Syst. Biol. 57(2):202–215, 2008

ISSN: 1063-5157 print / 1076-836X online
DOI: 10.1080/10635150802032982

Inferring Species Membership Using DNA Sequences with Back-Propagation Neural Networks

A. B. ZHANG,[1,4,*] D. S. SIKES,[2] C. MUSTER,[3] AND S. Q. LI[1,*]

[1]*Institute of Zoology, Chinese Academy of Sciences, Beijing 100080, P. R. China; E-mail: zhangab2008@yahoo.com.cn; zhangab@ioz.ac.cn*
[2]*University of Alaska Museum, 907 Yukon Drive, Fairbanks, Alaska 99775-6960, USA*
[3]*Molecular Evolution and Animal Systematics, University of Leipzig, Talstrasse 33, D-04103 Leipzig, Germany*
[4]*Current Address: Albanova University Center, Royal Institute of Biotechnology, SE-106 91 Stockholm, Sweden; E-mail: abzh@kth.se*
[*]*To whom correspondence should be sent.*

Abstract.—DNA barcoding as a method for species identification is rapidly increasing in popularity. However, there are still relatively few rigorous methodological tests of DNA barcoding. Current distance-based methods are frequently criticized for treating the nearest neighbor as the closest relative via a raw similarity score, lacking an objective set of criteria to delineate taxa, or for being incongruent with classical character-based taxonomy. Here, we propose an artificial intelligence–based approach—inferring species membership via DNA barcoding with back-propagation neural networks (named BP-based species identification)—as a new advance to the spectrum of available methods. We demonstrate the value of this approach with simulated data sets representing different levels of sequence variation under coalescent simulations with various evolutionary models, as well as with two empirical data sets of COI sequences from East Asian ground beetles (Carabidae) and Costa Rican skipper butterflies. With a 630- to 690-bp fragment of the COI gene, we identified 97.50% of 80 unknown sequences of ground beetles, 95.63%, 96.10%, and 100% of 275, 205, and 9 unknown sequences of the neotropical skipper butterfly to their correct species, respectively. Our simulation studies indicate that the success rates of species identification depend on the divergence of sequences, the length of sequences, and the number of reference sequences. Particularly in cases involving incomplete lineage sorting, this new BP-based method appears to be superior to commonly used methods for DNA-based species identification. [Back-propagation; DNA barcoding; incomplete lineage sorting; neural networks; species identification.]

DNA barcoding has attracted considerable recent attention with promises to aid in species identification and bioinventory efforts (Hebert et al., 2003a, 2003b; Ebach and Holdrege, 2005; Gregory, 2005; Marshall, 2005; Schindel and Miller, 2005; Ratnasingham and Hebert, 2007). Although still controversial (Will and Rubinoff, 2004; Prendini, 2005; Hickerson et al., 2006; Meier et al., 2006; Whitworth et al., 2007), and certainly not a replacement of traditional taxonomy, numerous potential benefits of DNA barcoding have been generally acknowledged (Savolainen et al., 2005; Ratnasingham and Hebert, 2007).

However, one major issue that needs to be resolved is how to read the organismal barcode once it is generated (DeSalle et al., 2005). Most recently published approaches to DNA barcoding have used distance measures to infer species affiliation (Hebert et al., 2003a, 2003b, 2004). These include two frequently used methods—a simple BLAST approach (Altschul et al., 1990, 1997) and a tree-based genetic distance approach (Hebert et al., 2003a, 2003b; Steinke et al., 2005). These approaches generally use a raw similarity score to produce a nearest neighbor that is not necessarily the closest relative (Koski and Golding, 2001). Furthermore, an *a priori* similarity cut-off is needed to determine species status using these methods. It remains questionable whether such universal cut-off values exist, even among congeneric species (Ferguson, 2002; Hickerson et al., 2006; Whitworth et al., 2007). Thirdly, information is inevitably lost when differences among sequences are converted into genetic distances (Steel et al., 1988). Finally, these non–character-based methods are also criticized as being incompatible with classical character-based taxonomy (DeSalle et al., 2005).

Recently, two new strategies based on a Bayesian framework and decision theory, respectively (Nielsen and Matz, 2006; Abdo and Golding, 2007), have advanced DNA barcoding practice considerably by incorporating statistical approaches that include more information available in DNA sequences. However, these two methods, in essence, are still distance-based in the way they use sequence information, although they use the information in different ways. As we have mentioned above, it has been pointed out by Steel et al. (1988) that genetic information will inevitably be lost when the difference between two sequences is converted into genetic distances, regardless of the way the genetic distance is later used. Furthermore, as pointed out by Abdo and Golding (2007), the Bayesian method as currently implemented (Nielsen and Matz, 2006) cannot handle more than two populations/species at a time and requires a two-step procedure to resolve a "species tie," thereby limiting its use in the practice of DNA barcoding. Although the decision-theory method (Abdo and Golding 2007) uses more of the information in the data than simple distance-based methods, this power comes with a computational expense; e.g., the performance deteriorates even with a small sample size of 25 (in their study they claim that this was a large sample size). Finally, both of these methods rely on some rather restrictive assumptions, such as phylogenetic hypotheses, population genetic postulates, and evolutionary models that may not always apply to real data (Nielsen and Matz, 2006; Abdo and Golding, 2007).

In this paper, we propose a new method of allocating specimens to species using DNA sequence data, based on existing back-propagation neural network methods. Artificial neural networks (ANNs) were originally developed to model the function of connected neurons in the brain (Rosenblatt, 1958) and they continue to be used in cognitive science. However, their utility as a

general computational method was realized with the development of the back-propagation method (Werbos, 1974; Rumelhart et al., 1986; Parker, 1987). Smith (1993) described neural networks and the back-propagation procedure in detail. The method is nonlinear, can represent any function to an arbitrary precision, and makes no assumptions about the frequency distributions of the data. Although each individual neuron implements its function rather slowly and imperfectly, collectively a network can perform a surprising number of tasks quite efficiently (Reilly and Cooper, 1990). This information-processing characteristic makes ANNs a powerful computational device, able to learn from examples and capable of generalizing to examples never seen before (Zhang et al., 1998). They have been applied successfully in many fields, including the prediction of financial markets, speech synthesis, handwriting recognition, and medical diagnostics. In the fields of evolutionary biology and molecular biology, artificial neural networks have been applied to DNA/RNA and protein sequence analysis (Wu, 1997; Wu and Chen, 1997) such as protein and ribosomal RNA classification (Wu and Shivakumer, 1994; Wu et al., 1995; Wang, 1998) and phylogenetic reconstruction (Dopazo and Carazo, 1997).

Below we demonstrate using a set of simulated data sets and two empirical data sets how such an artificial intelligence–based approach can be used to assign an unknown sequence to a species name. The empirical data sets include examples of different phylogenetic distances comprising sets of related species and genera (ground beetles) and a complex of closely related cryptic species (skipper butterfly).

Materials and Methods

Neural Network

Definition of a neural network.—A neural network is a parallel computational model comprised of a large number of adaptive processing units (neurons) that communicate through interconnections with variable strengths (weights), in which the learned information is stored. A multiple layer network has one or more layers of hidden neurons, which enables the learning of complex tasks by extracting progressively more meaningful features from the input patterns (Wu, 1997). Figure 1a shows a typical neural network that contains one input layer, a few hidden layers, and one output layer (Zhang et al., 1998; Zhang et al., 2002; Appendix 1). In this figure, the circles indicate input neurons and the rectangles represent neurons that are extremely simple analog computing devices. In this study, we always use three layers (described as n-h-m network); the input layer contains the values for vector $X = [x_1, x_2, \ldots, x_n]$, a hidden layer that contains h codes ($h = int(\log_2(n))$), and one output vector $O = [o_1, o_2, \ldots, o_m]$ that gives the values of output. The lines connecting the neurons represent weights that could be described by two matrices:

$$W_{(1)} = \begin{pmatrix} w_{11} & \ldots & w_{1j} \\ \vdots & \ddots & \vdots \\ w_{i1} & \cdots & w_{hn} \end{pmatrix} \quad (1)$$

and

$$W_{(2)} = \begin{pmatrix} w_{11} & \ldots & w_{1j} \\ \vdots & \ddots & \vdots \\ w_{i1} & \cdots & w_{mh} \end{pmatrix} \quad (2)$$

The following activation function,

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

was used to compute the value of a neuron. Let the activation value for neuron j be o_j. Let the weight between neuron j and neuron i be w_{ij}(1, 2). These weights are what determine the output of the neural network. Therefore, it can be said that the connection weights form the memory of the neural network. Let the net input to neuron be net_j, then

$$net_j = \sum_{i=1,k} w_{ij} o_j \quad (4)$$

where k is the number of neurons feeding into neuron j and

$$o_j = f(net_j) = \frac{1}{1 + e^{-\sum_{i=1,k} w_{ij} o_j}} \quad (5)$$

Training a network using reference sequences.—Reference sequences were digitized using the following codes: A = 0.1, T = 0.2, G = 0.3, C = 0.4 and were used to train the network (Fig. 1b). A layer's weights and biases were initialized according to the Nguyen-Widrow initialization algorithm (Nguyen and Widrow, 1990), which chooses values in order to distribute the active region of each neuron in the layer evenly across the layer's input space. Each row vector $t_i (i = 1, 2, \ldots, m$, where m is the number of species) was contained in the following diagonal matrix

$$T = \begin{pmatrix} a_{11} & \ldots & 0 \\ \vdots & a_{ii} & \vdots \\ 0 & \cdots & a_{mm} \end{pmatrix} \quad (6)$$

where a_{ii} is equal to 1, representing species i. The training process is usually as follows (Zhang et al., 1998). First, examples of the training set are entered into the input nodes. The activation values of the input nodes are weighted and accumulated at each node in the first hidden layer. The total is then transformed by an activation function into the node's activation value. It in turn becomes an input into the nodes of the next layer, until eventually the output activation values are found. The training algorithm is used to find the weights that minimize some overall error measure such as mean squared

