6, 7]: Affine gap penalty assigns different penalty for opening a gap ($GapOpeningPenalty, GOP$) and extending a gap ($GapExtensionPenaly, GEP$) while computing gap penalty for a particular sequence. Then finally, summation of gap penalties of all sequences is to be minimized. An example is demonstrated in Figure S1b showing the calculation of gap penalty of one sequence. Here two ideas (i.e. percentage of gap and concentration of gap) are combined without explanation. Also researchers face trouble to fix the value of two penalties.

- **Maximize weighted sum of pairs score with affine gap penalties** [3, 8]: Here two objectives are combined in the form: weighted sum of pairs score - affine gap penalty. To calculate weighted sum of pairs score, score of each pair of characters are multiplied by the sequence weight between that the corresponding two sequences. This weight is computed using the Levenshtein distance between two non-aligned sequences. Levenshtein distance is the is the minimum number of insertions, deletions or substitutions needed to convert one sequence into the other.

- **Maximize number of totally aligned columns** [9, 2, 3, 8, 9, 10]: Maximizing the number of totally aligned columns is the most simple used objective. But for input data comprising large number of taxa, its value is confined to a few values.

- **Minimize percentage of gaps** [11, 9, 10]: High percentage of gaps means the sequences had to be significantly modified to align with each other. This is used as a minimizing objective function to find a better candidate solution. It can be also considered as percentage of non-gaps.

- **Maximize similarity** [5, 7]: For each column of MSA, similarity considers the ratio of the dominant character. This ratio is averaged over all columns. The closer the value of similarity is to one, the larger the probability that the candidate alignment will be discovered as the best possible alignment. Here we find similar problem as with entropy. Researchers discard gap while calculating ratio of characters in a column without sound reasoning.

S2 Multi-objective metaheuristics

While optimizing multiple objective functions simultaneously, a multi-objective metaheuristics determines a set of solutions (instead of a single solution) which represents the best-possible compromise of all objectives. A solution is said to dominate another one if and only if it is equal to that solution in all objectives and also better than that in at least one objective. A solution is said to be Pareto optimal if no other solutions can dominate it. The set of all Pareto optimal solutions is called Pareto set and the image of pareto set in the objective space is known as Pareto front. However, practically a multi-objective metaheuristics aims to approximate the Pareto front as precisely as possible with a finite number of solutions.

Among metaheuristics, multi-objective evolutionary algorithms (MOEAs) are well-suited to solve multi-objective optimization problems [12]. MOEAs deal with a set of possible solutions (known as population) at once which allows finding several members of the Pareto front in a single run of the algorithm. Moreover, they are black-box optimization methods which do not need particular assumptions like continuity or differentiability of the decision space.

A general structure of MOEAs is summarized in Algorithm S1. Here the *Crossover* and *Mutation* are popularly known as genetic operators. They generate offspring (new solutions) from parents (existing solutions). These are problem-specific and designed based on the actual problem

Algorithm S1 A General structure of MOEA

1: Randomly generate the initial population P_0
2: Evaluate the objective functions of each individual in P_0
3: $t \leftarrow 0$
4: **while** $t <$ maximum value of t **do**
5: Generate offspring population Q_t by applying *Crossover* and *Mutation* on P_t
6: Evaluate the objective functions of each individual in Q_t
7: Produce generation P_{t+1} from P_t and Q_t using *Ranking scheme*
8: $t \leftarrow t + 1$
9: **end while**

to be solved. *Ranking scheme* is used to choose appropriate solutions to form the next generation. This is problem-independent concept and provided by the developers of a specific algorithm. In this study, we considered the two widely used MOEAs for multi-objective optimization. We briefly describe them as follows.

(a) NSGA-II [13] follows the classical structure of a generational genetic algorithm. At first, it applies the typical genetic operators (selection, crossover, and mutation) on the current population to fill an auxiliary population. Then it builds the next-generation by incorporating the best individuals from both the current and auxiliary populations according to a Pareto ranking and the crowding distance operator. Perhaps it is the most commonly used algorithm for solving optimization problems having two or three objective functions.

(b) NSGA-III [14] is designed to handle a large number of objective functions. The skeleton of NSGA-III remains similar to its predecessor NSGA-II with notable changes in its selection mechanism. At each generation, it produces an offspring population from the current population by applying genetic operators. These two populations are merged to form a new population using the selection mechanism. NSGA-III continues to use Pareto dominance as the primary selection criterion to promote convergence. But it substitutes the crowding distance operator in NSGA-II with a clustering operator aided by a set of well-distributed reference points as the secondary selection criterion to maintain diversity. NSGA-III has been shown to perform reasonably.

We implemented the above two metaheuristics using jMetalMSA [15] which is a Java metaheuristic framework for MSA. The important parameters with corresponding values are listed in Table S2. We used the same mutation and crossover operator used by [9]. We provide a short description of these operators below.

- The mutation operator is termed as closed gap shifting, where consecutive gaps are randomly chosen and shifted to another random position in a sequence. This shifting may result columns having only gaps which are then removed. Thus this mutation tries to reduce the number of gaps in the MSA.

- The crossover operator is the single-point crossover over alignments proposed by 2. The operator randomly selects a column from one parent to split it into two blocks (let us refer to them P1a and P1b). The same selected positions are located in the other parent (which are not necessarily in the same column) and is tailored so that the right piece can be joined to the left piece of the first parent and vice versa (P2a and P2b). Finally, the selected blocks

are exchanged between these two parents to create two new individuals with the combination of the blocks: [P1a + P2b] and [P1a + P1b]. After that, any empty space that appears at the junction point is filled with gaps.

Table S2: Major parameters of our algorithms.

Algo.	Parameter	Value
All	Max. generations	500
	Mutation	Closed gap shifting
	Mutation rate	0.2
	Crossover	Single-point crossover
	Crossover rate	0.8
NSGA-II	Population size	100
	No. of runs	20
NSGA-III	No. of reference points	120
	Population size	120
	No. of runs	25

S3 Evaluation of estimated alignments

We evaluate estimated alignments with respect to reference alignment using two well-known alignment quality scores called TC score and SP score. These two scores are defined below:

- TC score is the ratio of the number of correctly aligned columns in the estimated alignment to the total number of aligned columns in the reference alignment. This is also known as column score.
- SP score is the ratio of the number of aligned pairs in the estimated alignment to the total number of aligned pairs in the reference alignment.

For both the measures, higher value implies better score.

S4 Phylogenetic tree estimation

For each of the generated alignment we estimate the phylogenetic tree using Maximum Likelihood (ML) method which is the standard way of estimating phylogenetic tree from sequence data [16]. FastTree[17] and RAxML [18] are the most widely used software for this purpose. FastTree can produce output very quickly with little (and in some cases no) degradation in tree accuracy, as compared to RAxML [16]. In this study we had to estimate a large number of phylogenetic trees. So we choose FastTree over RAxML.

S5 Evaluation of phylogenetic tree

We evaluate the quality of each estimated ML tree with respect to the true phylogenetic tree using a widely used measure known as the False Negative (FN) rate. FN rate is the percentage of edges present in the true tree but missing in the estimated tree. So a small value of FN rate is desirable. Although there are two more common tree error measures (False Positive (FP rate) and and Robinson-Foulds (RF) rate), all of them are identical when true and estimated trees are binary [19]. In this study we worked with binary trees only.

S6 Evaluation of objective functions

In the context of phylogeny estimation, a desired objective function for MSA should lead to such alignments which can produce highly accurate (having small FN rate) ML trees. Considering this fact, we try to evaluate the effectiveness of an objective function by studying how its values are associated with the corresponding FN rates. The objective function that frequently exhibits positive correlation with FN rate is predicted to be a good optimization criteria. To accomplish this, we fit multiple linear regression model to calculate the degree of association (i.e., regression coefficient) between an objective and FN rate. Then we apply t-test, with null hypothesis that there is no association, to check the significance of individual regression coefficients. It should be noted that, such regression results does not necessarily indicate the strength of an objective as an optimization criterion. However, such results can definitely be utilized as the starting point for experimentation for further validation.

S7 Supplementary results

S7.1 Dataset statistics

S7.1.1 100-taxon simulated dataset

We used five randomly selected replicates (R0, R4, R9, R14, R19) of simulated nucleotide dataset from the study of 20. It is publicly available at `https://sites.google.com/eng.ucsd.edu/datasets/sate-i`. Table S3 gives the reference alignment statistics for this dataset.

Table S3: Reference alignments for 100-taxon simulated dataset.

Feature	Value
Number of taxa	100
Number of sites	1698.2
Percent indels	40.4
Avg. gap length	3.1

S7.1.2 Biological rRNA datasets

We analyzed two biological ribosomal RNA datasets, 23S.E and 23S.E.aa_ag, from 20 which are challenging for phylogeny estimation methods. Each of these datasets is given with a highly reliable, curated reference alignment from Gutell Lab. The statistics of the reference alignments of these datasets are presented in Table S4. Reference trees for these datasets were generated from the reference alignments by running RAxML [18] with bootstrapping, and retaining only the highly supported edges. We evaluated generated alignments with respect to the reference alignment using the tool FastSP [21].

Table S4: Reference alignments for two biological rRNA datasets.

Feature	23S.E.aa_ag	23S.E
Number of taxa	144	117
Number of sites	8,619	9,079
Percent indels	61.1	59.7
Avg. gap length	13.5	12.6

S7.1.3 BAliBASE datasets

BAliBASE 3.0 [22] is the most widely used benchmark alignment databases of protein families. It provides manually refined reference alignments of high quality based on 3D structural superposition. These datasets are organized into six groups according to their families and similarities: RV11 (very divergent sequences, residue identity below 20%), RV12 (medium to divergent sequences, 20%-40% residue identity), RV20 (families with one or more highly divergent sequences), RV30 (divergent subfamilies), RV40 (sequences with large terminal N/C extensions), and RV50 (sequences with large internal insertions). In this study, we selected four to five representative datasets from each group as reported in Table S5. We generated reference trees for these datasets by running RAxML [18] with bootstrapping. We evaluated estimated alignments with respect to the core blocks (regions for which reliable alignments are known to exist) using the program bali_score available at `http://www.lbgi.fr/balibase/BalibaseDownload/`.

Table S5: BAliBASE datasets selected for this study.

