Comparative Genomics on Artificial Life

Priscila Biller[1,2], Carole Knibbe[2,3], Guillaume Beslon[2,3], and Eric Tannier[2,4(✉)]

[1] University of Campinas, São Paulo, Brazil
[2] INRIA Grenoble Rhône-Alpes, 38334 Montbonnot, France
eric.tannier@inria.fr
[3] Université Lyon 1, LIRIS, UMR5205, 69622 Villeurbanne, France
[4] Université Lyon 1, LBBE, UMR5558, 69622 Villeurbanne, France

Abstract. Molecular evolutionary methods and tools are difficult to validate as we have almost no direct access to ancient molecules. Inference methods may be tested with simulated data, producing full scenarios they can be compared with. But often simulations design is concomitant with the design of a particular method, developed by a same team, based on the same assumptions, when both should be blind to each other. *In silico* experimental evolution consists in evolving digital organisms with the aim of testing or discovering complex evolutionary processes. Models were not designed with a particular inference method in mind, only with basic biological principles. As such they provide a unique opportunity to blind test the behavior of inference methods. We give a proof of this concept on a comparative genomics problem: inferring the number of inversions separating two genomes. We use Aevol, an *in silico* experimental evolution platform, to produce benchmarks, and show that most combinatorial or statistical estimators of the number of inversions fail on this dataset while they were behaving perfectly on ad-hoc simulations. We argue that biological data is probably closer to the difficult situation.

Keywords: Comparative genomics · In silico experimental evolution · Benchmark · Rearrangements

1 Validation of Evolutionary Inferences

The comparative method in evolutionary biology consists in detecting similarities and differences between extant organisms, and, based on more or less formalized hypotheses on the evolutionary processes, infer ancestral states explaining the similarities and an evolutionary history explaining the differences.

A common concern in all evolutionary studies is the validity of the methods and results. Results concern events that were supposed to occur in a deep past (up to 4 billion years) and they have no other trace today than the present molecules used by the comparative method.

As we cannot travel back in time to verify the results, there are several ways to assess the validity of molecular evolution studies: theoretical considerations about the models and methods (realism, consistency, computational complexity,

A. Beckmann et al. (Eds.): CiE 2016, LNCS 9709, pp. 35–44, 2016.
DOI: 10.1007/978-3-319-40189-8_4

model testing, ability to generate a statistical support or a variety of the solutions) [24], coherence with fossil records [26], or ancient DNA [11], or empirical tests when the solution is known, on experimental evolution [17] or simulations. Each method has its caveats. Models for inference have to adopt a compromise between realism, consistency and complexity. Ancient DNA is rarely available, usually not in an assembled shape. Fossils are also rare and provide a biased sampling of ancient diversity. Experimental evolution is expensive, time-consuming and limited in the number of generations it can provide.

Simulation is the most popular validation tool. Genome evolution can be simulated *in silico* for a much higher number of generations than in experimental evolution, at a lower cost. All the history can be recorded in details, and compared with the inference results. A problem with simulations, however, is that they necessarily oversimplify genome evolution processes. Moreover, very often, even if they are designed to be used by another team for inference [4,10,14,15,23], they encode the same simplifications as the inference methods. For example, only fixed mutations are generated because only these are visible by inference methods, selection is tuned to fit what is visible by the inference methods; genes are evolutionary units in simulations because they are the units taken for inference. Everything is designed thinking of the possibilities of the inference methods, leading to easy unrealistic instances.

This mode of *ad-hoc* simulation has been widely applied to test estimators of rearrangement distances, and in particular inversion distances [5,7,9,12,22]. The problem consists in comparing two genomes and estimating the number of inversions (a rearrangement that reverses the reading direction of a genomic segment) that have occurred in the evolutionary lineages separating them. To construct a solution, conserved genes or synteny blocks are detected in the two genomes, and a number of inversions explaining the differences in gene orders is estimated. A lot of work has consisted in finding shortest scenarios [13]. Statistical estimations need a model. The standard and most used model depicts genomes as permutations of genes and assumes that an inversion reverses a segment of the permutation, taken uniformly at random over all segments. When simulators are designed to validate the estimators, they also use permutations as models of gene orders, and inversions on segments of this permutations, chosen uniformly at random. Estimators show good performances on such simulations, but transforming a genome into a permutation of genes is such a simplification from both parts that it means nothing about any ability to estimate a rearrangement distance in biological data [8].

