Supporting information

Table S1: The nematode collagens identified in this work from species in the genus *Caenorhabditis*.

Species	# of collagen genes
C. angaria	127
C. brenneri	209
C. briggsae	173
C. elegans	181
C. japonica	197
C. remanei	145
C. sinica	163
C. tropicalis	155

Article S1. The MIPhy model of gene family evolution

Here we present the MIPhy reconciliation and clustering algorithm. It proceeds in two phases, the first inferring the gene events and using them to generate an initial clustering, and the second phase incorporating traditional data clustering techniques to refine the clusters. The model of gene family evolution is derived from the core reconciliation methods of NOTUNG, with key modifications. That algorithm only allows incongruence (in the form of incomplete lineage sorting) at polytomies. Incongruence may appear due to errors in sequencing/gene-finding, incompletely resolved branches in tree-building software, horizontal gene transfer (HGT), or it may be due to selective pressures acting on one or more species. As such we freely include these events in our reconciliation. Moreover, we assume that incongruence is more likely than a duplication event followed by several independent loss events, so the latter case is not considered as possible history. As an example, node n6 in Fig. S1 is an incongruence event because genes a1 and b1 are closer to gene d1 than to c1, whereas the species tree suggests that a1 and b1 should be closer to gene c1 than to d1. If we did not allow incongruence events, n6 would instead be classified as a duplication, and the reconciliation would require three additional loss events.

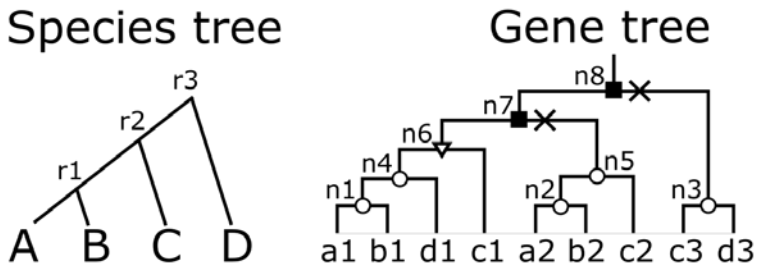

Figure S1: Example species and gene trees. In the gene tree the gene events are indicated with filled squares, open circles, open triangles, and Xs, representing duplication, speciation, incongruence, and loss events, respectively. Nodes a1 and a2 in the gene tree represent two distinct genes from species A, b1 and b2 are genes from species B, and so on.

It should be noted that this, and many other parsimony algorithms, define a duplication to be the presence of at least one gene from the same species in both children of some tree node. This is not sufficient to rigorously prove that a duplication has taken place in some ancestral species, but this definition has been found to perform well in practice. Another difference compared to NOTUNG is that MIPhy does not attempt to model HGT explicitly. Instead, these events will be classified as incongruence or duplications, which both contribute to the phylogenetic instability cost function. This also allows the algorithm to classify these gene events using purely local information in a single pass through T_G, decreasing the time complexity by orders of magnitude.

Terms and definitions

Given a gene tree T_G, let g represent some node, where l and r are its children. If g is a terminal leaf, its originating species $ori(g)$ is defined to be the species in the species tree T_S from which the sequence g was collected. If g is an internal node, $ori(g)$ is defined to be the most recent common ancestor in T_S of $ori(l)$ and $ori(r)$. The lineage of a node $lin(g)$ is the set of species nodes (including ancestral species) tracing $ori(g)$ back to the root of T_S. The set of all terminal leaves in the subtree of T_G rooted by g is given by $lvs(g)$. The species represented in the subtree of T_G rooted at g, $spc(g)$, is the set obtained by applying $ori(c)$ to every leaf c in $lvs(g)$; S_G is the set of species in T_S with at least one gene in T_G. The represented species of a node g not present in the represented species of a node h is given by $miss(g,h) = spc(g) - spc(h)$.

One of three mutually exclusive gene events must take place at each internal node in T_G: duplication, speciation, or incongruence. These are quantified by the binary variables $E_D(g)$, $E_S(g)$, and $E_I(g)$, respectively, constrained such that $E_D(g) + E_S(g) + E_I(g) = 1$.