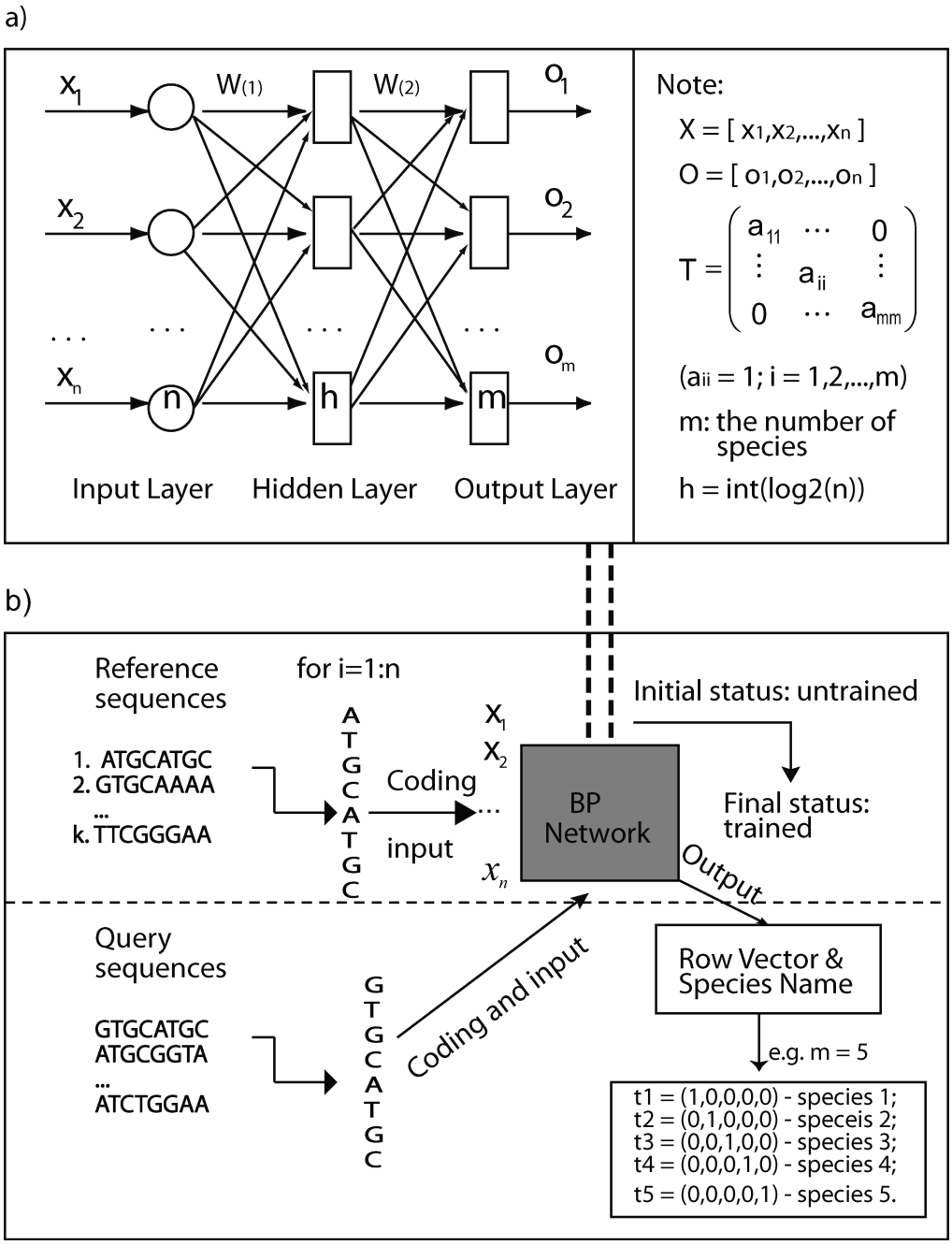
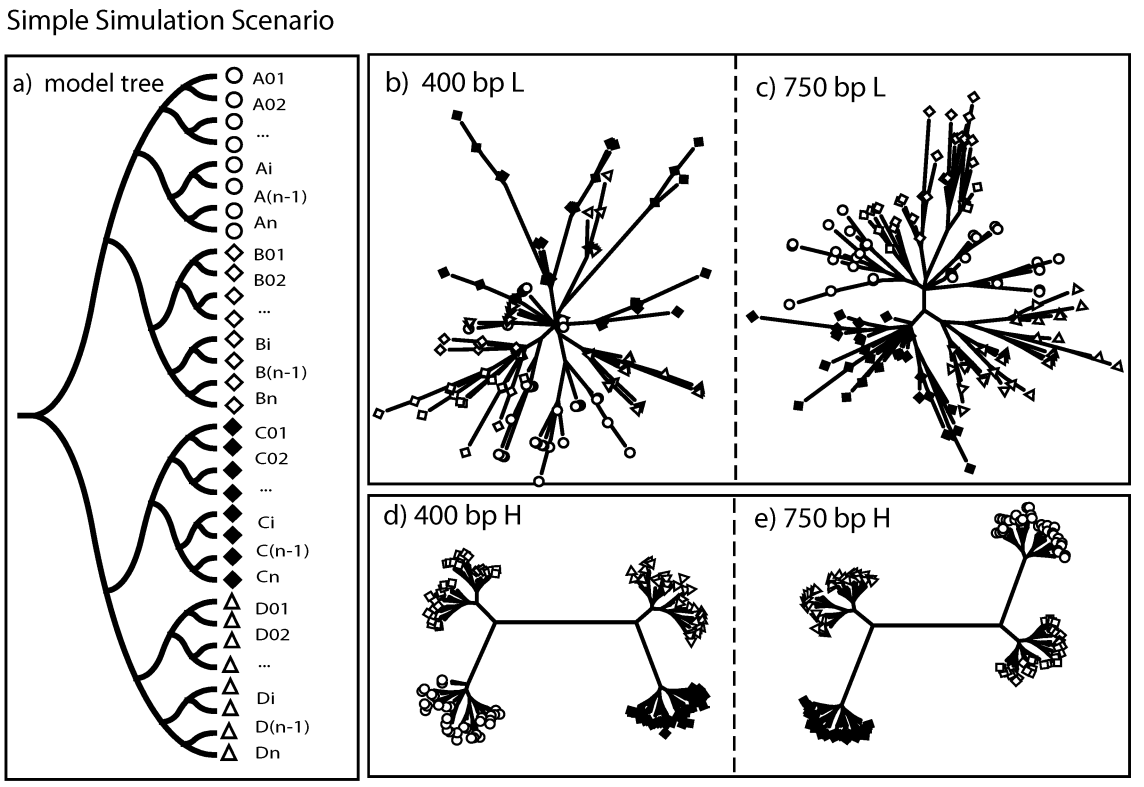


FIGURE 1. Neural network and processing scheme of sequences involved. (a) A typical neural network, including one input layer, a few hidden layers, and one output layer. In this study, we use a three-layer BP network (see text). $X = [x_1, x_2, \ldots, x_n]$ is the input layer vector, and $O = [o_1, o_2, \ldots, o_m]$ is the output layer vector. The circles or rectangles are the neurons. $W_{(1)}$, $W_{(2)}$, together with the lines connecting the neurons, represent the weights for each layer respectively (see text for the definitions). (b) Processing scheme of references sequences (training data set) and query sequences (test data set). Above the dotted line is the training data set and below are the test data set cases. The line with arrow indicates the direction of processing. The sequences were coded using the method described in the text. A set of weights and biases were obtained once a network was trained. A trained network is ready to assign a query sequence to a known species by producing a corresponding row vector. The double vertical dashed line indicates how the top graph fits into the bottom graph.

errors (MSEs). Hence the network training is actually an unconstrained nonlinear minimization problem. Before a network is trained, the weights and biases are evaluated using the Nguyen-Widrow initialization algorithm (Nguyen and Widrow, 1990). To put it simply, the training process will try to adjust the weights so that the network will generate correct target outputs for given network inputs. We used mean squared error (MSE)—the average squared error between the networks and the target outputs as a performance function. The weights and biases are updated in the direction of the negative gradient of the performance function using a technique called


Simple Simulation Scenario
a) model tree
A01
A02
...
Ai
A(n-1)
An
B01
B02
...
Bi
B(n-1)
Bn
C01
C02
...
Ci
C(n-1)
Cn
D01
D02
...
Di
D(n-1)
Dn
b) 400 bp L
c) 750 bp L
d) 400 bp H
e) 750 bp H
Coalescent Simulation Scenario
f) 648 bp
1 Ne
C
G
A
E
D
F
B
H
50,000 generations
g) 648 bp
10 Ne
B
G
A
H
E
F
C
D
500,000 generations

back-propagation (Werbos, 1974; Parker, 1982; Rumelhart and McClelland, 1986; Smith, 1993), which involves performing computations backwards through the network. To provide faster convergence and allow a network to respond not only to the local gradient but also to recent trends in the error surface, momentum has been added to back-propagation learning by making weight changes equal to the sum of a fraction of the last weight change and the new change suggested by the back-propagation rule. Briefly, back-propagation is used to calculate derivatives of performance *perf* with respect to the weight and bias variables *X*. Each variable is adjusted according to gradient descent with momentum,

$$dX = mc \times dXprev + lr \times (1 - mc) \times {}^{dperf}/_{dX} \qquad (7)$$

where $dXprev$ is the previous change to the weight or bias, *mc* is the value of momentum, and *lr* represents the learning parameters. One hundred thousand or more iterations (epochs) were used to achieve smaller values of mean square errors. For a trained network, the main parameters, weights and the bias, were saved. A value of 0.95 (the highest theoretical value is 1) for the projected vector as the BP identification score was used because higher values would need longer training times.