We propose to use simulations that were not designed for validation purposes. It is the case, in artificial life, of *in silico* experimental evolution [18], and in particular of the *Aevol* platform [3,19]. Aevol contains, among many other features, all what is needed to test rearrangement inference methods. The genomes have gene sequences and non coding sequences organized in a chromosome, and evolve with inversions, in addition to substitutions, indels, duplications, losses, translocations. Rearrangements are chosen with a uniform random model on the genome, which should fit the goals of the statistical estimators, but is different

from a uniform random model on permutations [8]. We tested 10 different estimators of inversion distance on 18 different datasets generated by Aevol. The difference with ad-hoc simulations is striking. Most estimators completely fail to give a close estimate in a vast majority of conditions. We argue that the reason for this failure lies in realistic features in artificial genomes that are very likely to reproduce the failure on real data.

We first describe the principle of the estimators, then the principles of the simulator, with its goals and its functioning. We will show how to process its results to test statistical estimators of rearrangement distances.

2 Comparative Genomics: Estimating an Inversion Distance

We tested 10 estimators of the number of inversions separating two genomes, called ID (the inversion distance) [16], CL for Caprara and Lancia [9], EH for Eriksen and Hultman [12], Badger [20], BD for Berestycki and Durrett [5], LM for Lin and Moret [22], BGT for Biller, Guéguen and Tannier [7], AA for Alexeev and Alekseyev [2], ER1 and ER2 for Erdös-Renyi 1 and 2 [8].

For 8 of them (ID, LM, BGT, Badger, EH, BD, CL, AA), a genome is defined as a signed permutation, π over $\{1, \ldots, n\}$, that is, an ordering of the elements of $\{1, \ldots, n\}$ where each element is given a sign, $+$ or $-$ ($+$ usually omitted), representing the reading direction of an element. The elements of the permutation are *genes*, or *solid regions*, the ones that are never cut by inversions. All inversions have the same probability. For the two remaining estimators (ER1 and ER2), a genome is made up of two components: the same signed permutation, and in addition a vector p of $n+1$ breakage probabilities, $p_i > 0$, $0 \leq i \leq n$, with $\sum_i p_i = 1$. An inversion of the segment $[\pi_i, \ldots, \pi_j]$ has probability $p_{i-1}p_j$.

Suppose A and B are two signed permutations. We define the *breakpoint graph* of A and B as the graph with $2n+2$ vertices and $2n+2$ edges: for each element $i \in \{1, \ldots, n\}$, define two vertices i_t and i_h, plus two additional vertices 0_h and $n+1_t$; then for any two consecutive numbers ab of A, join two extremities by an A-edge: first is a_h if a is positive, a_t otherwise, second is b_t if b is positive, b_h otherwise. Additionally, if a is the first element of the permutation, join 0_h and a_t if a is positive, a_h otherwise, and if b is the last element of the permutation, join $n+1_t$ and b_h if b is positive, b_t otherwise. Do the same for B, and call the edges B-edges.

An *adjacency* of a genome A is an A-edge in the breakpoint graph. It is a *common adjacency* with a genome B if it is also a B-edge, otherwise it is a *breakpoint*. Breakpoint graphs have a uniform degree of 2 on all vertices, thus they are sets of disjoint cycles alternating between A-edges and B-edges. We note b the number of breakpoints, c the number of cycles of the breakpoint graph, and c_2 the number of cycles with 4 edges.

The parsimony estimator (ID) is the minimum number of inversions necessary to transform A into B, which is close to $n+1-c$ [16]. Badger is a Bayesian sampler of inversion scenarios and computes an *a posteriori* probable distance.