Event inference

If T_S and T_G are given by the species and gene trees in Fig. S1, the gene events taking place at every internal node are inferred as follows:

- $E_D(g) = 1\ if\ spc(l) \cap spc(r) \neq \emptyset$: Node g is a duplication event if its children share any represented species. As an example, $E_D(n7) = 1$ because $spc(\mathrm{n6}) = \{A, B, C, D\}$ and $spc(n5) = \{A, B, C\}$, so $spc(n6) \cap spc(n5) = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$.
- $E_S(g) = 1\ if\ E_D(g) = 0\ and\ ori(l) \notin lin(r) \wedge ori(r) \notin lin(l)$: Node g is a speciation event if it is not a duplication event, and the originating species of neither child is contained in the lineage of the other. As an example, $E_S(n4) = 1$ because $E_D(n4) = 0$, $ori(n1) \notin lin(d1)$ $(r1 \notin \{D, r3\})$, and $ori(d1) \notin lin(n1)$ $(D \notin \{r1, r2, r3\}$.
- $E_I(g) = 1\ if\ E_S(g) + E_I(g) = 0$: More explicitly, node g is an incongruence event if it is not a duplication event, and the originating species of one child is contained in the lineage of the other. As an example, $E_I(n6) = 1$ because $E_D(n6) = 0$ and $ori(n4) \in lin(c1)$ $(r3 \in \{C, r2, r3\})$.

Minimum instability groups (MIGs) are defined by the most recent common ancestor in T_G of the leaves in that group, and the numbers of duplication and incongruence events counted in the MIG defined by node g are found by the recursive equations:

$$D(g) = E_D(g) + D(l) + D(r), \tag{1}$$

and

$$I(g) = E_I(g) + I(l) + I(r). \tag{2}$$

A speciation event indicates that genes from one species (or ancestral species) will be found exclusively in the descendants of one child and not the other. Conversely, for both children of a duplication event node there should be one gene from every species that has not yet been excluded by a previous speciation or incongruence event. Loss events are therefore counted at duplication nodes, as the number of represented species of each child not present in the other:

$$L'(g) = E_D(g) \cdot loss(g) + L'(l) + L'(r),$$

$$\text{where } loss(g) = |miss(l, r)| + |miss(r, l)|.$$

This would only be accurate if every species is represented by at least one gene in the total species of each MIG. To complete this concept, we introduce a new quantity $M(g)$ that compares the represented species under g with the total represented species S_G. Thus, the total loss events counted in the descendants of some node g would be given by:

$$L(g) = L'(g) + M(g), \tag{3}$$

$$\text{where } M(g) = |S_G - spc(g)|.$$

If T_G is from Fig. S1, $L(n5) = L'(n5) + M(n5) = 0 + 1 = 1$ because no genes from species D are present in $leaves(n5)$, while $L(n7) = L'(n7) + M(n7) = 1 + 0 = 1$. As demonstrated

here, the M term does not propagate up the tree, and tends to disappear as the algorithm progresses further from the leaves.

The above equations are somewhat naïve, as they do not allow for loss in ancestral species. If T_S is given by Fig. S1 and we consider the MIG rooted by node n3, the above equations would calculate that two loss events have occurred, once each for species A and B. However, a more parsimonious explanation is that the ancestral homolog was only lost once, in species r1, the ancestor of A and B. We therefore redefine the constituents of equation (3) to account for these processes:

$$L'(g) = E_D(g) \cdot loss(g) + L'(l) + L'(r),$$

$$\text{where } loss(g) = \max_{s \in spc(l)} |\{ori(s,t) | t \in miss(r,l)\}| + \max_{s \in spc(r)} |\{ori(s,t) | t \in miss(l,r)\}|,$$

$$\text{and } M(g) = \max_{s \in spc(g)} \left|\{ori(s,t) | t \in \{S_G - spc(g)\}\}\right|.$$