Identifying query sequences using a trained network.—The query sequences were coded using the method described above. Each numeral coding of each nucleotide site became one element of the input vector *X* (Fig. 1b). Then, the input vector *X* was fed into the trained network, and one output row vector *O*, corresponding to a different species following Formula 6, was obtained for each input vector *X* (see Fig. 1 for details). The aim of training a network is to let $o_1, o_2, ..., o_m$ be close to target vector T, whose sub-row-vectors, such as (1, 0, 0, 0) for a four-species example, represent species 1 (predefined). After training, the output vector of the network for one of sequences selected from species 1 could be like (0.9989, 0, 0, 0). In our study, we use 0.95 as a threshold. Higher values (than 0.95) could be used but may need longer training time (the highest theoretical value is 1). In the example above, the vector would refer to species 1, whereas species 2 would correspond to (0, 1, 0, 0). The success rate of species identification was based on the following formula:

$$Rate_{success} = \frac{Number_{hit}}{Number_{test}} \qquad (8)$$

where $Number_{hit}$ and $Number_{test}$ are the numbers of sequences successfully hit by the present method and the number of total query sequences examined, respectively.

Simulated Data Sets

We used computer simulations to investigate the power of our new approach in different situations. Firstly, using a relatively simple model of molecular evolution, we evaluated the effects of sequence length and the size of the training data set on the success rate of species identification with different methods. Secondly, we fixed the length of sequences and further evaluated the influence of the size of the training data sets, together with incomplete lineage sorting, on the success rate of species identification under coalescent simulations with more complex evolutionary models.

Simulation with simple evolutionary models.—A total of 128 sequences was generated using Monte Carlo simulation of DNA sequence evolution implemented in Seq-Gen (Rambaut and Grassly, 1997) for a model tree with four species (A, B, C, D), each including 32 individuals (Fig. 2a). We randomly chose 4, 8, 16, 24, and 28 sequences from each species to construct data sets containing 16, 32, 64, 96, and 112 reference sequences, respectively. The remaining sequences from the corresponding data set were used as query sequences.

The F84 model (Felsenstein, 1984; Yang, 1993) was used to generate the simulated data (Fig. 2a). We set the transition/transversion ratio (k) equal to 10, the gamma parameter (Γ) to 10, and the frequencies of nucleotides A, C, G, and T, g_A, g_C, g_G, and g_T, respectively, to 0.35, 0.15, 0.15, and 0.35. The L1 and L2 values, which indicate the levels of sequence divergence on the model trees, were set to represent a range of divergence levels from high to low (L2/L1 = 0.01/0.2 and L2/L1 = 0.001/0.0015, respectively), where L1 and L2 represent substitution rate per site among species and within species, respectively. Each branch length is assumed to denote the mean number of nucleotide substitutions per site that will be simulated along that branch. For each parameter combination, the topologies displayed in Fig. 2a were simulated 20 times, generating random data sets of 400 bp and 750 bp in length, respectively. The longer sequence corresponds to the standard fragment length that is used in animal barcoding (Herbert et al., 2003a, 2003b). The 400-bp fragment was used to investigate the feasibility of using shorter sequences in DNA barcoding. The resulting sets of sequences were used to generate the data sets of reference sequences and query sequences described

FIGURE 2. Simple/coalescent simulation scenario. (a–e) Model tree and neighbor-joining (NJ) trees (one example each from twenty simulated datasets) of the simulated sequences of different divergence in the simple simulation scenario. (a) Model tree, which contains four species, each including 32 individuals; (b) NJ tree of 400-bp sequence with low sequence variation (400 bp L); (c) NJ tree of 750-bp sequence with low sequence variation (750 bp L); (d) NJ tree of 400-bp sequence with high sequence variation (400 bp H); (e) NJ tree of 750-bp sequence with high sequence variation (750 bp H). The different terminal symbols on each tree correspond to the four species in (a) (f, g) Gene tree (white, inside) simulated by neutral coalescence within simulated species tree (black, outside) in the coalescent simulation scenario ($GTR + \Gamma + I$ model). (f) Example of a gene tree contained in a species tree of recent divergence (total depth of species tree =1 N_e, where $N_e = 100, 000$). (g) Example of a gene tree contained in a species tree of ancient divergence (total depth of species tree =10 N_e, where $N_e = 100, 000$). Thirty-two sequences were simulated for each species. More topologies of species trees simulated in this study can be found in online Appendix 1.

above. The success rate was calculated using Equation 8. The average success rate of 20 runs was used for comparisons.

Coalescent simulations with complex evolutionary models.—For these simulations, we took into account the possible discordance between species trees and gene trees resulting from incomplete lineage sorting (divergence time in generations less than 1 N_e), together with complex evolutionary models. All simulations were performed using Mesquite version 1.12 (Maddison and Maddison, 2006).

The simulation strategy is illustrated in Figure 2f, g. First, species trees were generated by a pure birth process using Mesquite's Uniform Speciation (Yule) module. We generated 20 species trees with different topologies (online Appendix 1; available at www.systematicbiology.org). Within each species tree, coalescent simulations were performed to generate gene trees. We then simulated sequence evolution along those gene trees to generate a set of sequence matrices using the $GTR + \Gamma + I$ model (two different settings: GTR_1 and GTR_2; see below). We fixed the length of the sequence to 648 base pairs, which is a commonly used length (Hebert et al., 2003a, 2003b), and we had already investigated the effect of sequence lengths on the success rate of species identification in the simulations above. For both GTR models, we considered deeper species trees (total depth of 10 N_e generations) and shallower species trees (depth $= 1N_e$). Parameter values used in GTR_1 ($GTR + \Gamma + I$) were derived from Roe and Sperling's (2007) study, although they could be assigned arbitrarily: base frequencies 0.35 A, 0.15 C, 0.25 G, 0.25 T; rates AC = 2, AG = 4, AT = 1.8, CG = 1.4, CT = 6, and GT = 1; gamma shape parameter was set as 0.5, and proportion of invariable sites was equal to 0.26. For GTR_2 ($GTR + \Gamma + I$), we used the following settings: base frequencies 0.32 A, 0.10 C, 0.12 G, 0.46 T; rates matrix 10.6 AC, 16.7 AG, 8.8 AT, 1.5 CG, 122.9 CT, and 1.0 GT; gamma shape parameter 0.85; and proportion of invariable sites 0.58. An effective population size (N_e) of 100,000 and a scaling factor of 3×10^{-8} were used for all simulations.

We simulated eight species, each containing 32 individuals, resulting in 256 OTUs for each sequence matrix. We selected 1, 4, 12, 24, and 28 individuals from each of eight species as training data in each sequence matrix, resulting in training data sets with 8, 32, 96, 192, and 224 sequences, respectively. The remaining sequences were used as query sequences.

To compare with commonly used approaches, we also calculated success rates using both the simple BLAST approach (Altschul et al., 1990, 1997) and a distance-based approach (Hebert et al., 2003a, 2003b; Steinke et al., 2005) in each simulated data set. We used a standalone BLAST program for Windows (BLASTN 2.2.14; Altschul et al., 1997, ftp://ftp.ncbi.nlm.nih.gov/blast/executables/LATEST-BLAST/), whose main advantage is the ability to create our own BLAST databases using reference sequences. Each query sequence was submitted and compared with the contents of the BLAST databases. The sequence producing the maximum score in the database was considered to be conspecific with the query sequence. We also calculated corrected pairwise genetic distances between each query sequence and reference sequence under the F84 or GTR models using PAUP* version 4.0b10 (Swofford, 2002). The query sequence was considered conspecific with the least distant reference sequence. The success rate of species identifications were calculated using Equation 8 as above. To study the relationship of the success rate among these methods and our BP-based method, we further performed correlation analysis among the three methods under complex simulations.