Group	Datasets selected
RV11	BB11005, BB11018, BB11020, BB11033
RV12	BB12001, BB12013, BB12022, BB12035, BB12044
RV20	BB20001, BB20010, BB20022, BB20033, BB20041
RV30	BB30002, BB30008, BB30015, BB30022
RV40	BB40001, BB40013, BB40025, BB40038, BB40048
RV50	BB50001, BB50005, BB50010, BB50016

S7.2 Selection of appropriate multi-objective formulations

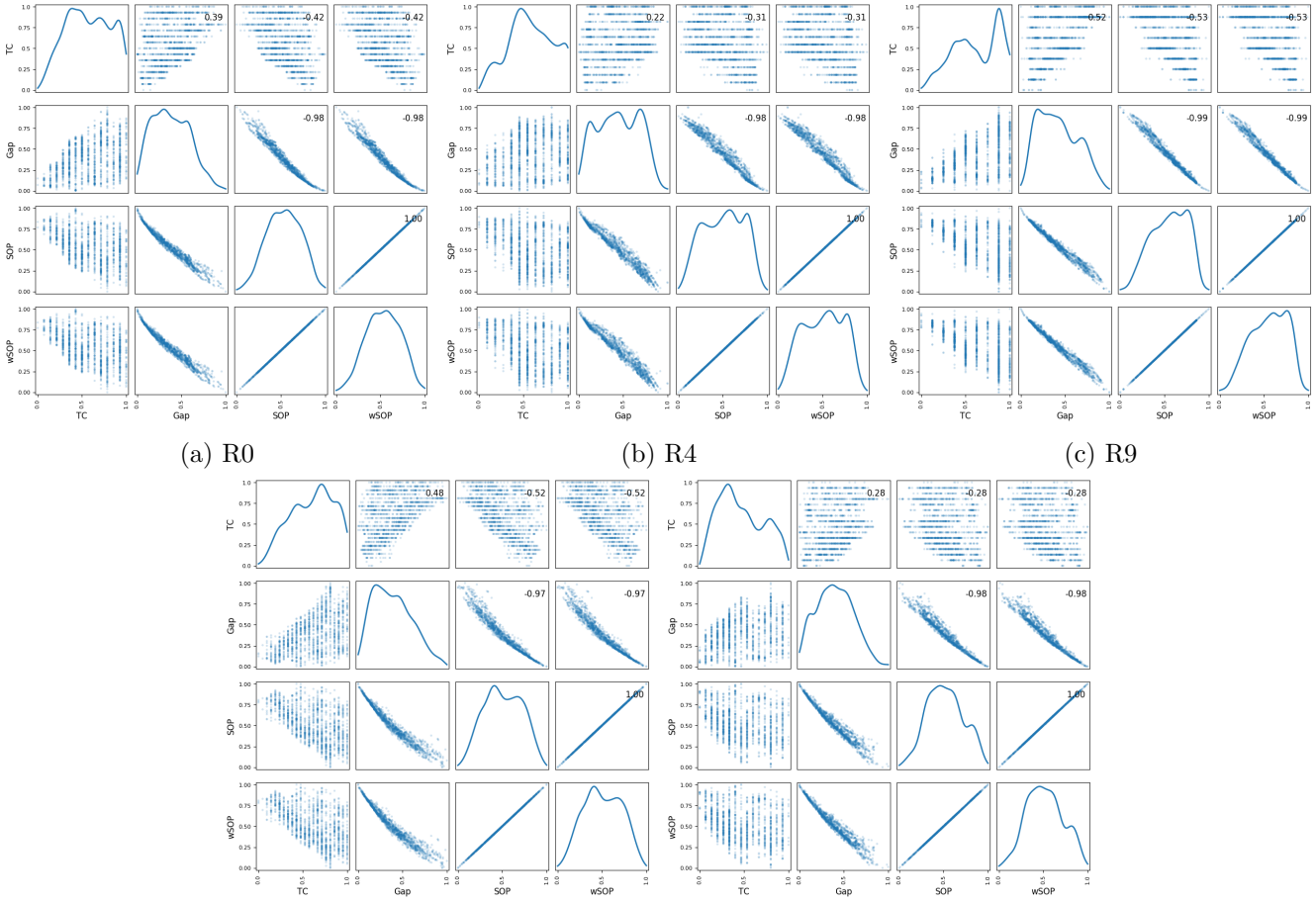
(The following should be read in conjunction with the description presented in Section III-B of the main text)

We visualize the interrelations among the objective values of the solutions, obtained by running NSGA-III to optimizes the objective set {Gap, SOP, wSOP, TC} on five randomly selected replicates (R0, R4, R9, R14, R19) of 100-taxon simulated dataset, using a 4×4 scatter-plot matrix [23] as shown in Figure S2. Here each diagonal cell of a matrix depicts the distribution of the values of an objective function estimated using kernel density estimation which is a non-parametric way to estimate the probability density function of a random variable. And the non-diagonal cells show the correlation between each pair of objective functions. As our evolutionary algorithms tries to minimize all objective functions, we treat the maximization objective values by multiplying with -1. In the sequel, we normalize all the objective values using min-max technique and as such the maximization objectives are turned into minimization ones.

We estimate the coefficients of multiple linear regression model associating FN rate with each of the objective function from {Gap, SOP, wSOP, TC} using least-squares method and illustrate them using partial regression plots [24] in Figure S3. We apply t-test on individual regression coefficient (i.e., slope) β_i (with null hypothesis $\beta_i = 0$) to test the significance of that association. The test results (slope, p-value) are incorporated in the figure.

We measure the strength of each objective set based on the FN rate achieved by the members of generated solution set. To accomplish this, For each set of objective functions, we run NSGA-II [13] for 20 times following the standard practice of operations research (OR) literature (due to the stochastic nature of metaheuristics). Each run generates a set of solutions that represents the trade-offs in satisfying all objectives. Afterwards, we inferred ML tree for each of the generated alignment. We collected the best FN rates from each of the 20 solution sets and describe the

distribution of these FN rates using boxplots which are shown in Figure S4. In these boxplots we also incorporate the FN rates achieved by the state-of-the-art tools for comparison using horizontal lines.



(a) R0 (b) R4 (c) R9

(d) R14 (e) R19

Figure S2: 100-taxon simulated dataset: Scatter-plot matrices depicting the pairwise relationship of all objective functions on five randomly selected replicates. We turn each objective function into minimization form and then normalize using min-max technique. In each matrix, the diagonal cells show the distribution of objective values (estimated using kernel density estimation which is a non-parametric way to estimate the probability density function of a random variable) while the non-diagonal cells show the correlation between pairs of objective functions. Each upper-diagonal cell contains the value of correlation coefficient r of the corresponding pair of objective functions.