The others all work with the method of moments. This consists in computing an expected value for one or two observable parameters of the breakpoint graph (b, c_2, c or a combination of two of them) if A and B are separated by k inversions. It is a function of k and n: $f_n(k)$. It is never computed exactly, approximate formulas or computation principles are given. Then k is estimated as $\hat{k} = f_n^{-1}(p)$ for the observed value p of the parameter. LM, CL, BGT, ER1 are based on the expected value of b. EH and BD are based on the expected value of c. ER2 is based on expected values for b and c_2, and AA uses expected values for b and c. The two latter use two values because they also consider n as unknown and estimate it as well as k.

3 Artificial Life: *In Silico* Experimental Evolution and the Aevol Platform

Unlike many simulators used to validate phylogenetic inference methods [4,10, 14,15,23], Aevol does not represent a species by a single lineage undergoing fixed mutations. Like forward-in-time simulators used in population genetics, it explicitly represents all genotypes present in the population and simulates spontaneous mutations, which can be deleterious, neutral or beneficial. An important difference, however, is that the selection coefficients of mutations are not predefined for each locus nor drawn from a random distribution. Instead, an artificial chemistry is used to decode each genome present in the population and compute its phenotype, which is its ability to perform a computational task (see details below). Point mutations or small indels can alter gene sequences and non coding sequences. A local mutation in a gene can have a different effect on phenotype and fitness, depending on the genomic background (other genes). Chromosomal rearrangements like duplications, deletions, translocations or inversions can occur anywhere in the chromosome sequence. They can alter gene number and gene order and disrupt genes.

Figure 1 summarizes the functioning of Aevol. We give a high level description here, and emphasize that the tool has many other possibilities than being used as a bench mark. For a complete description and some of its possibilities, see [3,19]. Genomes are circular sequences on a binary alphabet. A population of typically 1000 genomes lives at a given generation. Genes are segments situated on a transcribed sequence (*i.e.*, a sequence starting after a promoter and ending at a terminator sequence) starting after a Ribosome Binding Start and a Start codon and ended by a Stop codon on the same reading frame. Inside a gene, a coding sequence is translated into a protein sequence using a genetic code on size three codons. This protein sequence encodes the parameters of a piecewise linear function that indicates the contribution (in $[-1, 1]$) of the protein to each abstract "phenotypic trait" in $[0, 1]$. All proteins encoded in a genome are summed to produce the phenotype, which is thus a piece-wise linear function indicating the level of each phenotypic trait in $[0, 1]$.

This phenotype is then compared with a target function indirectly representing the environment of the individual. The difference between the two is