Ground Beetle Data

We examined an empirical data set taken from Zhang et al. (2005, 2006) and Zhang and Sota (2007), consisting of 159 mitochondrial COI sequences (690 bp) from nine ground beetle species that belong to two subgenera of *Carabus* (Coleoptera: Carabidae), *Leptocarabus* and *Coptolabrus* (online Appendix 2; available at www.systematicbiology.org). Six to 30 individuals of each species were sampled from different locations on the Korean peninsula and Japanese islands. The beetles were determined based on characters of external and genital morphology. We divided the sequences into two categories, reference sequences and query sequences, by randomly choosing half of the individuals from each species. This resulted in 79 reference sequences and 80 query sequences (online Appendix 2). The former were used to train a three-layer network, and the latter were fed into the trained network to output row vectors corresponding to species. The success rate of species identification was calculated using Equation 8. Additionally, as mentioned above, to examine the power of shorter sequences in species identification, we simply divided the 690-bp COI sequence into the first half and the second half, each 345 bp in length. As above, with these shorter lengths we used 79 reference sequences and 80 query sequences. Two new networks were constructed and trained, corresponding to these two data sets.

Neotropical Skipper Butterfly Data

We also used an empirical data set of the Neotropical skipper butterfly "*Astraptes fulgerator*" (Lepidoptera: Hesperiidae), which recently has been proposed to form a complex of at least 10 separate species on the basis of DNA barcoding (Hebert et al., 2004; but see Brower, 2006). Four hundred and seven mitochondrial COI sequences of *Astraptes fulgerator* were obtained from the published DNA barcoding project (Code-EPAF: http://barcodinglife.org/views/projectlist.php?&). We removed sequences that were too short or contained ambiguous characters. The remaining sequences were aligned using ClustalX version 1.83 (Chenna et al., 2003), resulting in an alignment of 630 bp (online Appendix 2). This empirical data set provides an ideal basis for comparison of our approach with other recently developed

barcoding identification strategies, because it was used in both the Nielsen and Matz (2006) and Abdo and Golding (2007) studies. Abdo and Golding (2007) have shown that their decision-theory method resulted in higher rates of correct species assignment than the Nielsen and Matz (2006) method, we therefore focused on the comparison between Abdo and Golding's and our approaches. In their simulation, Abdo and Golding took almost all available sequences as training data, and only one sequence was drawn as the query sequence from all the available sequences. To contrast against the Abdo and Golding (2007) method, we only chose one third, half, and all except for one of the sequences of each species randomly as training data (note: in this latter case, we still used fewer training data since we withheld nine sequences as query sequences). The corresponding remaining sequences were used as query sequences. Obviously, our training data sets were much smaller than theirs. As the number of available training sequences is limited in most real barcoding projects, we regard the method that requires fewer reference sequences for an equally good performance as superior.

Phylogenetic Analysis

Phylogenetic trees, under the maximum likelihood (ML) criterion, were inferred using PAUP* 4.0b10 (Swofford, 2002) and Garli v.0.951 (Zwickl, 2006); Bayesian methods were implemented using MrBayes v3.1.2 (Ronquist and Huelsenbeck, 2003). We used the $GTR + \Gamma + I$ (carabid data set) and $HKY + \Gamma + I$ (hesperiid data set) models chosen by implementation of the AIC in the program MrModelTest v2.2 (Nylander, 2004). For the carabid data set PAUP* was used to first optimize parameter values via an iterative fixation and relaxation of parameters combined with heuristic searching with TBR branch swapping. This strategy is described in Sullivan et al. (2005). Once parameter values stabilized with additional searches, we fixed them for subsequent ML bootstrapping. Bootstrapping entailed heuristic searching with TBR branch swapping on starting trees obtained by neighbor joining with a limitation of 1000 rearrangements evaluated for each of 100 searches. This was repeated and the (nearly identical) bootstrap values of the two runs were averaged for reporting on the presented trees. Due to the large size of the hesperiid data set (407 OTUs), PAUP* could not be used to perform ML bootstrapping. Instead, we used the genetic algorithm approach implemented in the program Garli v.0.951 (Zwickl, 2006), which enabled us to complete 100 pseudoreplicate ML bootstrap analyses for these 407 OTUs in 2 CPU days (on a 2.16-GHz Intel Core Duo Macintosh).

Bayesian analyses were conducted by using MrBayes' default strategy of running two simultaneous analyses, allowing for monitoring of the average standard deviation of the split frequencies to help assess when stationarity of the MCMC chains had been reached. These chains were run for 5 million steps, sampling one of every 1000 trees. This was repeated for a total of four independent runs. For the carabid data set, the average standard deviation of the split frequencies reached 0.015 by step 2.5 million, so burn-in was set at 50%, resulting in 2500 trees from each run. For the hesperiid data set, the average standard deviation of the split frequencies of the first analysis never got below 0.9, whereas this metric dropped below 0.05 for the second analysis by step 2.5 million, indicating that the runs had converged. The uncorrected potential scale reduction factor (PSRF) of Gelman and Rubin (1992), which should approach 1 as runs converge, was 1.00 for all post–burn-in parameter estimates. Examination of the trace files for these MCMC runs also showed all four analyses had reached the same parameter space. The carabid data set chains reached stationarity with nearly identical harmonic means of the marginal log-likelihoods (-4133 to -4134, combined ESS of 954). Tracer v1.3 was used to calculate the autocorrelation times (the distance separating independent samples) of each of these four runs, which were 9585 to 11,460, suggesting our sampling strategy of one tree per 1000 was oversampling by a factor of 10. The harmonic means of the marginal log-likelihoods for the hesperiid data set were also virtually identical (-2001 to -2010) and the combined ESS for all parameters was >309, indicating that sufficient independent samples had been taken to estimate the model parameters. The 50% majority-rule consensus phylogram built from the post–burn-in trees of the first two independent runs of the carabid data set and the second two runs of the hesperriid data set was used to present the inferred phylogenies.

RESULTS

Simulated Data Sets

Simple model scenario.—The network was trained using the reference sequences with 100,000 iterations (epochs) for each simulation data set. This produced a mean squared error less than 0.0001. It took 10 min for a data set of 16 sequences of 400 bp from four species to about 5 h for a data set of 224 sequences of 648 bp from eight species to train a network on a Windows PC (Intel (R), Core (TM) 2 CPU 6400, 2.13 GHz, 0.99 GB of RAM, depending on the size of data set. Once a data set was trained, it could identify thousands of test sequences within a few seconds or minutes. It's also possible to continue the training of one network by adding additional training data. This could be very useful in the DNA-barcoding practice.

All compared methods—BP-based species identification, BLAST, and distance-based approaches—can identify species with almost 100% average success rates in the case of high levels of sequence variation (interspecific divergence greatly exceeding intraspecific divergence), regardless of the length of sequences and the number of reference sequences (results not shown). In simulations with extremely low levels of sequence variation, the success rate of species identification to a large extent depends on the size of data set (number of reference sequences) and length of sequence (Fig. 3a, b). However, our method can identify species with higher success rate than traditional BLAST and distance-based