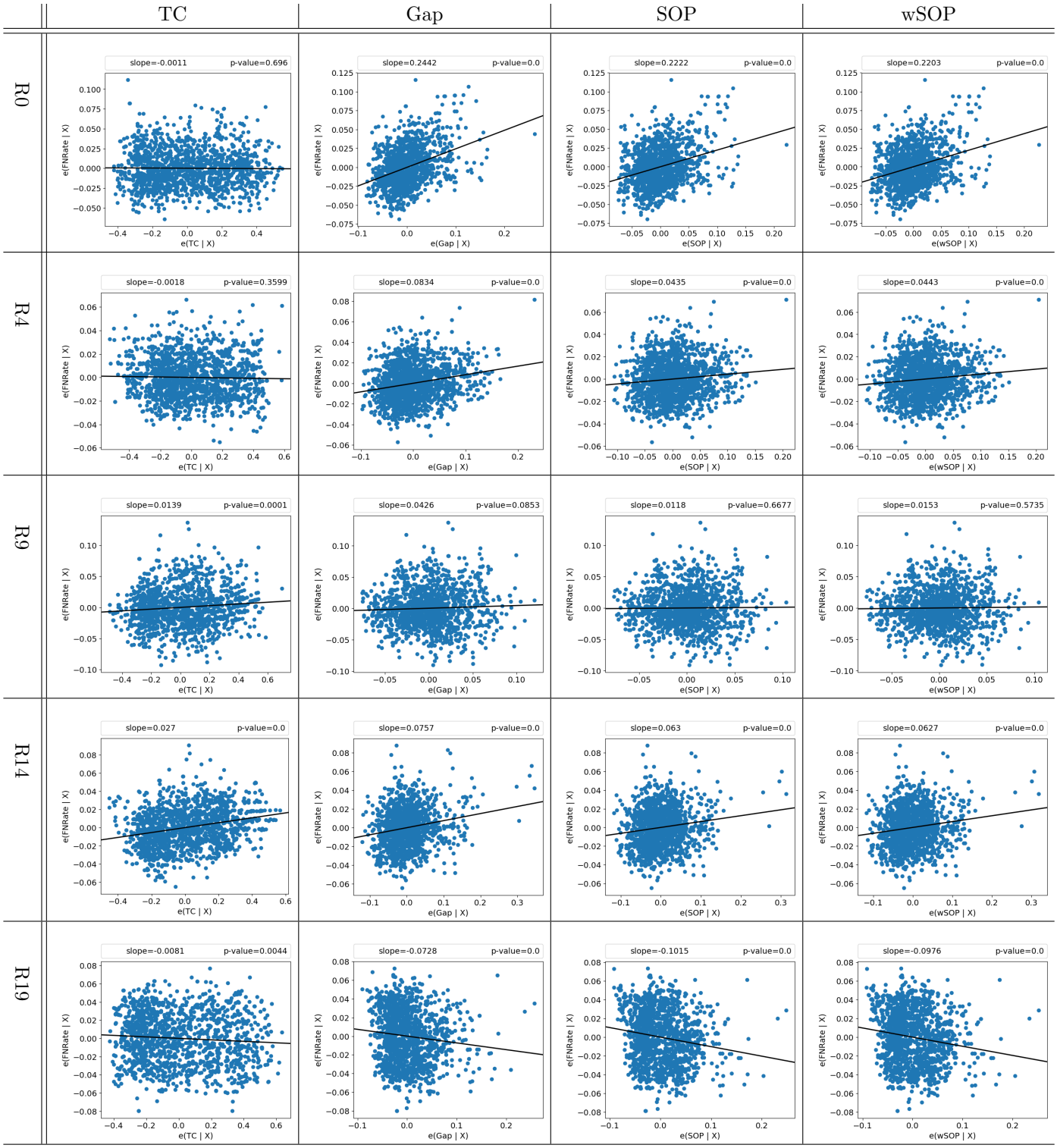
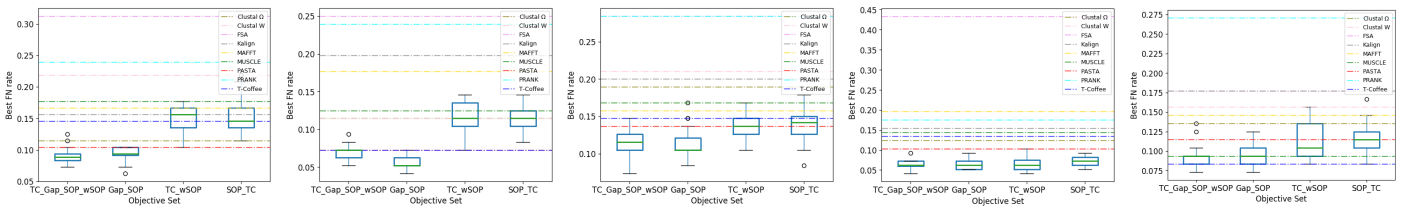
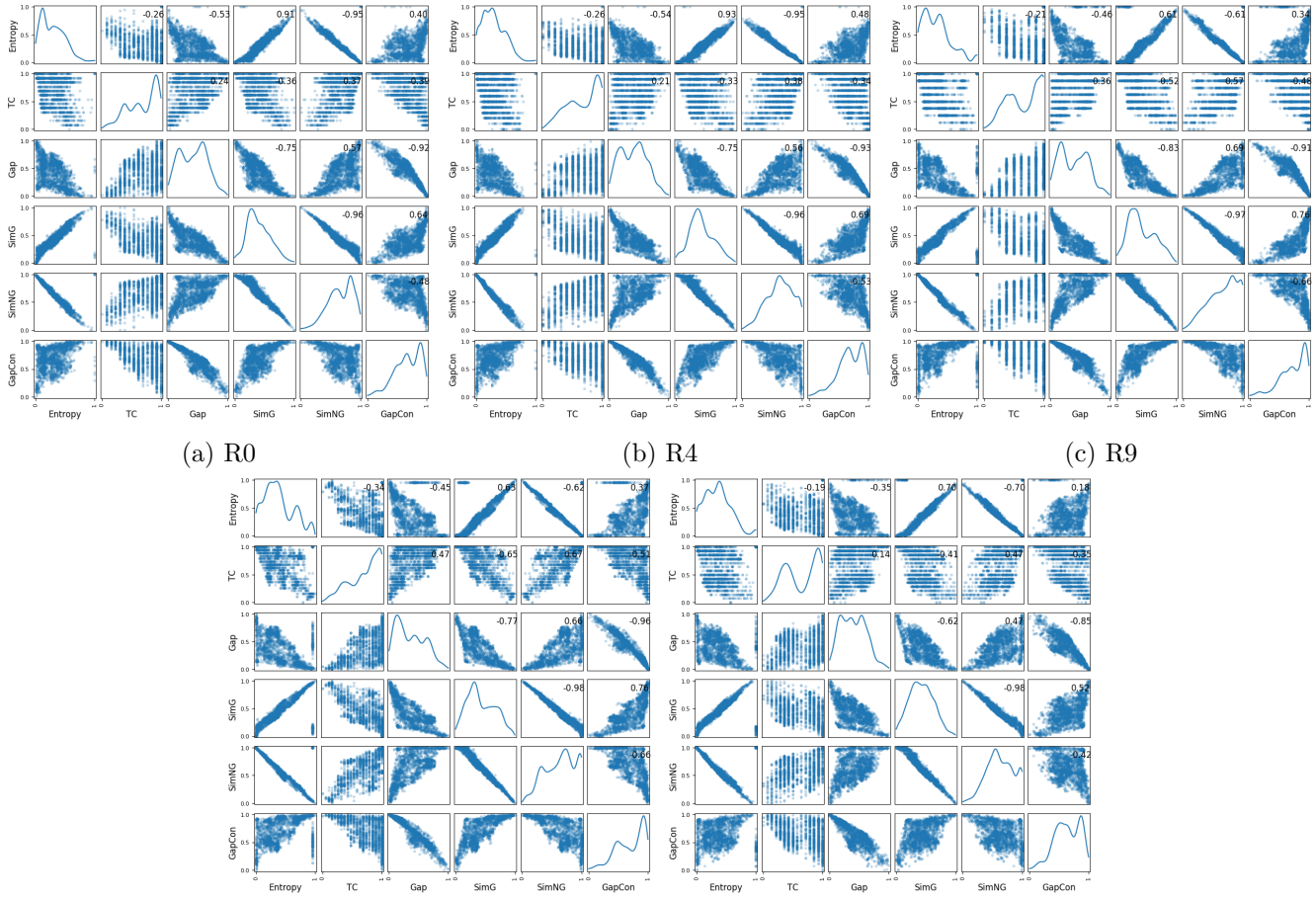


Figure S3: 100-taxon simulated dataset: Multiple linear regression model for identifying the association among FN rate and three objective functions (TC, Gap and SOP/wSOP) fitted to five randomly selected replicates. There is one figure for each possible combination (replicate, objective function). Each partial regression plot shows the association between an objective function and FN rate while holding the remaining two objectives constant. In a plot for an objective function OF, the horizontal axis, $e(OF|X)$, denotes the residuals from regressing OF against the remaining objective functions and the vertical axis, $e(FNRate|X)$, denotes the residuals from regressing FN rate against all the objective functions except OF.



(a) R0 (b) R4 (c) R9 (d) R14 (e) R19

Figure S4: 100-taxon simulated dataset: Comparison among objective sets based on the distribution of the collection of the best FN rates from each run. The performance of the state-of-the-art tools are shown using horizontal lines.



(a) R0 (b) R4 (c) R9

(d) R14 (e) R19

Figure S5: 100-taxon simulated dataset: Scatter-plot matrices depicting the pairwise relationship of all objective functions on five randomly selected replicates. We turn each objective function into minimization form and then normalize using min-max technique. In each matrix, the diagonal cells show the distribution of objective values (estimated using kernel density estimation which is a non-parametric way to estimate the probability density function of a random variable) while the non-diagonal cells show the correlation between pairs of objective functions. Each upper-diagonal cell contains the value of correlation coefficient r of the corresponding pair of objective functions.

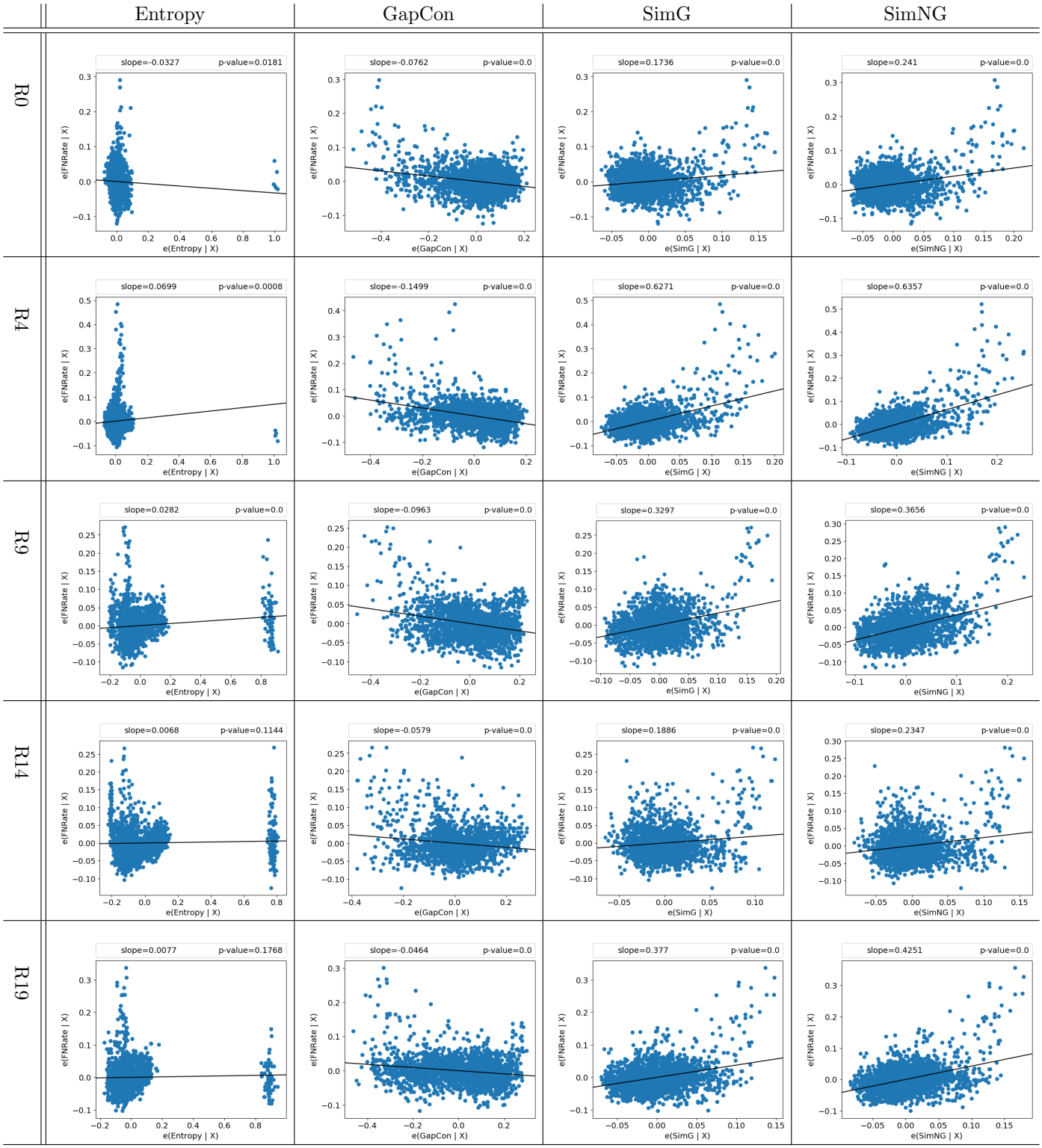
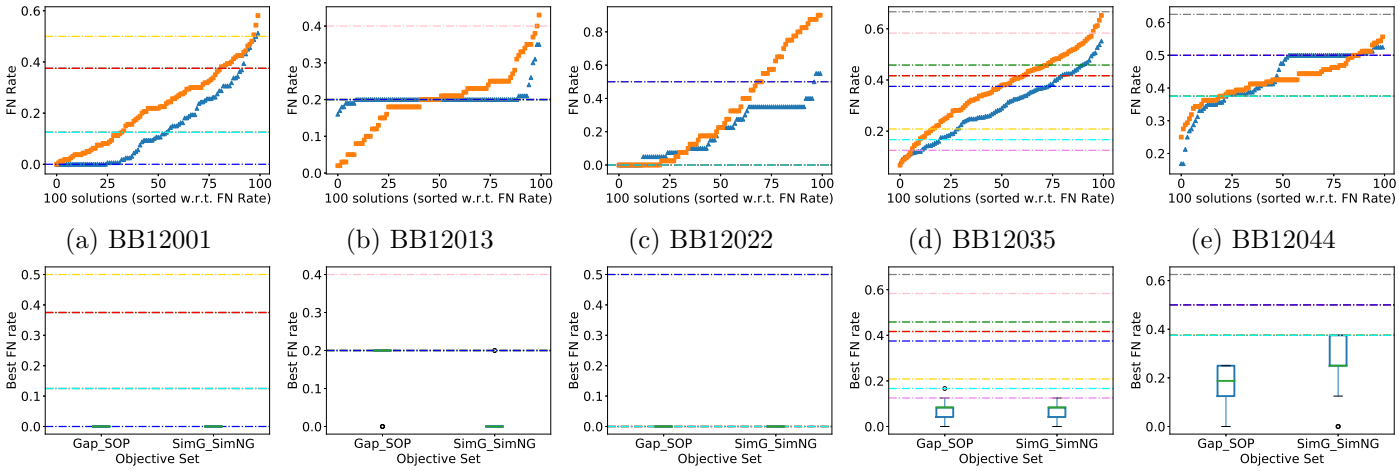