An example from Fig. S1:

$$M(n3) = \max(|\{ori(C,A), ori(C,B)\}|, |\{ori(D,A), ori(D,B)\}|)$$

$$M(n3) = \max(|\{r2\}|, |\{r3\}|) = \max(1,1) = 1$$

$$\therefore L(n3) = L'(n3) + M(n3) = 0 + 1 = 1$$

Initial clustering

This 'Initial clustering' section is also described in the main text, but is reproduced here for ease of reading. For a node g, $migs(g)$ is a set of sets describing the most parsimonious clustering pattern for those sequences in $lvs(g)$, where each inner set describes one MIG. These groups are built iteratively by comparing $sep(g)$, the score if the existing clustering patterns $migs(l)$ and $migs(r)$ are kept intact, with $cmb(g)$, the score if all descendants are combined into a single MIG; the minimum of these two values is stored as $best(g)$. After the initial clustering phase, the overall clustering pattern is described by $migs(root)$; every sequence from T_G is contained exactly once in $migs(root)$. Worked examples of these variables can be found in Table S2.

The weighted sum of equations (1), (2), and (3) constitutes the score function:

$$score(g) = \theta_D \cdot D(g) + \theta_I \cdot I(g) + \theta_L \cdot L(g) + \theta_P \cdot P(g), \quad (4)$$

where the θ values are the strictly positive weights applied to each event, and $P(g)$ is a the "relative spread" metric defined by equation (5) in the next section; for this initial phase of the algorithm it is set to 0. Each node g in T_G is visited in a post-order depth-first traversal. The algorithm is described by the following pseudocode:

If g is a terminal node:

$best(g) = M(g)$

$migs(g) = \{\{g\}\}$

Otherwise:

$cmb(g) = score(g)$

$sep(g) = best(l) + best(r)$

If $cmb(g) \leq sep(g)$, all descendants of g are merged into one MIG:

$best(g) = cmb(g)$

$migs(g) = \{lvs(g)\}$

Otherwise the existing cluster patterns of nodes l and r are kept intact:

$best(g) = sep(g)$

$migs(g) = migs(l) \cup migs(r)$

Transforming phylogenetic distances to coordinate points with multi-dimensional scaling

The second phase of the MIPhy algorithm evaluates and refines the clusters generated by in the initial phase. Several metrics exist to measure the spread between points in a cluster compared to the rest of the data. However, many require that these points be embedded into a coordinate system, such as Euclidian space, having properties such as the concept of a mean. A phylogenetic tree does not possess these properties, so we use multi-dimensional scaling to transform the nodes of the tree into a set of coordinate points that respect the phylogenetic distances between each sequence.

First, the full pairwise distance matrix from T_G is generated as the matrix D, such that D_{ij} is the phylogenetic distance (measured as the sum of the branch lengths) between the leaves i and j. The Gram matrix M can then be generated by:

$$M_{ij} = \frac{D_{i1}^2 + D_{1j}^2 - D_{ij}^2}{2}$$

where 'sequence 1' is an arbitrary choice held constant throughout the calculation of the matrix (this sequence will be located at the origin of the coordinate system). The coordinate points can then be found by eigenvalue decomposition. If $M = USU^T$ is solved, the ith row of the matrix $X = U\sqrt{S}$ contains the coordinates for the point representing leaf i from T_G.

Cluster refinement

These coordinate points are used in the "relative spread" calculation:

$$P(g) = \frac{\sigma(g)}{\bar{\sigma}} - 1, \quad (5)$$

where $\sigma(g)$ is the standard deviation of the points representing the sequences in the MIG rooted by g, and $\bar{\sigma}$ is the median standard deviation of all MIGs (excluding singleton clusters). The spread quantity is normalized around 0, so $P(g) = 1.0$ indicates that the spread of MIG g is 100% larger than the median spread, while $P(h) = -0.3$ indicates that the spread of MIG h is 30% smaller than $\bar{\sigma}$. Though MIPhy currently measures spread using a simple standard deviation, clustering-specific methods like the Davies-Bouldin index or silhouette could be easily substituted. As in the initial clustering phase, each node g in T_G is again visited in turn. The clustering procedure is repeated, this time including the relative spread term in the full score function in equation (4).