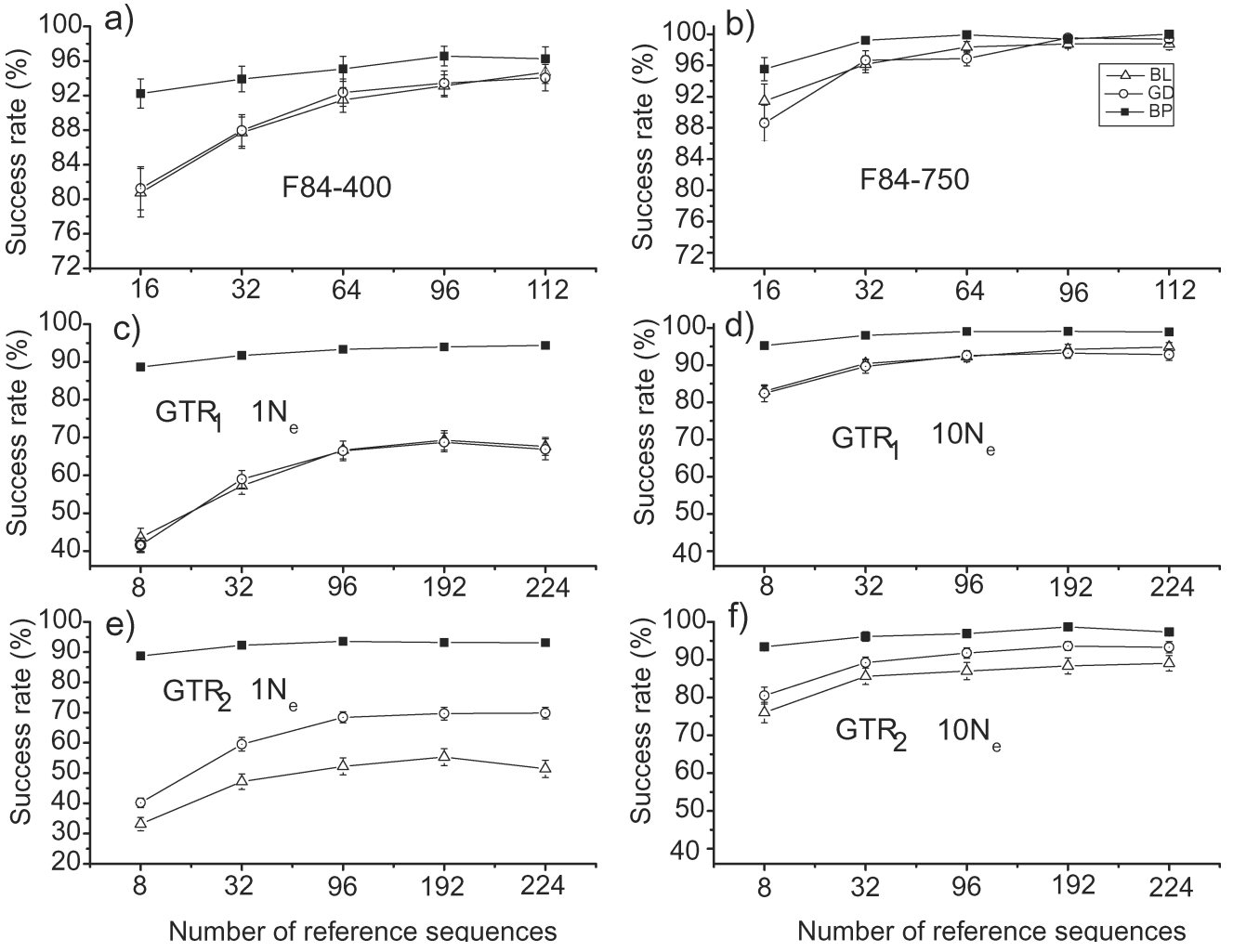


FIGURE 3. Success rates of species identifications with BLAST method, genetic distance method, and BP-based method in the simple simulation/model scenario (with low sequence divergence) and under coalescent simulation with the $GTR + \Gamma + I$ model. All above simulations were conducted with 8, 32, 96, 192, and 224 reference sequences, respectively. Detailed settings of parameters of each model can be found in the text. Triangle, circle, and solid squares indicate the success rates of BLAST method (BL), genetic distance (GD), and BP-based methods (BP), respectively. Horizontal bars below and above each symbol represent standard errors.

approaches in almost all cases of low levels of sequence variation, especially with smaller data sets. For example, for the 16-sequence data set with 750-bp length sequence, the BLAST and distance-based methods only assigned 91.43% ± 2.19% and 88.61% ± 2.31% of the query sequences to the correct species, respectively, whereas our BP-based species identification approach allowed correct identifications with a substantially higher success rate (95.54% ± 0.08%; Fig. 3b).

Generally, with low sequence variation, an overall increase in the success rate of species identification was observed with increasing reference sequence data set size for all methods; e.g., from 91.43% ± 2.19% success rate (16 sequences data set) to 98.75% ± 0.73% success rate (112 sequences database) for BLAST method, and from 95.54% ± 1.50% success rate to 100.00% ± 0.00% success rate for our BP-based method (Fig. 3b). Short sequences (400 bp) yielded much lower success rates than long sequences (750 bp), regardless of the number of reference sequences in the data set (80.76% ± 2.80% with 400 bp versus 91.43% ± 2.19% with 750 bp for the BLAST method; 92.23% ± 1.70 % with 400 bp versus 95.54% ± 1.50% with 750 bp for our BP-based method; Fig. 3a, b).

Coalescent simulations with complex models.—Figure 3c to f summarizes the simulation results of two different coalescent models (GTR_1 and GTR_2). In all cases using these more complex simulated data, the average success rate of the BP-based method was significantly greater than that of BLAST or distance-based methods (Fig. 3c to f; online Appendix 3; available at www.systematicbiology.org), especially in cases involving incomplete lineage sorting ($1N_e$). Both distance-based and BLAST methods performed poorly in situations of incomplete lineage sorting with a small number of reference sequences; e.g., BLAST and distance-based methods could only identify species

with success rates of 33.16% ± 2.19% and 40.18% ± 1.57%, respectively, when only one sequence of each species was selected as the reference sequence (Fig. 3c, e). With increasing of numbers of reference sequences, both the BP-based method and the BLAST and distance-based methods attained higher success rates (Fig. 3c to f). There is a large difference in correct species identification between deeper and shallower species trees (total depth of 10 N_e generations versus 1 N_e) for all three methods (Fig. 3c to f). All presented higher success rates with deeper internal branches than with shallower, regardless of the underlying evolutionary models and the number of reference sequences. For example, the BLAST and the genetic distance methods obtained success rates of 51.54% ± 2.84% and 69.84 ± 1.94%, respectively, under the model of GTR_2 with 224 reference sequences (shallow species trees: 1 N_e), whereas the BP-based method attained a 93.13% ± 1.29% success rate in the same situation. However, they achieved success rates of 89.06%, 93.28%, and 97.34%, respectively, with deeper species trees (deep species trees: 10 N_e). The distance-based method demonstrated slightly higher success rate of species identification than the simple BLAST approach under the model of GTR_2, although both methods identified species with lower success rates than the BP-based method (Fig. 3c to f, online Appendix 3).

Significant correlations of success rates between the BLAST and distance-based methods were found (P = 0.00128 or <0.0001), whereas no correlation was found between the BP-based method and the BLAST or genetic distance methods (P = 0.78–0.96 in all cases). This analysis indicates that the BP-based method performs species identifications in a quite different (and more successful) way than distance-based and BLAST approaches.

Empirical Data Sets

Bayesian trees from four independent runs for nine ground beetle species are presented in Figure 4a. Detailed relationships for three closely related, nonmonophyletic *Carabus* species, *C. (L.) arboreus, C. (L.) procerulus,* and *C. (L.) hiurai,* are presented in Figure 4b. Figure 5 shows Bayesian trees from four independent runs for the species *Astraptes fulgerator* for 407 OTUs.

The network was trained for the empirical data sets using the same method used for the simulated data; the connection weights and output vectors are shown in online Appendix 4 (available at www.systematicbiology.org). A total of 159 identified ground beetle specimens were used. We randomly selected 79 sequences from all of the nine ground beetle species (half of each species) as reference sequences to train a three-layer network. Among 80 query sequences, 78, 76, and 78 sequences were successfully assigned to the correct species (97.50%, 95.00%, and 97.50% success rate, respectively) with the first half (345 bp), the second half (345 bp), and the entire 690 bp of COI. The sequences not assigned to their correct species belong to two closely related species, *Carabus* (*Leptocarabus*) *arboreus* and *C. (L.) hiurai,* which may exhibit trans-species mitochondrial polymorphism (Kim et al., 2000a, 2000b; see also Fig. 4b).

For the skipper butterfly, the training data sets included 132, 202, and 398 sequences, and the corresponding sizes of query data sets were 275, 205, and 9 sequences (Fig. 5). Of these, 263, 197, and 9 sequences were successfully assigned to their correct species (95.63%, 96.10%, and 100% success rates, respectively). We have not achieved a 100% success rate in the situations of training data sets with sizes of 132 and 202, which were one third and half of the total 407 sequences, due to the low level of divergence of sequences among these putative "species." However, our method attained a success rate of 100% when 398 sequences from a total of 407 sequences (97.78%) were used as training data, whereas the decision-theory method attained the same success rate with 462 training sequences from a total of 463 sequences (99.78% of the total sequences; Abdo and Golding, 2007). Because these authors did not conduct a study on smaller training data sets, like we have done here, we cannot make a thorough comparison with their methods. With large training data sets, we found that our method achieved the same success rate (100%) as theirs.

DISCUSSION

Our results suggest that a BP approach has potential to become a powerful tool for inferring species membership via DNA sequence comparison. This artificial intelligence–based approach, which is entirely different from current distance-based approaches, does not require *a priori* cut-off to identify species. The neural network used will obtain and remember this information from the reference sequences via adjusting weights and biases of the network automatically. Our method uses more sequence information than other currently available methods, such as BLAST, simple genetic distance–based methods, the Bayesian method of Nielsen and Matz (2006), or the decision-theory method of Abdo and Golding (2007). These approaches identify species on the basis of differences between two sequences via raw scores, simple genetic distances, or genetic distances corrected by evolutionary models. In contrast, our BP approach takes into account not only differences between sequences but also the pattern of the differences; e.g., the relative position of variable sites. Our correlation analysis of success rates of species identification among the BLAST approach, the genetic distance method, and the BP-based method also indicates that the BP-based method performs species identification in a fundamentally different way from distance-based and BLAST approaches.