Figure S6: 100-taxon simulated dataset: Multiple linear regression model for identifying the association among FN rate and three objective functions (SimNG, GapCon and SimG/Entropy) fitted to five randomly selected replicates. There is one figure for each possible combination (replicate, objective function). Each partial regression plot shows the association between an objective function and FN rate while holding the remaining two objectives constant. In a plot for an objective function OF, the horizontal axis, $e(OF|X)$, denotes the residuals from regressing OF against the remaining objective functions and the vertical axis, $e(FNRate|X)$, denotes the residuals from regressing FN rate against all the objective functions except OF.

S7.3 Further results on BAliBASE datasets

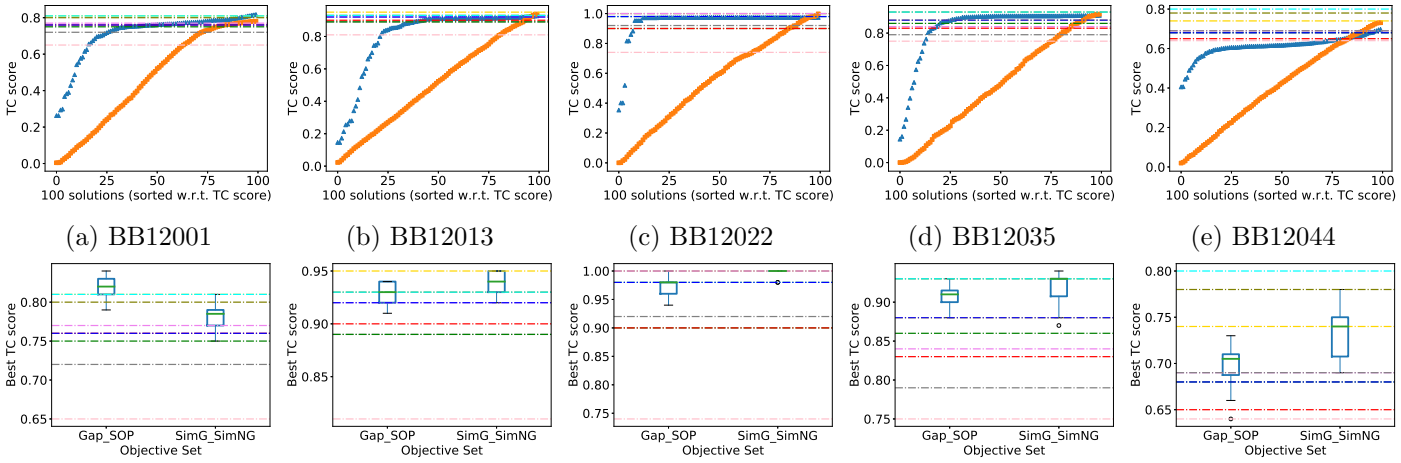
Here we first discuss our findings for the five datasets under group RV12. Here According to FN rate (Figure S7), the multi-objective formulations outperform all the state-of-the-art tools for BB12013 and BB12035. In case of BB12035, {SimG, SimNG} reconstructs all the edges correctly as opposed to 20% FN rate attained by the trees estimated on the MSA generated by the best tool which is remarkable. For the remaining datasets (BB12001 and BB12022), the multi-objective formulations perform as good as the best tool. On all the datasets, the two objective sets generate several solutions that are equivalent or better than that of the best tool. However, as observed in previous datasets, we see contrasting results with respect to TC and SP score (Figure S8,S9). Here we find only a few cases where the two objective sets can outperform the best tool. We closely analyze this issue in Figure S10 where we find that there are several solutions that achieve better FN rates in spite of their poor alignment quality (TC and SP score). For the remaining groups, our obtained results are similar. For the sake of brevity, we only illustrate the results in Figures S11 to S26.



(a) BB12001 (b) BB12013 (c) BB12022 (d) BB12035 (e) BB12044

(f) BB12001 (g) BB12013 (h) BB12022 (i) BB12035 (j) BB12044

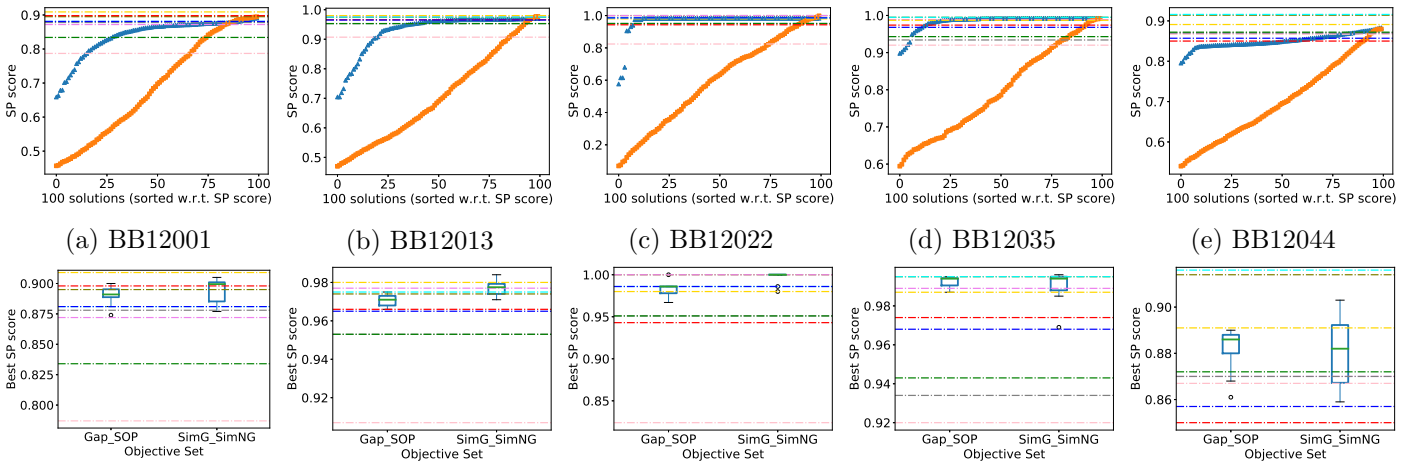
Figure S7: RV12: Top panel (part (a) - (e)) shows the FN rate of 100 solutions averaged over 20 runs. At first, we sort the FN rates of each solution set. Then we average the FN rates at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best FN rates collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB12001 (b) BB12013 (c) BB12022 (d) BB12035 (e) BB12044

(f) BB12001 (g) BB12013 (h) BB12022 (i) BB12035 (j) BB12044

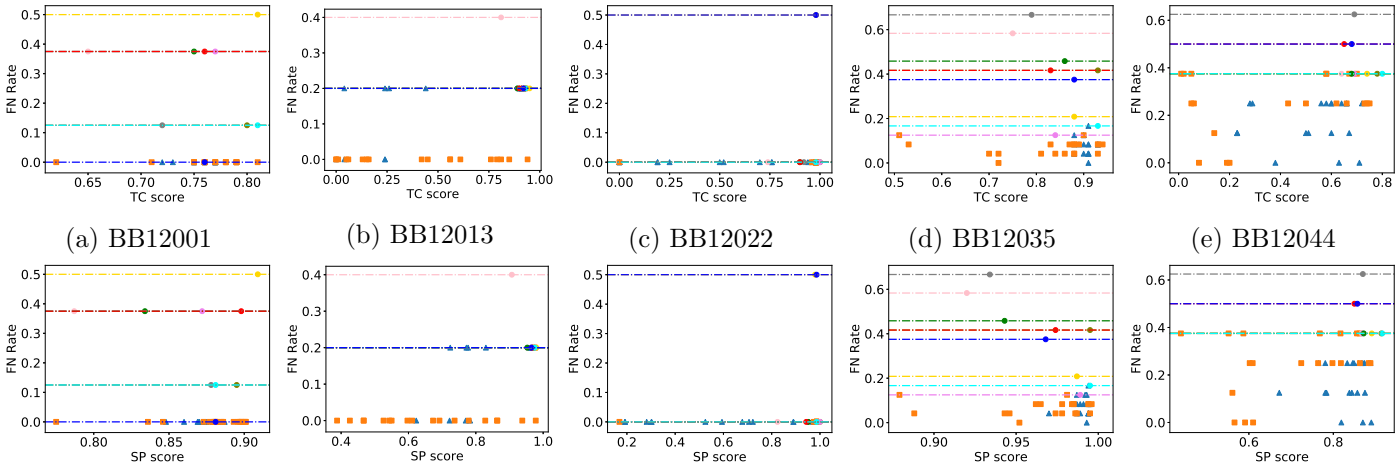
Figure S8: RV12: Top panel (part (a) - (e)) shows the TC score of 100 solutions averaged over 20 runs. At first, we sort the TC scores of each solution set. Then we average the TC scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best TC scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB12001 (b) BB12013 (c) BB12022 (d) BB12035 (e) BB12044