Table S2: Worked MIPhy example. This table provides the variables used in the Event inference and Initial clustering phases of the MIPhy algorithm applied to Fig. S1. Only one set of terminal leaves is included for brevity. The $best(g)$ value for each node isn't explicitly stated, but is indicated by the bolded score value in either $cmb(g)$ or $sep(g)$. The values θ_D, θ_I, and θ_L, indicate the weight of one duplication, incongruence, or loss event, respectively. The final clustering pattern is found at the root, $migs(n8)$, and here the algorithm predicts three clusters. Interestingly, the clustering pattern for this tree is invariant for all parameter weights such that $\theta_I < 3\theta_L$. This is because in the entire table, the $cmb(n6)$ to $sep(n6)$ comparison is the only one that is not invariant; as an example, $cmb(n5) < sep(n5)$ $(\theta_L < 4\theta_L)$ is true for all strictly positive weights – which is true of these parameter weights by definition – as is $cmb(n7) > sep(n7)$ $(\theta_D + \theta_I + \theta_L > \theta_I + \theta_L)$.

	Event inference values					Initial clustering values		
g	$ori(g)$	$lin(g)$	$lvs(g)$	$spc(g)$	Event	$cmb(g)$	$sep(g)$	$migs(g)$
a1	A	$\{A, r1, r2, r3\}$	$\{a1\}$	$\{A\}$	-	-	$\mathbf{3\theta_L}$	$\{\{a1\}\}$
b1	B	$\{B, r1, r2, r3\}$	$\{b1\}$	$\{B\}$	-	-	$\mathbf{3\theta_L}$	$\{\{b1\}\}$
c1	C	$\{C, r2, r3\}$	$\{c1\}$	$\{C\}$	-	-	$\mathbf{2\theta_L}$	$\{\{c1\}\}$
d1	D	$\{D, r3\}$	$\{d1\}$	$\{D\}$	-	-	$\boldsymbol{\theta_L}$	$\{\{d1\}\}$
n1	r1	$\{r1, r2, r3\}$	$\{a1, b1\}$	$\{A, B\}$	E_S	$\boldsymbol{\theta_L}$	$6\theta_L$	$\{\{a1, b1\}\}$
n2	r1	$\{r1, r2, r3\}$	$\{a2, b2\}$	$\{A, B\}$	E_S	$\mathbf{2\theta_L}$	$6\theta_L$	$\{\{a2, b2\}\}$
n3	r3	$\{r3\}$	$\{c3, d3\}$	$\{C, D\}$	E_S	$\boldsymbol{\theta_L}$	$3\theta_L$	$\{\{c3, d3\}\}$
n4	r3	$\{r3\}$	$\{a1, b1, d1\}$	$\{A, B, D\}$	E_S	$\boldsymbol{\theta_L}$	$3\theta_L$	$\{\{a1, b1, d1\}\}$
n5	r2	$\{r2, r3\}$	$\{a2, b2, c2\}$	$\{A, B, C\}$	E_S	$\boldsymbol{\theta_L}$	$4\theta_L$	$\{\{a2, b2, c2\}\}$
n6	r3	$\{r3\}$	$\{a1, b1,$ $c1, d1\}$	$\{A, B, C, D\}$	E_I	$\boldsymbol{\theta_I}$	$3\theta_L$	$\{\{a1, b1, c1, d1\}\}$
n7	r3	$\{r3\}$	$\{a1, b1,$ $c1, d1,$ $a2, b2, c2\}$	$\{A, B, C, D\}$	E_D	$\theta_D + \theta_I + \theta_L$	$\boldsymbol{\theta_I + \theta_L}$	$\{\{a1, b1, c1, d1\},$ $\{a2, b2, c2\}\}$
n8	r3	$\{r3\}$	$\{a1, b1, c1,$ $d1, a2, b2,$ $c2, c3, d3\}$	$\{A, B, C, D\}$	E_D	$2\theta_D + \theta_I + 2\theta_L$	$\boldsymbol{\theta_I + 2\theta_L}$	$\{\{a1, b1, c1, d1\},$ $\{a2, b2, c2\},$ $\{c3, d3\}\}$

175