The second apparent advantage to our method is that it is based on fewer or almost no assumptions when making inferences, whereas almost all current methods rely on a number of more or less restrictive assumptions that may not apply to real data (Nielsen and Matz, 2006). For example, BLAST and simple distance methods assume that extreme scores or minimal genetic distances indicate close relationship between species, which does not hold

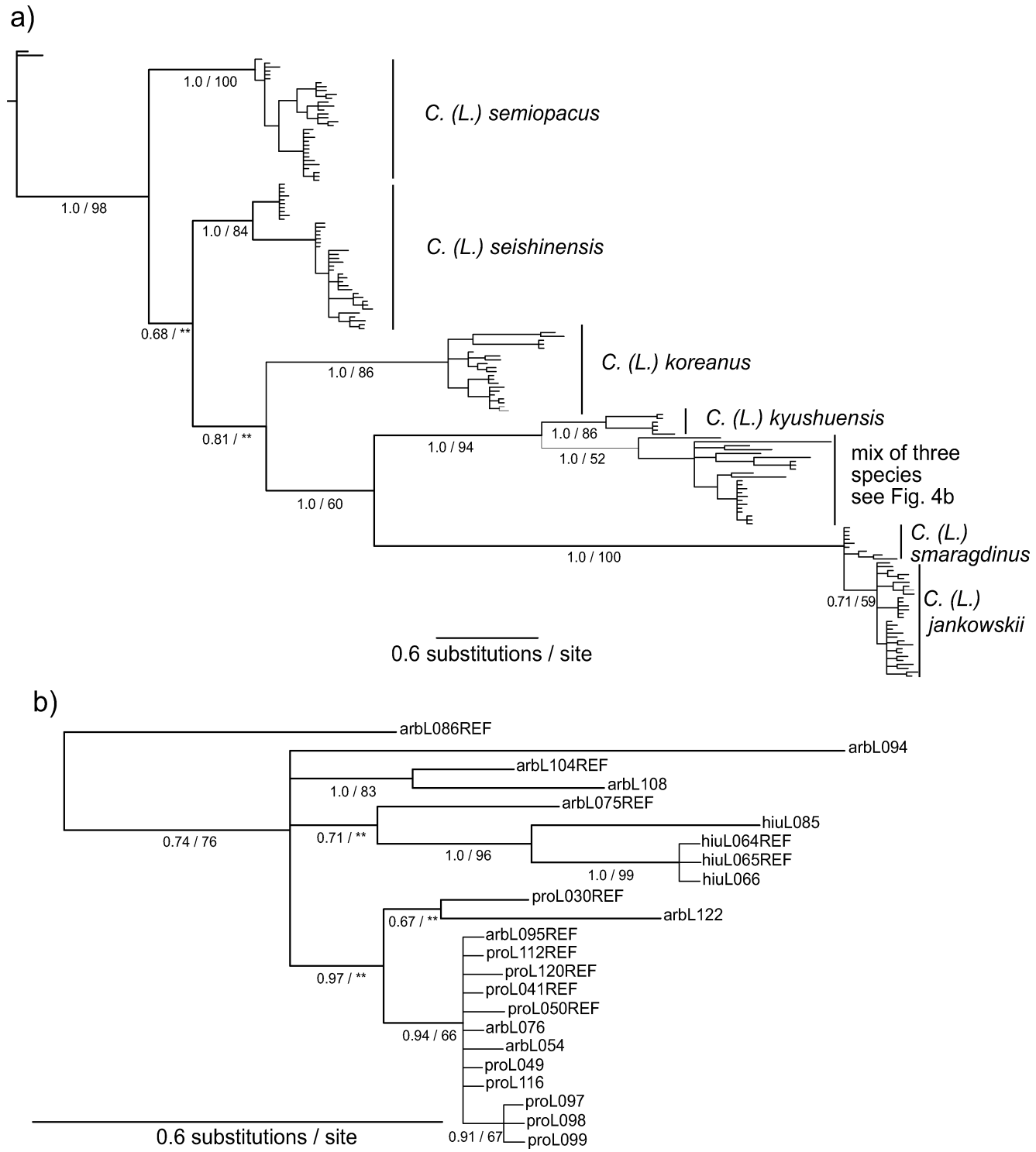


FIGURE 4. Analyses conducted using MrBayes v3.1.2 with $GTR + \Gamma + I$ model chosen by MrModelTest. Branch support values are estimated posterior probabilities on the left, maximum likelihood bootstrap proportions on the right, based on 100 pseudoreplicate heuristic searches using PAUP* with parameter values fixed. Double asterisks indicate branches not recovered in >50% of ML bootstrap searches. (a) The 50% majority-rule consensus phylogram of 5000 post–burn-in Bayesian trees from four independent runs for nine ground beetle species based on 690 base pairs of mitochondrial DNA sequences (COI). (b) Three closely related, nonmonophyletic *Carabus* species (from (a); see text for detail). Terminal codes starting with "arb," "pro," and "hiu" indicate *C. (L.) arboreus*, *C. (L.) procerulus*, and *C. (L.) hiurai*, respectively. The data matrix was listed in online Appendix 2.

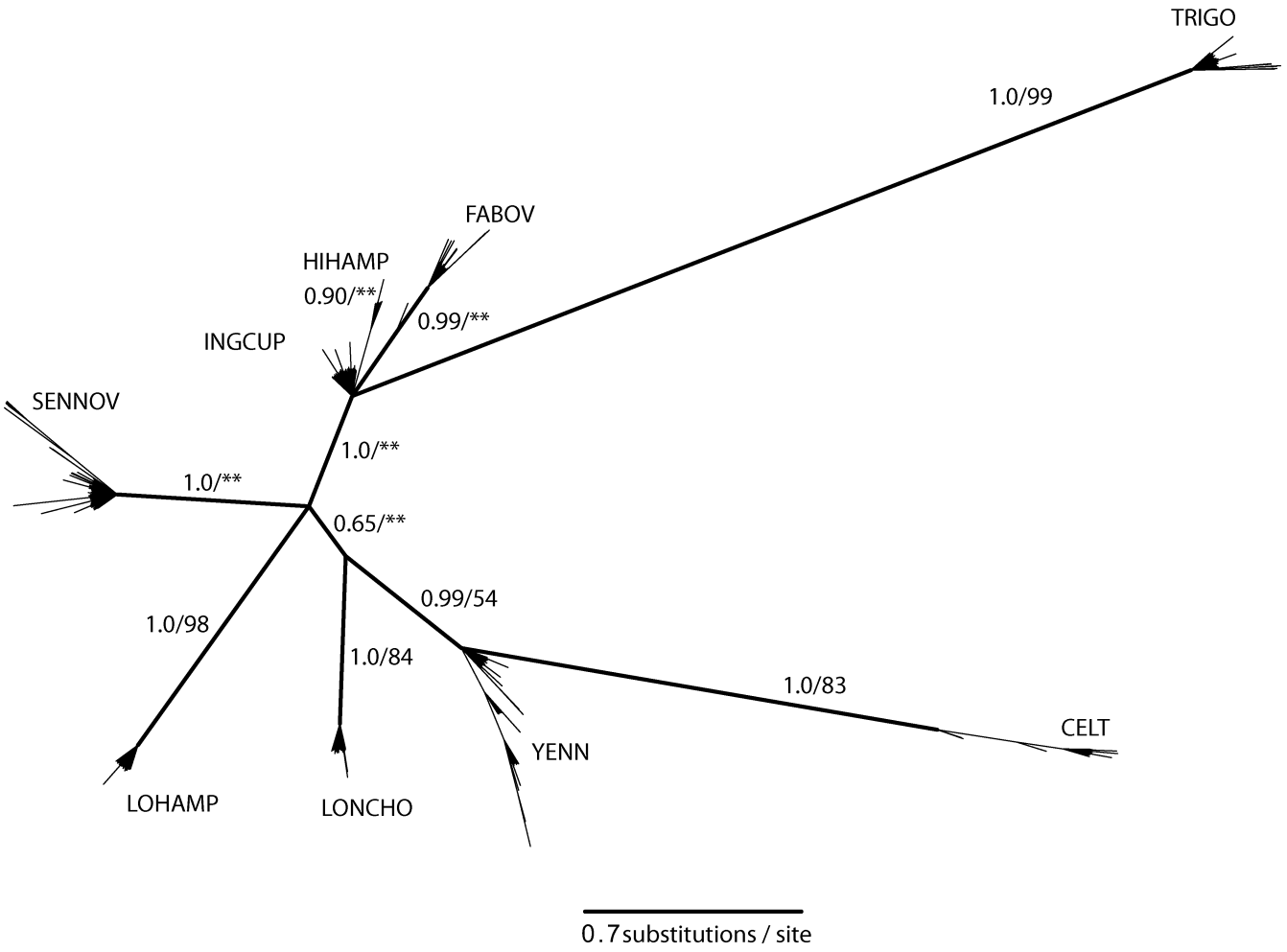


FIGURE 5. The 50% majority-rule consensus (unrooted) phylogram of 5000 post–burn-in Bayesian trees from four independent runs for the species *Astraptes fulgerator* (Lepidoptera: Hesperiidae) based on 630 base pairs of mitochondrial DNA sequences (COI) for 407 OTUs (sequences listed in online Appendix 2). Note: For clarity, only branches with greater than 0.89 posterior probability are provided with branch support values. Clade names correspond to those used in Hebert et al. (2004). Analyses conducted using MrBayes v3.1.2 with $HKY + \Gamma + I$ model chosen by MrModelTest. Branch support values are estimated posterior probabilities on the left, maximum likelihood bootstrap proportions on the right, based on 100 pseudoreplicate heuristic searches using GARLI with parameter values fixed. Double asterisks indicate branches not recovered in >50% of ML bootstrap searches.

true in the obvious case of incomplete lineage sorting. The Bayesian method of Nielsen and Matz (2006) and the decision-theory method of Abdo and Golding (2007) depend on various phylogenetic or population genetic assumptions. For example, the latter assumes an ideal panmictic population for all species or groups under study without recombination, migration, and so on, so that the evolutionary process within each group is governed by only one parameter; i.e., the number of mutational steps between two individuals within that group. Even so, both of these methods cannot estimate population genetic parameters in the case where only one sequence is known from each species. In this extreme case, the BP-based method has a clear advantage, as we have shown with simulated data (e.g., Fig. 3c, e).