(f) BB12001 (g) BB12013 (h) BB12022 (i) BB12035 (j) BB12044

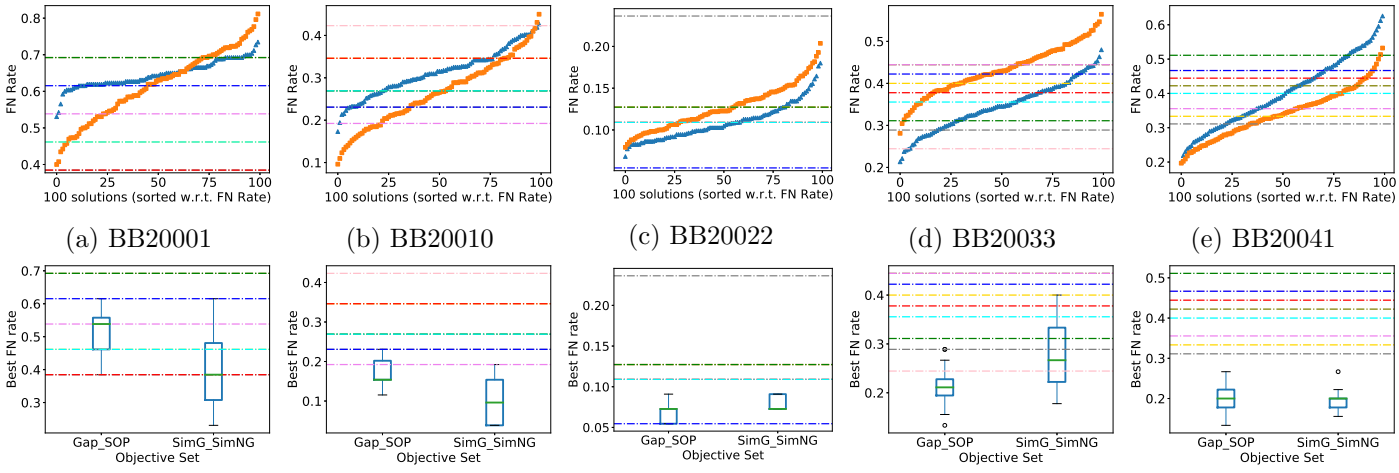
Figure S9: RV12: Top panel (part (a) - (e)) shows the SP score of 100 solutions averaged over 20 runs. At first, we sort the SP scores of each solution set. Then we average the SP scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best SP scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB12001 (b) BB12013 (c) BB12022 (d) BB12035 (e) BB12044

(f) BB12001 (g) BB12013 (h) BB12022 (i) BB12035 (j) BB12044

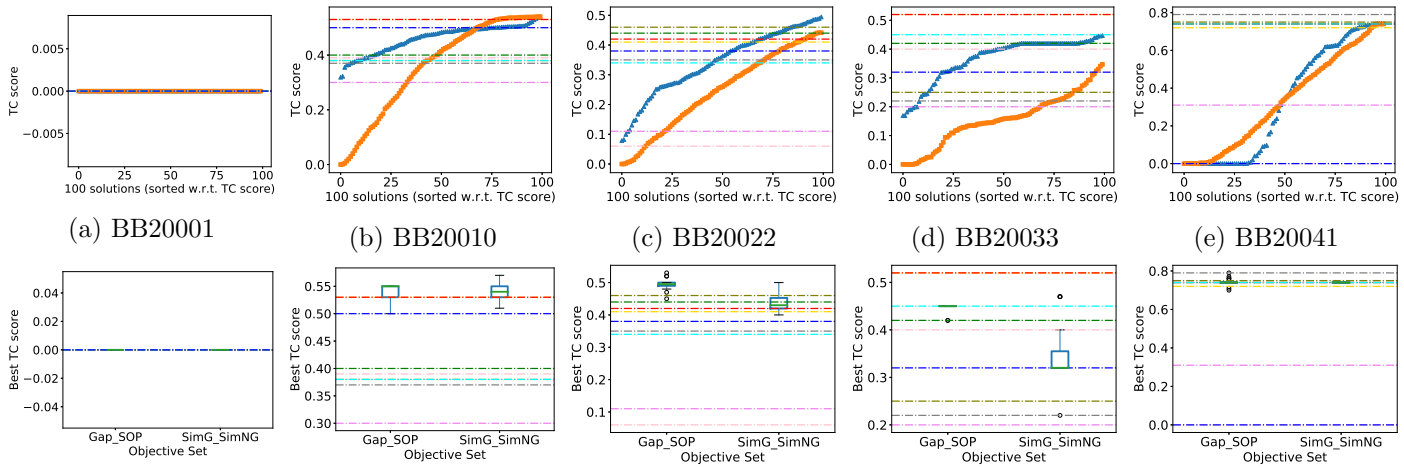
Figure S10: RV12: Top panel (part (a) - (e)) shows the relationship between FN rate and TC score for different alignments. And bottom panel (part (f) - (j)) shows the relationship between FN rate and SP score. The horizontal lines mark the FN rates achieved by the state-of-the-art tools.



(a) BB20001 (b) BB20010 (c) BB20022 (d) BB20033 (e) BB20041

(f) BB20001 (g) BB20010 (h) BB20022 (i) BB20033 (j) BB20041

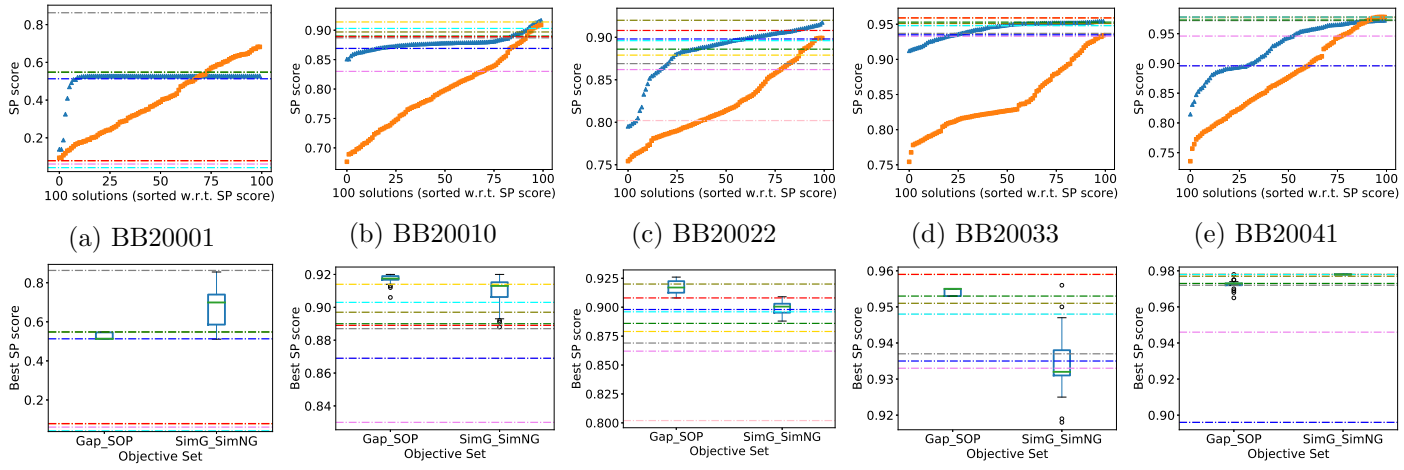
Figure S11: RV20: Top panel (part (a) - (d)) shows the FN rate of 100 final solutions averaged over 20 runs. At first, we sort the FN rates of each solution set. Then we average the FN rates at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best FN rates collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB20001 (b) BB20010 (c) BB20022 (d) BB20033 (e) BB20041

(f) BB20001 (g) BB20010 (h) BB20022 (i) BB20033 (j) BB20041

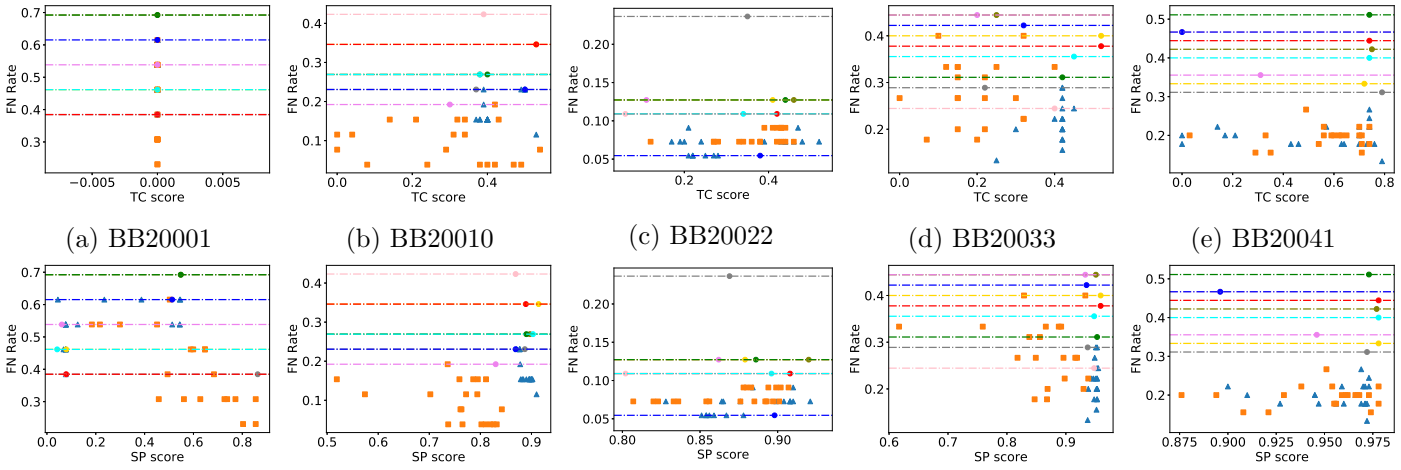
Figure S12: RV20: Top panel (part (a) - (e)) shows the TC score of 100 final solutions averaged over 20 runs. At first, we sort the TC scores of each solution set. Then we average the TC scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best TC scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB20001 (b) BB20010 (c) BB20022 (d) BB20033 (e) BB20041

(f) BB20001 (g) BB20010 (h) BB20022 (i) BB20033 (j) BB20041

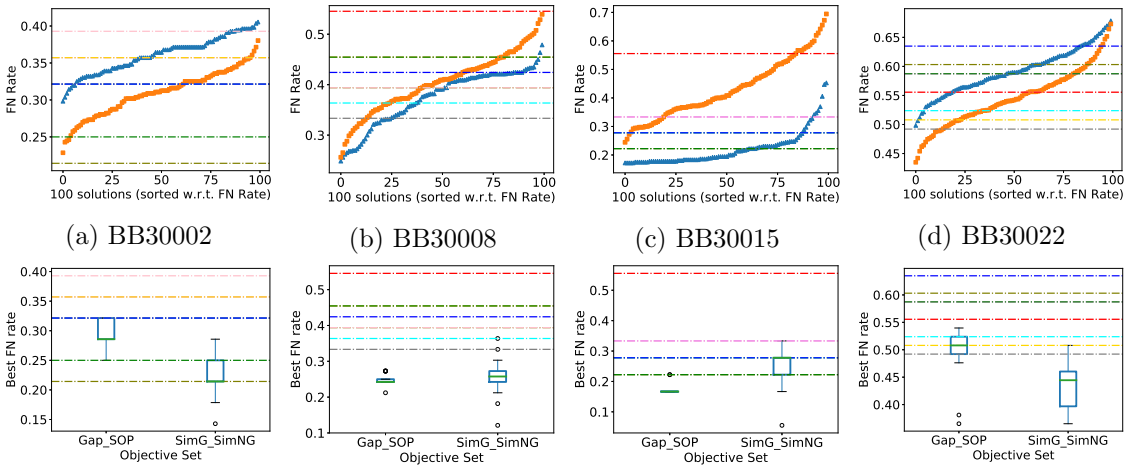
Figure S13: RV20: Top panel (part (a) - (e)) shows the SP score of 100 final solutions averaged over 20 runs. At first, we sort the SP scores of each solution set. Then we average the SP scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best SP scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB20001 (b) BB20010 (c) BB20022 (d) BB20033 (e) BB20041