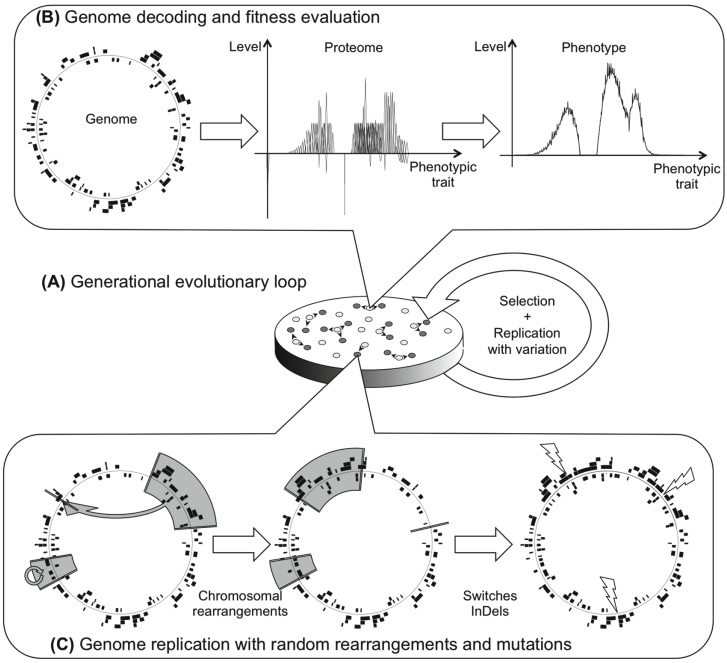

Fig. 1. Overview of the Aevol model. **(A)** A population of genomes is simulated. At each generation, all genomes are evaluated and the fittest ones are replicated to produce the next generation. The replication process includes variation operators. The joint actions of selection, genetic drift and variation make the population evolve. **(B)** Overview of the genome decoding process. Left: Each individual own a circular double-strand genome with scattered genes. Right: The individual's phenotype is the level of each abstract phenotypic trait in [0, 1]. It is compared a target representing the optimal phenotype given the environment. Middle: Each gene is decoded into a protein that contributes to a small subset of phenotypic traits. More precisely, the sequence of the gene is decoded into three reals that specify the mean, width and height of a triangular kernel function. All the proteins are then summed up to compute the phenotype. The individual displayed here was obtained after 460.000 generations of evolution in Aevol under a mutational pressure of 10^{-6} mutations/bp/generation for local events and 10^{-5} mutations/bp/generation for chromosomal rearrangements (see below). Its genome is 6898 bp long. It encodes 113 genes and 35 RNAs (not shown). 28.4 % of the genome is non-coding. **(C)** Overview of the replication process. During its replication each genome may undergo chromosomal rearrangements affecting large DNA segments (here an inversion and a translocation) and local mutations (point mutations or small InDels).

used to compute the *fitness* of the genome. To produce the next generation, genomes with high fitness are replicated in the following generation with higher probability than genomes with low fitness. During replication, local mutations and chromosomal rearrangements are performed on the genomes, at a spontaneous rate fixed by simulation parameters.

The population is initialized with a same random genome containing at least one gene. As generations go by, neutral, deleterious or beneficial mutants appear and their frequencies in the population vary depending on natural selection and genetic drift. The target function is better and better approximated and the genome structure evolves to eventually contain between tens to hundreds of genes (depending on the evolutionary conditions) scattered along the genome.

In silico experimental evolution allows for perfect recording of all mutational events that have occurred in the lineage of any organism. It is thus possible to trace the evolution of a single gene along the generations, and thus to compare genomes from different generations by identifying genes that descend one from the other.

4 Inversion Distance Estimators on Artificial Genomes

We propose 18 runs of Aevol to be used as benchmark datasets for comparative genomics studies. All estimators were computed for the 18 experiments, and we show the results for two experiments in Fig. 2. Experiments with 6 different conditions were run 3 times each, with a different seed each time. The conditions concern the allowed mutation types, among: inversions, duplications (where the copied segment is pasted anywhere on the genome), tandem duplications (where it is pasted next to the position of the ancestral segment), losses, translocations, point mutations and small InDels. Mutation rates were set to 5.10^{-6} mutation per base per generation for point mutations and InDels, and 10^{-5} for larger allowed rearrangements. All runs were stopped after 15000 generations with a genome containing approximately 100 genes. We make accessible, for each of the 18 runs, the input parameters, and for each generation, the list of genes, their coordinates on the genome, and their genealogy (how they relate to each other across generations). Material can be uploaded here: http://aevol.inrialpes.fr/resources/benchmark/cie_2016.

From Aevol output we compute signed permutations of genes without duplicates which model the relative order of genes compared with the last generation. We keep, in each generation, only the genes that have a unique descendant in the last generation, with no duplication event in its history between this generation and the last. Only the last generations can contain such genes, so permutations are only computed for a few hundred generations.