Our method has the potential to use other kinds of characters easily, such as morphological characters, or even behavioral data, by simply coding them together with DNA data. This would reduce the danger of relying on a single DNA fragment for identifying and delimiting species (Roe and Sperling, 2007), although it would increase the per specimen processing cost. Our method therefore would be compatible with current taxonomic practices, and it is more appropriate for the construction of a barcode reader (DeSalle et al., 2005). The BLAST and genetic distance methods are obviously not able to incorporate nonmolecular characters, whereas the Bayesian (Nielsen and Matz, 2006) and model-based decision methods (Abdo and Golding, 2007) require extra assumptions.

We also note that our BP-based method is not without problems, although we have shown its powerful capacity in species identification compared to other currently employed methods. The first limitation of our approach is that an input sequence will always be assigned to a known species when a sequence is successfully as-

signed. This means that our BP-based method is only useful for identification purposes in samples of predefined taxa, and it is neither applicable for ambiguous cases of species identification nor for the discovery of unknown species, because this method, in essence, is governed by a process of supervised training. Second, the parameter settings that were used to train the networks, such as choosing a three-layer network, a hidden layer that contains h codes ($h = int(\log_2(n))$), a value range of 0.0001 to 0.00001 of mean squared errors, and 0.2 to 0.5 learning rates, could have been set differently. Although these settings worked well in our study, changing these parameters, theoretically, may have an effect on the training process. Presumably this will not affect our basic conclusions. We have tested some cases with different settings of parameters and found that the output vectors always tended to converge to values that corresponded to certain species, despite the values of the main parameters used to train the network. The only differences were observed in the training times. A full exploration of the training parameter space is beyond the scope of this study, because we decided that it was more important to propose this new method that resolves some problems of current methods for the ongoing DNA-barcoding practice as soon as possible. Third, we used a very simple approach of sequence encoding that seemed to perform well in our study. However, we have not examined whether our encoding method is better than other encoding methods (Brunak et al., 1991; Demeler and Zhou, 1991; Uberbacher and Mural, 1991).

Undoubtedly, using a larger DNA fragment would help to minimize the influence of nucleotide variability caused by random variation (Roe and Sperling, 2007), and larger fragments of DNA contain more information than short ones. Although both computer simulations and the real data used in this study have shown that long and short sequences differed in their success rates in identifying species, it is still difficult to address questions like "How long does a gene sequence need to be to achieve correct assignment of specimens to known species?" because the ability to identify species using DNA-barcoding methods may rely on many factors, such as the number of reference sequences, the level of sequence divergence, and patterns of DNA sequence evolution (Roe and Sperling, 2007). Therefore, we suggest that researchers should use as long fragments in species identification as possible in addition to considering the underlying variability of sequences.

Although the retention of ancestral polymorphisms (simulated cases) or possible introgressive hybridization (ground beetles data) are problematic issues in DNA barcoding (Moritz and Cicero, 2004), our simulations under coalescent models have demonstrated that the proposed artificial intelligence–based approach has more power than BLAST and distance-based methods in such a situation. Its power may be ascribed to its specific capacity of dealing with complicated nonlinear systems. However, even so, the maximal success rate with the BP-based method in our simulated cases of incomplete lineage sorting ($GTR + \Gamma + I$ model, 1 N_e) was less than 95%, whereas both the BLAST and genetic distance methods could reach a maximum success rate of less than 70% (the minimum success rate was around 40%; Fig. 3c). On the other hand, our simulations demonstrate that increasing the number of references could improve the success rates of species identification for all three methods even in such difficult situations. But, with an increasing number of reference sequences, the success rates of species identifications tended to plateau (the BLAST and genetic distance methods yielded success rates in the range of 50% to 70%, whereas rates of 93% to 94% were seen for the BP-based method). In our beetle data, there are three nonmonophyletic species, *C. (L.) arboreus*, *C. (L.) procerulus*, and *C. (L.) hiurai* (Fig. 4b). The average within- and between-species differences for these taxa overlap (not shown). Under these difficult circumstances of possible retention of ancestral polymorphisms or introgressive hybridization, it is unlikely that any sequence-based identification method would succeed for all taxa. To achieve higher success rates in such difficult cases, we suggest going beyond DNA barcoding. Standard DNA barcoding can be used to identify groups of closely related species, then longer sequences, or more loci can be used for refined species identification within this group. Phenotypic characters can also be used to solve such difficult problems. We have subsequently successfully applied four nuclear genes to this beetle group and obtained correct species identifications (Zhang and Sota, 2007). However, generalizations are not possible in the absence of more thorough studies of more empirical data. Such an inherent problem of DNA barcoding will continue to challenge systematists for some time.

To implement our approach, we have developed a new program in C++ named BPSI (BP-based Species Identification) that was used to assist this analysis (the program is freely available from zhangab2008@yahoo.com.cn).

ACKNOWLEDGMENTS

We are grateful to Dr. T. Sota, Department of Zoology, Graduate School of Science, Kyoto University, Japan, for his kind help and useful comments. We gratefully acknowledge the constructive comments of Jack Sullivan, Marshal Hedin, and two anonymous referees on an earlier version of the manuscript. This study was supported by the National Natural Sciences Foundation of China (NSFC-30340420464) and by the National Science Fund for Fostering Talents in Basic Research (Special Subjects in Animal Taxonomy, NSFC-J0630964/J0109).

REFERENCES

Abdo, Z., and G. B. Golding. 2007. A step toward barcoding life: A nodel-based, decision-theoretic method to assign genes to preexisting species groups. Syst. Biol. 56:44–56.

Altschul, S. F., W. Gish, W. Miller, E. W. Meyers, and D. J. Lipman. 1990. Basic local alignment search tool. J. Mol. Biol. 215:403–410.

Altschul, S. F., T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. Nucleic Acids Res. 25:3389–3402.

Brower, A. V. Z. 2006. Problems with DNA barcodes for species delimitation: "Ten species" of *Astraptes fulgerator* reassessed (Lepidoptera: Hesperiidae). Syst. Biodivers. 4:127–132.

Brunak, S., J. Engelbrecht, and S. Knudsen. 1991. Prediction of human mRNA donor and acceptor sites from the DNA sequence. J. Mol. Biol. 220:49–65.