(f) BB20001 (g) BB20010 (h) BB20022 (i) BB20033 (j) BB20041

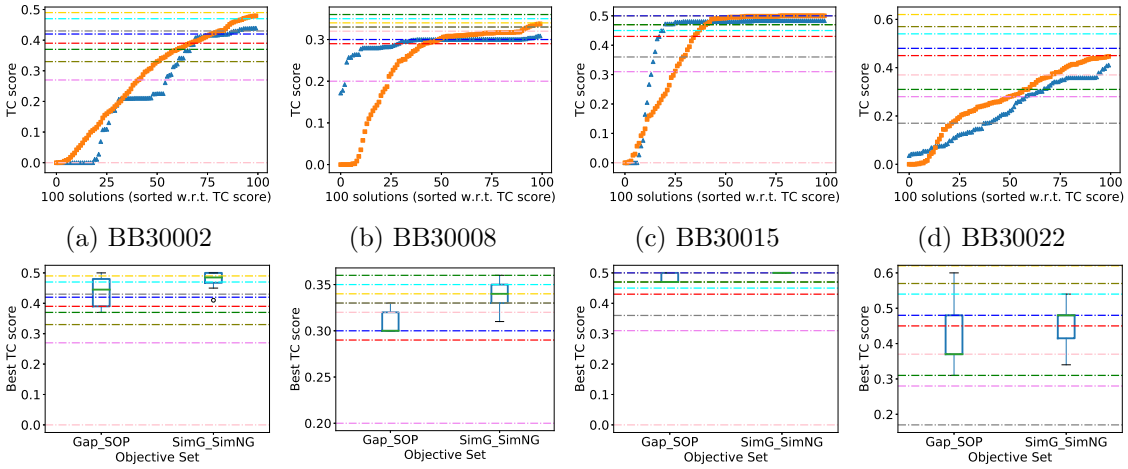
Figure S14: RV20: Top panel (part (a) - (e)) shows the relationship between FN rate and TC score for different alignments. And bottom panel (part (f) - (j)) shows the relationship between FN rate and SP score. The horizontal lines mark the FN rates achieved by the state-of-the-art tools.



(a) BB30002 (b) BB30008 (c) BB30015 (d) BB30022

(e) BB30002 (f) BB30008 (g) BB30015 (h) BB30022

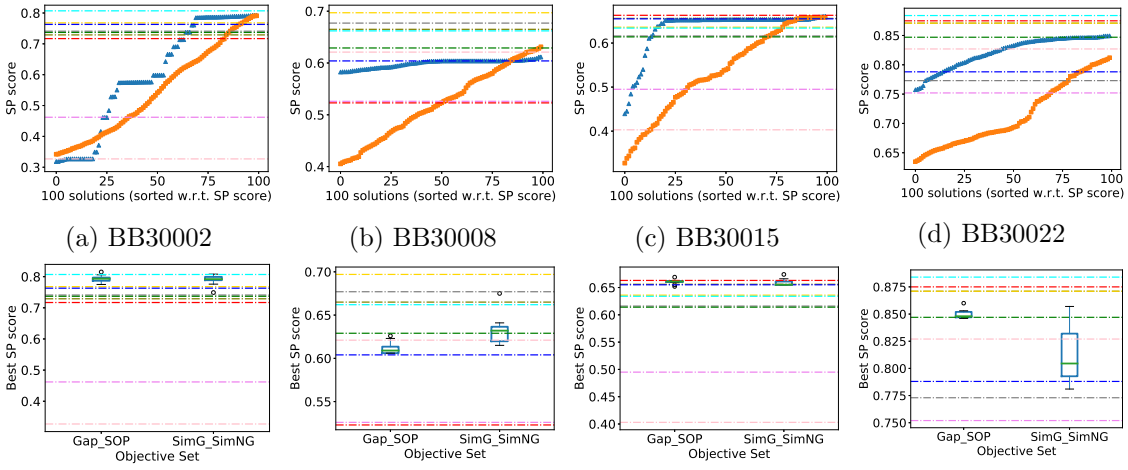
Figure S15: RV30: Top panel (part (a) - (d)) shows the FN rate of 100 final solutions averaged over 20 runs. At first, we sort the FN rates of each solution set. Then we average the FN rates at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best FN rates collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB30002 (b) BB30008 (c) BB30015 (d) BB30022

(e) BB30002 (f) BB30008 (g) BB30015 (h) BB30022

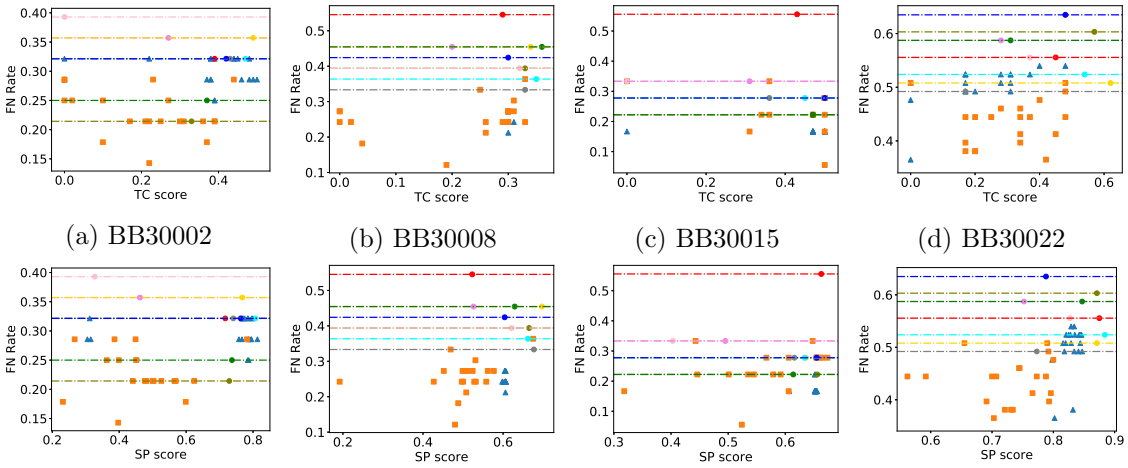
Figure S16: RV30: Top panel (part (a) - (d)) shows the TC score of 100 final solutions averaged over 20 runs. At first, we sort the TC scores of each solution set. Then we average the TC scores at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best TC scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB30002 (b) BB30008 (c) BB30015 (d) BB30022

(e) BB30002 (f) BB30008 (g) BB30015 (h) BB30022

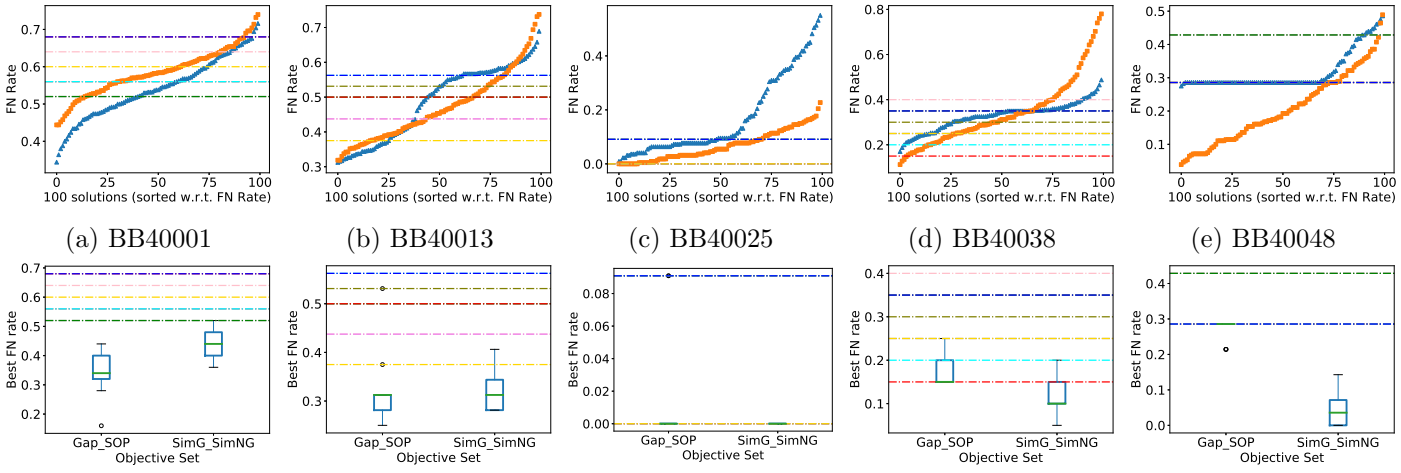
Figure S17: RV30: Top panel (part (a) - (d)) shows the SP score of 100 final solutions averaged over 20 runs. At first, we sort the SP scores of each solution set. Then we average the SP scores at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best SP scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB30002 (b) BB30008 (c) BB30015 (d) BB30022