The results for two different runs out of 18 are shown in Fig. 2. The two were chosen for extreme but informative behaviors. The first run allowed for inversions, duplications and small mutations (A), while the second one allowed for translocations and tandem duplications in addition (B). At each generation we keep the genome in the ancestry lineage of the fittest genome at generation 15 000. The true number of inversions is compared with the estimated one, according to 7 estimators (we removed 3 of them because the curves were indistinguishable from others). The results highly depend on the conditions. On the (A) part, all estimators except AA are estimating a rather good number of inversions up to 50 events. On the (B) part, we cut the graph after 100 generations

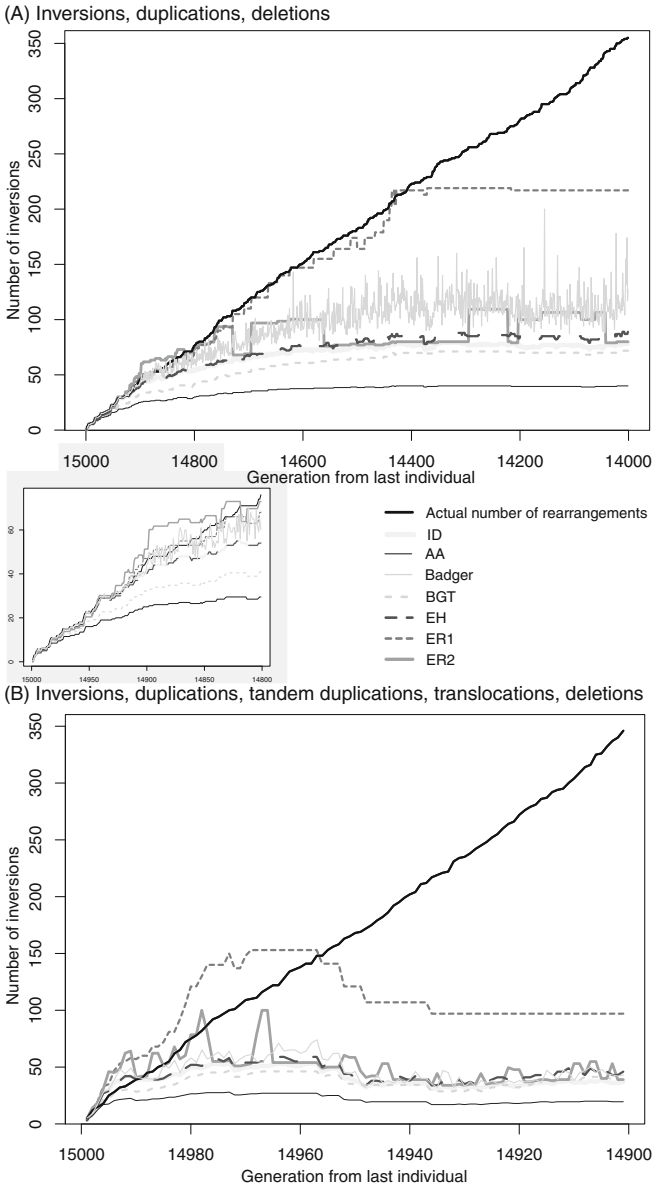

Fig. 2. The results of 7 estimators: ID, Badger, EH, BGT, ER1, ER2, AA. The other ones gave indistinguishable curves (BD from EH, LM and CL from BGT). x axis is the generation number. y axis is the number of inversions. All generations are compared with the last one, number 15000. The true number is the black solid line, and the others are estimated numbers. These Aevol runs includes (A) inversions, duplications and deletions (B) inversions, duplications, tandem duplication, translocations and deletions. The number of compared genes is from 119 to 109 (A), and from 92 to 49 (B).

because all estimators had lost the signal for a long time. This rapid signal loss is expected because of the accumulation of translocations that blur the inversion signal. On both runs, EH and BD are giving estimates which are almost equal to ID, the parsimony value. BGT, LM, CL and AA are constantly worse than the parsimony value. Only ER1 gives a better estimate than parsimony until generation 600 (after ∼200 inversions) in the first run. On ad-hoc simulators reported in the papers describing the estimators, all 10 estimators gave significantly better results than parsimony for any variation of parameters [5,7,8,12,20,22].

5 Discussion

In our experiments there are many quality differences between estimators. But a general tendency is that after a low true number of events ($\sim n/3$, where n is the number of genes), most of them significantly underestimate the true value. This highly contrasts with the claimed performances of these estimators. For example ID is supposed to have great chance of giving the right value up to $n/2$, while LM, EH, BD, BGT all have been tested on simulations and reported to give the right value far above n [5,7,12,22].