Chenna, R., H. Sugawara, T. Koike, R. Lopez, T. J. Gibson, D. G. Higgins, and J. D. Thompson. 2003. Multiple sequence alignment with the Clustal series of programs. Nucleic Acids Res. 31:497–500.
Demeler, B., and G. W. Zhou. 1991. Neural network optimization for *E. coli* promoter prediction. Nucleic Acid. Res. 19:1593–1599.
DeSalle, R., M. G. Egan, and M. Siddall. 2005. The unholy trinity: Taxonomy, species delimitation and DNA barcoding. Phil. Trans. R. Soc. B 360, 1975–1980.
Dopazo, J., and J. M. Carazo. 1997. Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree. J. Mol. Evol. 44:226–233.
Ebach, M. C., and C. Holdrege. 2005. DNA barcoding is no substitute for taxonomy. Nature 434:697.
Felsenstein, J. 1984. Distance methods for inferring phylogenies—A justification. Evolution 38:16–24.
Ferguson, J. W. H. 2002. On the use of genetic divergence for identifying species. Biol. J. Linn. Soc. 75:509–516.
Gelman, A., and D. B. Rubin. 1992. Inference from iterative simulation using multiple sequences. Stat. Sci. 7:457–472.
Gregory, T. R. 2005. DNA barcoding does not compete with taxonomy. Nature 434:1067.
Hebert, P. D. N, A. Cywinska, S. L. Ball, and J. R. DeWaard. 2003a. Biological identifications through DNA barcodes. Proc. R. Soc. Lond. B. Biol. Sci. 270:313–321.
Hebert, P. D. N., E. H. Penton, J. M. Burns, D. H. Janzen, and W. Hallwachs. 2004. Ten species in one: DNA barcoding reveals cryptic species in the neotropical skipper butterfly *Astraptes fulgerator*. Proc. Natl. Acad. Sci. USA 101:14812–14817.
Hebert, P. D. N., S. Ratnasingham, and J. R. deWaard. 2003b. Barcoding animal life: Cytochrome c oxidase subunit 1 divergences among closely related species. Proc. R. Soc. B 270(Suppl.):96–99.
Hickerson, M. J., C. P. Meyer, and C. Moritz. 2006. DNA barcoding will often fail to discover new animal species over broad parameter space. Syst. Biol. 55:729–739.
Kim, C. G., O. Tominaga, Z. H. Su, and S. Osawa. 2000a. Differentiation within the genus *Leptocarabus* excl. *L. kurilensis* in the Japanese Islands as deduced from mitochondrial ND5 gene sequences Coleoptera, Carabidae. Genes Genet. Syst. 75:335–342.
Kim, C. G., H. Z. Zhou, Y. Imura, O. Tominaga, Z. H. Su, and S. Osawa. 2000b. Pattern of morphological diversification in the *Leptocarabus* ground beetles Coleoptera, Carabidae as deduced from mitochondrial ND5 gene and nuclear 28S rDNA sequences. Mol. Biol. Ecol. 17:137–145.
Koski, L. B., and G. B. Golding. 2001. The closest BLAST hit is often not the nearest neighbor. J. Mol. Evol. 52:540–542.
Maddison,W. P., and D. R. Maddison. 2006. Mesquite: A modular system for evolutionary analysis. Version 1.12. http://mesquiteproject.org.
Marshall, E. 2005. Taxonomy—Will DNA bar codes breathe life into classification? Science 307:1037.
Meier, R., K. Shiyang, G. Vaidya, and P. K. L. Ng. 2006. DNA barcoding and taxonomy in Diptera: A tale of high intraspecific variability and low identification success. Syst. Biol. 55:715–728.
Moritz, C., and Cicero, C. 2004. DNA barcoding: Promise and pitfalls. PloS Biol. 2:279–354.
Nguyen, D., and B. Widrow. 1990. Improving the learning speed of 2-layer neural network by choosing initial values of the adaptive weights. Proc. Int. Joint Conf. Neural Networks 3:21–26.
Nielsen, R., and M. Matz. 2006. Statistical approaches for DNA barcoding. Syst. Biol. 55:162–169.
Nylander, J. A. A. 2004. MrModelTest v2.2. Program distributed by the author. Evolutionary Biology Center, Uppsala University.
Parker, D. B. 1982. Learning-logic Invention Report 581-64, File 1. Office of Technology Licensing, Stanford University, Palo Alto, California.
Parker, D. B. 1987. Optimal algorithm for adaptive networks: Second order back propagation, second order direct propagation, and second order Hebbian learning. Proc. Int. Joint Conf. Neural Networks 2:593–600.
Prendini, L. 2005. Comment on "Identifying spiders through DNA barcoding." Can. J. Zool. 83:498–504.
Rambaut, A., and N. C. Grassly. 1997. Seq-Gen: An application for the Monte Carlo simulation of DNA evolution along phylogenetic trees. Comput. Appl. Biosci. 13:235–238.
Ratnasingham, S., and P. D. N. Hebert. 2007. BOLD: The Barcode of Life Data System (www.barcodinglife.org). Mol. Ecol. Notes 7:355-364.
Reilly, D. L., and L. N. Cooper. 1990. An overview of neural networks: Early models to real world systems. Pages 227–248 *in* An introduction to neural and electronic networks (S. F. Zornetzer, J. L. Davis, and C. Lau, eds.). Academic Press, New York.
Roe, A. D., and F. A. H. Sperling. 2007. Patterns of evolution of mitochondrial cytochrome coxidase I and II DNA and implications for DNA barcoding. Mol. Phyl. Evol. 44:325–345.
Ronquist, F., and J. P. Huelsenbeck. 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics 19:1572–1574.
Rosenblatt, F. 1958. The Perceptron: A probabilistic model for information storage and organization in the brain. Psychol. Rev. 65:386–408.
Rumelhart, D. E., G. E. Hinton, and R. J. Williams. 1986. Learning representations by backpropagating errors. Nature 323:533–536.
Rumelhart, D. E., and J. L. McClelland, eds. 1986. Parallel distributed processing, volumes 1 and 2. MIT Press, Cambridge, Massachusetts.
Savolainen, V., R. S. Cowan, A. P. Vogler, G. K Roderick, and R. Lane. 2005. Towards writing the encyclopaedia of life: An introduction to DNA barcoding. Phil. Trans. R. Soc. B 360:1805–1811.
Schindel, D. E., and S. E. Miller. 2005. DNA barcoding a useful tool for taxonomists. Nature 435:17.
Smith, M. 1993. Neural networks for statistical modeling. Van Nostrand Reinhold, New York.
Steel, M. A., M. D. Hendy, and D. Penny. 1988. Loss of information in genetic distances. Nature 336:118.
Steinke, D., M. Vences, W. Salzburger, and A. Meyer. 2005. TaxI—A software for DNA barcoding using distance methods. Phil. Trans. R. Soc. B 360:1975–1980.
Sullivan, J., Z. Abdo, P. Joyce, and D.L. Swofford. 2005. Evaluating the performance of a successive-approximations approach to parameter optimization in maximum-likelihood phylogeny estimation. Mol. Biol. Evol. 22:1386–1392.
Swofford, D. L. 2002. PAUP*: Phylogenetic analysis using parisimony (*and other methods). Version 4. Sinauer Associates, Sunderland, Massachusetts.
Uberbacher, E. C., and R. J. Mural. 1991. Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. Proc. Natl. Aacad. Sci. USA 88:11261–11265.
Wang, H. C., J. Dopazo, L. G. de la Fraga, Y. P. Zhu, and J. M. Carazo. 1998. Self-organizing tree-growing network for the classification of protein sequences. Protein Sci. 7:2613–2622.
Werbos, P. J. 1974. Beyond regression: New tools for prediction and analysis in the behavioral sciences. PhD thesis, Harvard University, Cambridge, Massachusetts.
Whitworth, T. L., R. D. Dawson, H. Magalon, and E. Baudry. 2007. DNA barcoding cannot reliably identify species of the blowfly genus *Protocalliphora* (Diptera: Calliphoridae). Proc. R. Soc. B 274:1731–1739.
Will, K. W., and D. Rubinoff. 2004. Myth of the molecule: DNA barcodes for species cannot replace morphology for identification and classification. Cladistics 20:47–55.
Wu, C. H. 1997. Artificial neural networks for molecular sequence analysis. Computers Chem. 40:237–256.
Wu, C., and H. Chen. 1997. Counter-propagation neural networks for molecular sequences classification: Supervised LVQ and dynamic node allocation. Appl. Intel. 7:27–38.
Wu, C., and S. Shivakumar. 1994. Back-progragation and counter-propagation neural networks for phylogenetic classification of ribosomal RNA. Nucleic Acids Res. 22:4291–4299.
Wu, C., S. Shivkumar, H. Lin, S. Veldurti, and Y. Bhatikar. 1995. Neural networks for molecular sequence classification. Math. Comput. Simu. 40:23–33.
Yang, Z. H. 1993. Maximum-likelihood-estimation of phylogeny from DNA-sequences when substitution rates differ over sites. Mol. Biol. Evol. 10:1396–1401.
Zhang, A. B., K. Kubota, Y. Takami, J. L. Kim, J. K. Kim, and T. Sota. 2005. Species status and phylogeography of two closely related *Coptolabrus* species Coleoptera, Carabidae in South Korea inferred from mitochondrial and nuclear genes. Mol. Ecol. 14:3823–3841.
Zhang, A. B., K. Kubota, Y. Takami, J. L. Kim, J. K. Kim, and T. Sota. 2006. Comparative phylogeography of three *Leptocarabus* ground beetle species in South Korea based on mitochondrial COI and nuclear 28S rRNA Genes. Zool. Sci. 23:745–754.

Zhang, A. B., and T. Sota. 2007. Nuclear gene sequences resolve species phylogeny and mitochondrial introgression in *Leptocarabus* beetles showing trans-species polymorphisms. Mol. Phyl. Evol. 45: 534–546.

Zhang, A. B., Z. J. Wang, and D. M. Li. 2002. Application of BP model and LOGIT model to prediction of occurrence of forest insect pest. Acta Ecol. Sin. 21:2159–2165.

Zhang, G. Q., B. E. Patuwo, and M. Y. Hu. 1998. Forecasting with artificial neural networks: The state of the art. Int. J. Forecast. 14:35–62.

Zwickl, D. J. 2006. Genetic algorithm approaches for the phylogenetic analysis of large biological sequence datasets under the maximum likelihood criterion. PhD dissertation, The University of Texas at Austin. www.bio.utexas.edu/faculty/antisense/garli/Garli.html.

First submitted 4 February 2007; reviews returned 3 May 2007; final acceptance 11 January 2008

Associate Editor: Marshal Hedin

APPENDIX 1

General definitions of terms related to BPNN.

Activation function/transfer function: The general function used to compute the value of a neuron can be any differentiable function, such as logistic function.

Back propagation (BP): A supervised learning technique used for training artificial neural networks. It was first described by Werbos (1974) and further developed by Rumelhart et al. (1986).

Convergence: The approach towards the target vector (a fixed state of the output) via adjustments to the weights of the network as the training of the network proceeds.

Epoch/iteration: An epoch consists of a few steps during the training of a network—take the values of input vector, find the values of nodes for the hidden and output layers, adjust the weights of the output and hidden layers according to the target vector. The whole process is repeated many times so that the output vector becomes closer to target vector.

Error surface: A (k+1)-dimensional surface representing the error terms of a model depending on k parameters. The coordinates of the error surface consist of the k parameters of the model function and the error term. The error surface can be used to find the best fit of a model by finding the minimum of the error surface.

Initialization algorithm: A technique to initialize a layer's weights and biases before a network is trained; e.g., with minor random weights and biases.

Layer: A common style in which the neurons in a network are arranged. A typical BP network contains an input layer, one or more hidden layers and an output layer. Each layer consists of a certain number of neurons depending on problems being solved.

Momentum/momentum factor: A numerical value incorporated into the BP algorithm by making weight changes equal to the sum of a fraction of the last weight change and the new change suggested by the back-propagation rule, which avoids getting trapped in a local minimum in the error surface during the training process.

Neuron/node: A model of a neural cell in animals and humans in a NN context, an extremely simple analog computing device, which can take values from one or more neurons and output to other neurons via an activation or transfer function.