(e) BB30002 (f) BB30008 (g) BB30015 (h) BB30022

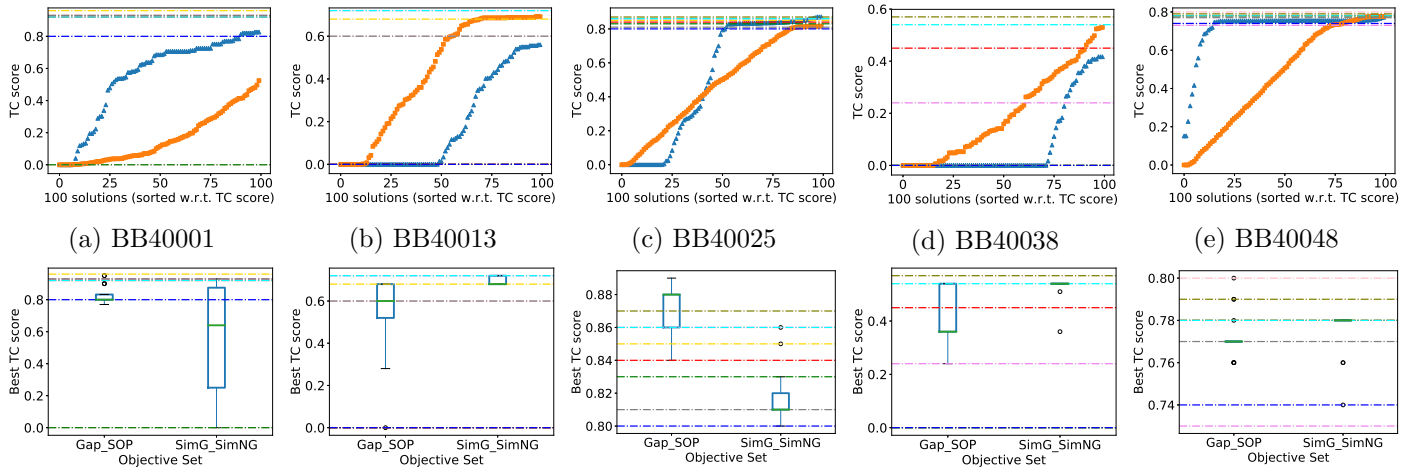
Figure S18: RV30: Top panel (part (a) - (d)) shows the relationship between FN rate and TC score for different alignments. And bottom panel (part (e) - (h)) shows the relationship between FN rate and SP score. The horizontal lines mark the FN rates achieved by the state-of-the-art tools.



(a) BB40001 (b) BB40013 (c) BB40025 (d) BB40038 (e) BB40048

(f) BB40001 (g) BB40013 (h) BB40025 (i) BB40038 (j) BB40048

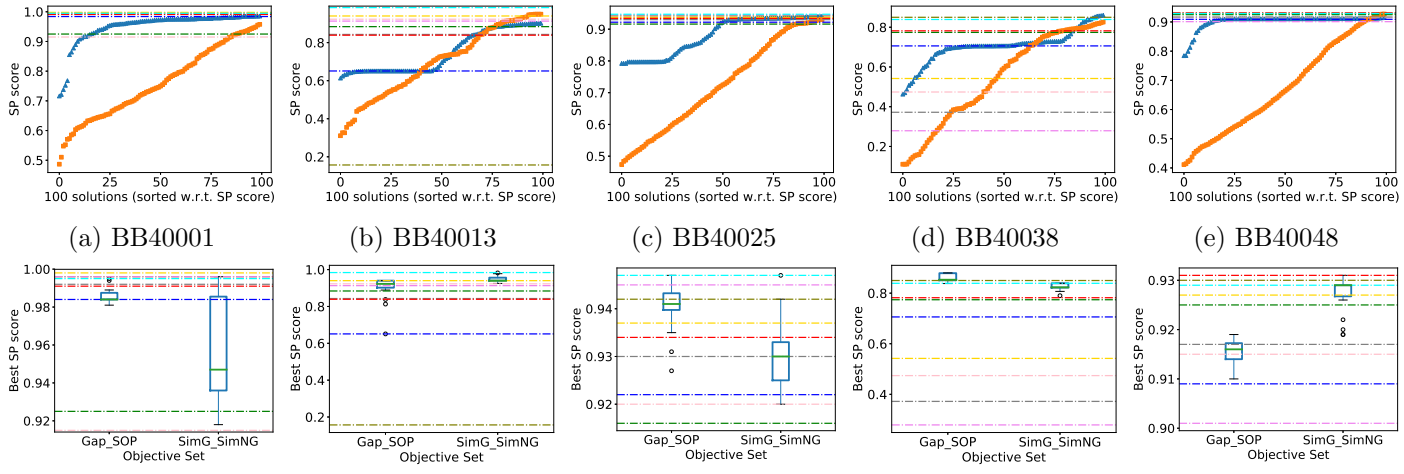
Figure S19: RV40: Top panel (part (a) - (d)) shows the FN rate of 100 final solutions averaged over 20 runs. At first, we sort the FN rates of each solution set. Then we average the FN rates at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best FN rates collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB40001 (b) BB40013 (c) BB40025 (d) BB40038 (e) BB40048

(f) BB40001 (g) BB40013 (h) BB40025 (i) BB40038 (j) BB40048

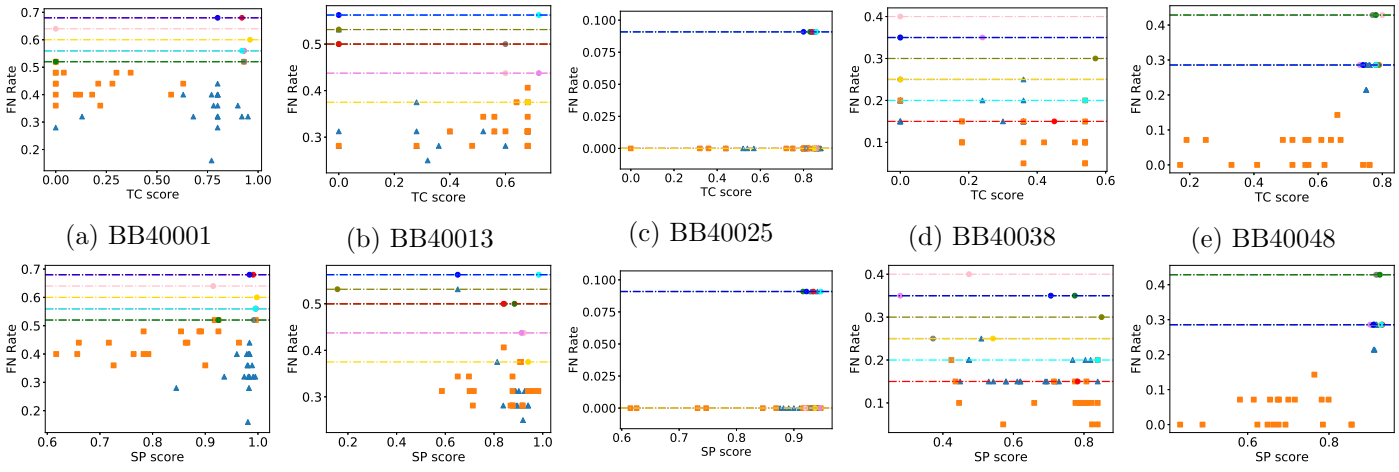
Figure S20: RV40: Top panel (part (a) - (e)) shows the TC score of 100 final solutions averaged over 20 runs. At first, we sort the TC scores of each solution set. Then we average the TC scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best TC scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB40001 (b) BB40013 (c) BB40025 (d) BB40038 (e) BB40048

(f) BB40001 (g) BB40013 (h) BB40025 (i) BB40038 (j) BB40048

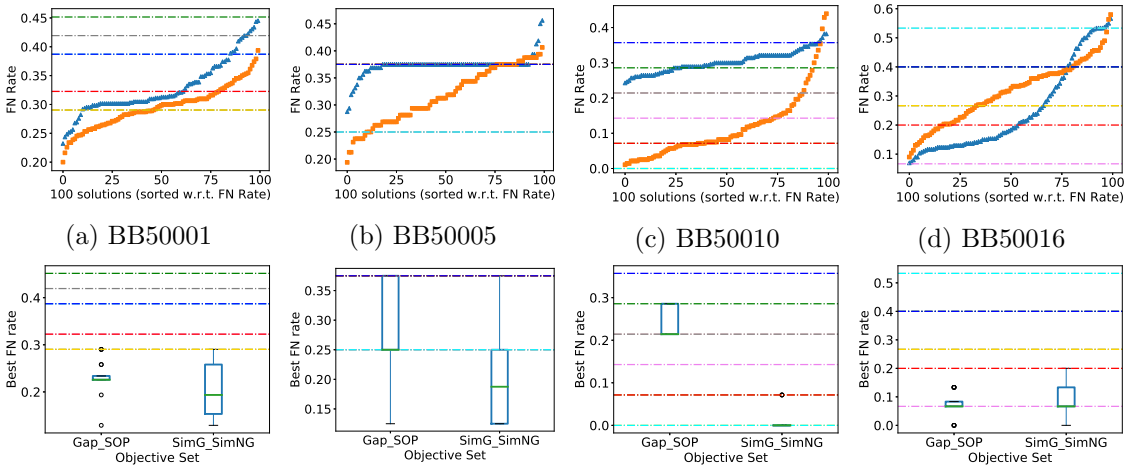
Figure S21: RV40: Top panel (part (a) - (e)) shows the SP score of 100 final solutions averaged over 20 runs. At first, we sort the SP scores of each solution set. Then we average the SP scores at each sorted position of all the sets. Bottom panel (part (f) - (j)) shows the distribution of the best SP scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB40001 (b) BB40013 (c) BB40025 (d) BB40038 (e) BB40048

(f) BB40001 (g) BB40013 (h) BB40025 (i) BB40038 (j) BB40048

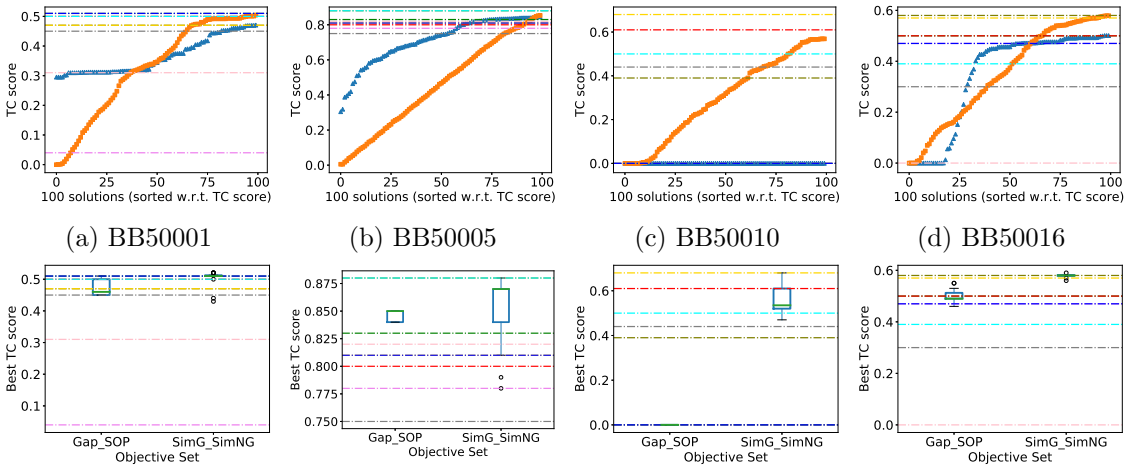
Figure S22: RV40: Top panel (part (a) - (e)) shows the relationship between FN rate and TC score for different alignments. And bottom panel (part (f) - (j)) shows the relationship between FN rate and SP score. The horizontal lines mark the FN rates achieved by the state-of-the-art tools.