We argue that our datasets are not artefactually difficult (nor purposely *made* difficult), and that the poor results encountered here are susceptible to reflect real results on biological data. One argument for this is the better behavior of ER1 in several situations, including the one depicted on Fig. 2(A). The addition of ER1 compared to the other estimators is that it takes into account the distribution of intergene sizes. It suggests that part of the failure of the other estimators can be explained by this ignorance of intergene sizes. In biological data, intergene sizes influence probabilities of breakages, as it has been shown several times on mammals for example [6,21].

Some estimators have been tried on biological data. The inversion distance is often used. Badger has been used several times to reconstruct bacterial or mitocondrial gene orders [20], and AA has been used to estimate distances between Yeast genomes [2], and ER2 on amniote data [7]. The results have to be read in regard of this study on artificial life.

Part of the discrepancy between the true value and the estimated value remains unexplained. The complexity of the real scenarios probably blurs the signal that estimators are able to capture. But again, this complexity is not a specificity of Aevol, and is probably encountered in biological data. So by this simple experiment we can worry that none of the existing estimators of rearrangement distance would be able to produce a plausible value on real genomes.

Future Work. We tested only the estimation of the number of inversions. But only with the runs we have already computed, a lot more can be done: estimation of the proportion of translocations as in [1], or estimating both inversions and duplications as in [25]. Artificial genomes could in principle not only be used by comparative genomics inference methods, but by a larger set of molecular evolution studies. For the moment the sequences are made of 0s and 1s, which is

not a problem to study gene order, but can be disturbing for sequence analyses. This way of coding sequences is on another hand a good sign that Aevol was not developed for benchmarking purposes. In a close future nucleotidic and proteic sequences with the biological alphabet will be added to extend the benchmarking possibilities of the model.

Also we work with only one lineage, and compare only two genomes here, because Aevol evolves a single population. A useful addition will be speciation processes, in order to be able to compare several genomes.

On the Blind Multidisciplinarity. This study experiments a singular kind of interdisciplinarity. Obviously communities from comparative genomics and artificial life have to work together in order to make such results possible. But, on the opposite, these results are only possible because both communities first work in relative isolation. If they had defined their working plans together, spoke to each other too often or influenced each other's way of thinking evolutionary biology, the work would have lost some value. Indeed, what makes the difficulty here for comparative genomicists is that they have to infer histories on data for which they have no stranglehold on the processes, just as for biological data, but on which they also have the correct answer, just not as for biological data.

Acknowledgement. This work was funded by FAPESP grant 2013/25084-2 to PB, ANR-10-BINF-01-01 Ancestrome to ET and ICT FP7 European programme EVOEVO to CK and GB.

References

1. Alexeev, N., Aidagulov, R., Alekseyev, M.A.: A computational method for the rate estimation of evolutionary transpositions. In: Ortuño, F., Rojas, I. (eds.) IWBBIO 2015, Part I. LNCS, vol. 9043, pp. 471–480. Springer, Heidelberg (2015)
2. Alexeev, N., Alekseyev, M.A.: Estimation of the true evolutionary distance under the fragile breakage model. Arxiv (2015). http://arxiv.org/abs/1510.08002
3. Batut, B., Parsons, D.P., Fischer, S., Beslon, G., Knibbe, C.: In silico experimental evolution: a tool to test evolutionary scenarios. BMC Bioinformatics **14**(S15), S11 (2013)
4. Beiko, R.G., Charlebois, R.L.: A simulation test bed for hypotheses of genome evolution. Bioinformatics **23**(7), 825–831 (2007)
5. Berestycki, N., Durrett, R.: A phase transition in the random transposition random walk. Probab. Theory Relat. Fields **136**, 203–233 (2006)
6. Berthelot, C., Muffato, M., Abecassis, J., Crollius, H.R.: The 3d organization of chromatin explains evolutionary fragile genomic regions. Cell Rep. **10**(11), 1913–1924 (2015)
7. Biller, P., Guéguen, L., Tannier, E.: Moments of genome evolution by double cut-and-join. BMC Bioinform. **16**(Suppl 14), S7 (2015)
8. Biller, P., Knibbe, C., Guéguen, L., Tannier, E.: Breaking good: accounting for the diversity of fragile regions for estimating rearrangement distances. Genome Biol. Evol. (2016, in press)