(a) BB50001 (b) BB50005 (c) BB50010 (d) BB50016

(e) BB50001 (f) BB50005 (g) BB50010 (h) BB50016

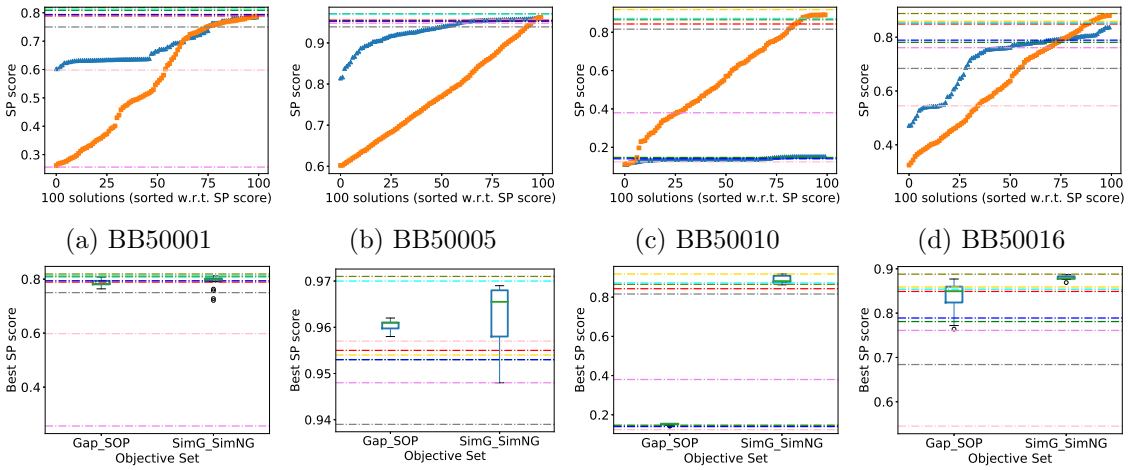
Figure S23: RV50: Top panel (part (a) - (d)) shows the FN rate of 100 final solutions averaged over 20 runs. At first, we sort the FN rates of each solution set. Then we average the FN rates at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best FN rates collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB50001 (b) BB50005 (c) BB50010 (d) BB50016

(e) BB50001 (f) BB50005 (g) BB50010 (h) BB50016

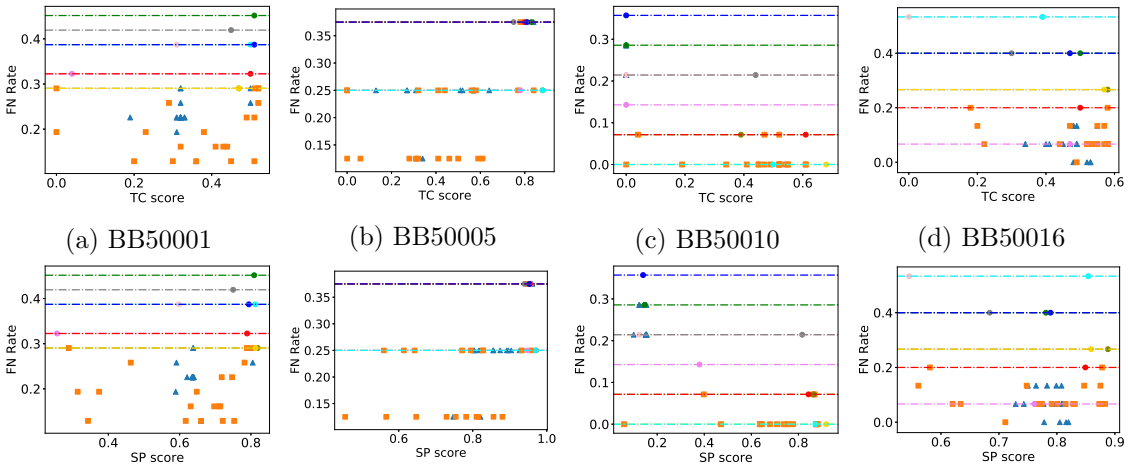
Figure S24: RV50: Top panel (part (a) - (d)) shows the TC score of 100 final solutions averaged over 20 runs. At first, we sort the TC scores of each solution set. Then we average the TC scores at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best TC scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB50001 (b) BB50005 (c) BB50010 (d) BB50016

(e) BB50001 (f) BB50005 (g) BB50010 (h) BB50016

Figure S25: RV50: Top panel (part (a) - (d)) shows the SP score of 100 final solutions averaged over 20 runs. At first, we sort the SP scores of each solution set. Then we average the SP scores at each sorted position of all the sets. Bottom panel (part (e) - (h)) shows the distribution of the best SP scores collected from all runs. In each figure, the horizontal lines show the performance of the state-of-the-art tools.



(a) BB50001 (b) BB50005 (c) BB50010 (d) BB50016

(e) BB50001 (f) BB50005 (g) BB50010 (h) BB50016

Figure S26: RV50: Top panel (part (a) - (d)) shows the relationship between FN rate and TC score for different alignments. And bottom panel (part (e) - (h)) shows the relationship between FN rate and SP score. The horizontal lines mark the FN rates achieved by the state-of-the-art tools.

S7.4 Computational time

We ran the multi-objective metaheuristics on a server with Intel(R) Xeon(R) CPU E5-4617 @ 2.90GHz processor and 64GB of RAM. In Table S6, we give a rough estimate of the total computational time that we invested to derive our results.

Table S6: Computational time invested to study the impact of multi-objective formualtion of MSA.

Dataset	Total time (hours)
100-taxon simulated dataset	1269.38
Biological rRNA dataset	311.64
BAliBASE dataset	45.88

References

[1] Pasut Seeluangsawat and Prabhas Chongstitvatana. A multiple objective evolutionary algorithm for multiple sequence alignment. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pages 477–478. ACM, 2005.

[2] Fernando José Mateus da Silva, Juan Manuel Sánchez Pérez, Juan Antonio Gómez Pulido, and Miguel A Vega Rodríguez. Alineaga—a genetic algorithm with local search optimization for multiple sequence alignment. *Applied Intelligence*, 32(2):164–172, 2010.

[3] Álvaro Rubio-Largo, Miguel A Vega-Rodríguez, and David L González-Álvarez. A hybrid multiobjective memetic metaheuristic for multiple sequence alignment. *IEEE Transactions on Evolutionary Computation*, 20(4):499–514, 2016.

[4] Wilson Soto and David Becerra. A multi-objective evolutionary algorithm for improving multiple sequence alignments. In *Brazilian Symposium on Bioinformatics*, pages 73–82. Springer, 2014.

[5] Mehmet Kaya, Abdullah Sarhan, and Reda Alhajj. Multiple sequence alignment with affine gap by using multi-objective genetic algorithm. *Computer methods and programs in biomedicine*, 114(1):38–49, 2014.

[6] Huazheng Zhu, Zhongshi He, and Yuanyuan Jia. A novel approach to multiple sequence alignment using multiobjective evolutionary algorithm based on decomposition. *IEEE journal of biomedical and health informatics*, 20(2):717–727, 2016.

[7] R Ranjani Rani and D Ramyachitra. Multiple sequence alignment using multi-objective based bacterial foraging optimization algorithm. *Biosystems*, 150:177–189, 2016.

[8] Álvaro Rubio-Largo, Miguel A Vega-Rodríguez, and David L González-Álvarez. Hybrid multiobjective artificial bee colony for multiple sequence alignment. *Applied Soft Computing*, 41:157–168, 2016.

[9] Francisco M Ortuño, Olga Valenzuela, Fernando Rojas, Hector Pomares, Javier P Florido, Jose M Urquiza, and Ignacio Rojas. Optimizing multiple sequence alignments using a genetic algorithm based on three objectives: structural information, non-gaps percentage and totally conserved columns. *Bioinformatics*, 29(17):2112–2121, 2013.

[10] Cristian Zambrano-Vega, Antonio J Nebro, José García-Nieto, and José F Aldana-Montes. Comparing multi-objective metaheuristics for solving a three-objective formulation of multiple sequence alignment. *Progress in Artificial Intelligence*, pages 1–16, 2017.

[11] Maryam Abbasi, Luís Paquete, and Francisco B Pereira. Local search for multiobjective multiple sequence alignment. In *International Conference on Bioinformatics and Biomedical Engineering*, pages 175–182. Springer, 2015.

[12] Shengxiang Yang, Miqing Li, Xiaohui Liu, and Jinhua Zheng. A grid-based evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 17(5):721–736, 2013.

[13] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[14] Kalyanmoy Deb and Himanshu Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Trans. Evolutionary Computation*, 18(4):577–601, 2014.

[15] Cristian Zambrano-Vega, Antonio J Nebro, José García-Nieto, and José F Aldana-Montes. A multi-objective optimization framework for multiple sequence alignment with metaheuristics. In *International Conference on Bioinformatics and Biomedical Engineering*, pages 245–256. Springer, 2017.

[16] Kevin Liu, C Randal Linder, and Tandy Warnow. Raxml and fasttree: comparing two methods for large-scale maximum likelihood phylogeny estimation. *PloS one*, 6(11):e27731, 2011.

[17] Morgan N Price, Paramvir S Dehal, and Adam P Arkin. Fasttree 2–approximately maximum-likelihood trees for large alignments. *PloS one*, 5(3):e9490, 2010.

[18] Alexandros Stamatakis. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

[19] Tandy Warnow. *Computational phylogenetics: an introduction to designing methods for phylogeny estimation*. Cambridge University Press, 2017.

[20] Kevin Liu, Sindhu Raghavan, Serita Nelesen, C Randal Linder, and Tandy Warnow. Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. *Science*, 324(5934):1561–1564, 2009.

[21] Siavash Mirarab and Tandy Warnow. Fastsp: linear time calculation of alignment accuracy. *Bioinformatics*, 27(23):3250–3258, 2011.

[22] Julie D Thompson, Patrice Koehl, Raymond Ripp, and Olivier Poch. Balibase 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics*, 61(1):127–136, 2005.

[23] Deb Kalyanmoy. *Multi objective optimization using evolutionary algorithms*. John Wiley and Sons, 2001.

[24] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*, volume 821. John Wiley & Sons, 2012.