9. Caprara, A., Lancia, G.: Experimental and statistical analysis of sorting by reversals. In: Sankoff, D., Nadeau, J.H. (eds.) Comparative Genomics, pp. 171–183. Springer, Amsterdam (2000)
10. Dalquen, D.A., Anisimova, M., Gonnet, G.H., Dessimoz, C.: ALF-a simulation framework for genome evolution. Mol. Biol. Evol. **29**(4), 1115–1123 (2012)
11. Duchemin, W., Daubin, V., Tannier, E.: Reconstruction of an ancestral yersinia pestis genome and comparison with an ancient sequence. BMC Genom. **16**(Suppl 10), S9 (2015)
12. Eriksen, N., Hultman, A.: Estimating the expected reversal distance after a fixed number of reversals. Adv. Appl. Math. **32**, 439–453 (2004)
13. Fertin, G., Labarre, A., Rusu, I., Tannier, E., Vialette, S.: Combinatorics of Genome Rearrangements. MIT Press, London (2009)
14. Fletcher, W., Yang, Z.: Indelible: a flexible simulator of biological sequence evolution. Mol. Biol. Evol. **26**(8), 1879–1888 (2009)
15. Hall, B.G.: Simulating DNA coding sequence evolution with EvolveAGene 3. Mol. Biol. Evol. **25**(4), 688–695 (2008)
16. Hannenhalli, S., Pevzner, P.A.: Transforming men into mice (polynomial algorithm for genomic distance problem). In: Proceedings of 36th Annual Symposium on Foundations of Computer Science (1995)
17. Hillis, D.M., Bull, J.J., White, M.E., Badgett, M.R., Molineux, I.J.: Experimental phylogenetics: generation of a known phylogeny. Science **255**(5044), 589–592 (1992)
18. Hindré, T., Knibbe, C., Beslon, G., Schneider, D.: New insights into bacterial adaptation through in vivo and in silico experimental evolution. Nat. Rev. Microbiol. **10**, 352–365 (2012)
19. Knibbe, C., Coulon, A., Mazet, O., Fayard, J.-M., Beslon, G.: A long-term evolutionary pressure on the amount of noncoding DNA. Mol. Biol. Evol. **24**(10), 2344–2353 (2007)
20. Larget, B., Simon, D.L., Kadane, J.B.: On a Bayesian approach to phylogenetic inference from animal mitochondrial genome arrangements (with discussion). J. Roy. Stat. Soc. B **64**, 681–693 (2002)
21. Lemaitre, C., Zaghloul, L., Sagot, M.-F., Gautier, C., Arneodo, A., Tannier, E., Audit, B.: Analysis of fine-scale mammalian evolutionary breakpoints provides new insight into their relation to genome organisation. BMC Genom. **10**, 335 (2009)
22. Lin, Y., Moret, M.E.: Estimating true evolutionary distances under the DCJ model. Bioinformatics **24**(13), i114–i122 (2008)
23. Mallo, D., De Oliveira Martins, L., Posada, D.: Simphy: phylogenomic simulation of gene, locus, and species trees. Syst Biol. **65**, 334–344 (2016)
24. Steel, M., Penny, D.: Parsimony, likelihood, and the role of models in molecular phylogenetics. Mol. Biol. Evol. **17**(6), 839–850 (2000)
25. Swenson, K.M., Marron, M., Earnest-DeYoung, J.V., Moret, B.M.E.: Approximating the true evolutionary distance between two genomes. J. Exp. Algorithmics **12**, 3.5 (2008)
26. Szőllösi, G.J., Boussau, B., Abby, S.S., Tannier, E., Daubin, V.: Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. Proc. Natl. Acad. Sci. U. S. A. **109**(43), 17513–17518 (2012)