nature communications

Article

https://doi.org/10.1038/s41467-025-59077-7

Probabilistic alignment of multiple networks

Received: 17 May 2024

Accepted: 10 April 2025

Published online: 27 April 2025

Check for updates

Teresa Lázaro[1], Roger Guimerà[1,2] ✉ & Marta Sales-Pardo[1] ✉

The network alignment problem appears in many areas of science and involves finding the optimal mapping between nodes in two or more networks, so as to identify corresponding entities across networks. We propose a probabilistic approach to the problem of network alignment, as well as the corresponding inference algorithms. Unlike heuristic approaches, our approach is transparent in that all model assumptions are explicit; therefore, it is susceptible of being extended and fine tuned by incorporating contextual information that is relevant to a given alignment problem. Also in contrast to current approaches, our method does not yield a single alignment, but rather the whole posterior distribution over alignments. We show that using the whole posterior leads to correct matching of nodes, even in situations where the single most plausible alignment mismatches them. Our approach opens the door to a whole new family of network alignment algorithms, and to their application to problems for which existing methods are perhaps inappropriate.

The problem of network alignment (also called graph matching) is ubiquitous across fields of science. In its basic formulation, the problem consists in finding the mapping of identities of nodes between two networks such that the structure of the networks is maximally preserved. For instance, in chemistry the interest is to elucidate structural similarity across molecules[1]; in bioinformatics, it is to annotate proteins by comparing protein-protein interaction networks between pairs of organisms[2–4]; in computer vision, to find equivalences between deformable objects[5–8]; in neuroscience, to align functional connectomes with the goal of identifying pathological changes or inter-species differences[2,9,10], and, more recently, to align neuron-to-neuron brain connectomes so as to identify variability across individuals and aid neuron annotation[11–13]. In other areas such as in computational social science, the problem definition has been extended to find matching between nodes across networks that are not necessarily topologically correlated to identify common roles of words in different knowledge graphs[14,15], and similarly behaving actors across different social platforms[16–18].

With few exceptions, a general assumption behind the network alignment problem is that networks are *alignable*, that is, that the networks share structural or topological similarities that can help in the alignment process. To quantify such similarities, several structural metrics, both local and global, have been proposed, and are optimized in the course of the network alignment process[2,6,7,19–23]. Perhaps the most notable and popular approach for global alignment consists in formulating the problem as a quadratic assignment problem (QAP), an NP-hard problem for which approaches to obtain good quality alignments for thousands of nodes already exist[11–13,19,21,23]. However, while powerful, approaches such as QAP and those closely related suffer from a number of caveats. First, they are heuristic and do not explicitly lay out the modeling assumptions on which they rest. Second, it is often hard to incorporate into them contextually relevant information, such as known classifications of nodes into groups (although recent kernel formulations of the QAP problem that allow for the incorporation of node and edge attributes[23]). Recently, in computer vision and in the analysis of protein interaction networks, a new generation of machine-learning approaches that use node embeddings to solve the network alignment problem have been proposed[6–8,20,24]. These approaches rely on contextual information that can typically be obtained from images or protein sequence and gene-expression data, making them unsuitable for situations in which only network topology and a few node attributes are available. Additionally, a caveat of most of the aforementioned approaches is that they are designed to align only pairs of networks.

The latter is an important shortcoming of network alignment approaches in many of the contexts previously outlined, and especially in biology, in which we typically have different observations we may want to align. For instance, being able to compare functional connectomes across multiple species can give important clues about how evolution has shaped brain functionality[10]. In this sense, the rapidly

[1]Department of Chemical Engineering, Universitat Rovira i Virgili, Tarragona, Catalonia, Spain. [2]ICREA, Barcelona, Catalonia, Spain.
✉e-mail: roger.guimera@urv.cat; marta.sales@urv.cat

increasing amount of data on physical, neuron-to-neuron connectomes[25–28] will necessitate a new generation of tools to provide reliable annotation, as well as intra and inter-species comparisons of connectomes, which hopefully can be linked to differences in behavior. There are a few approaches that address this shortcoming by relying on node attributes and a set of know identity mappings across networks[4,17,18], by extending QAP approaches and building some sort of consensus between pairwise alignments[29,30], and, more recently by using a kernel formulation of the QAP problem[23]. Many of these approaches, however, are also problematic mainly because of the lack of transparency of their assumptions, which hinders the interpretability of their results, and also because they are either heavily dependent on non-topological information or not able to incorporate non-topological information at all.

To cover this gap, we propose a probabilistic approach to the problem of multiple network alignment when little contextual information about the nodes is available, as well as the corresponding inference algorithms. Our approach assumes the existence of an underlying network blueprint from which observed networks are generated by noisily copying edges. This approach allows us to naturally consider multiple networks by recasting the network alignment problem as a problem of finding the identity assignment of each node in each observed network to a node in the blueprint. Importantly, our probabilistic approach enables us to consider an ensemble of alignments and their corresponding blueprints as opposed to considering a single, best alignment (which is the goal of heuristic approaches). This turns out to be a crucial property of our approach, since the best alignment typically does not recover the known ground truth even for small levels of noise in the observed networks. By contrast, considering an ensemble of plausible alignments often leads to recovery of the known ground truth alignment. Finally, our approach enables us to easily incorporate information such as group labels of nodes to guide the alignment sampling process and, at the same time, allows to infer missing group labels of nodes. This is an important problem in the annotation of biological networks in which the goal is often not to find the precise identity of a node, but to identify how this node can be classified into preexisting, biologically meaningful categories[27]. Our results for two real cases in two different domains (neuroscience and computational social science) show that ours is a general approach that opens the door to the development of powerful context-dependent methods for network alignment and network annotation.

Results

Probabilistic formulation of the problem of aligning multiple networks

Consider K network observations, with N nodes each and adjacency matrices $\{\mathbf{A}^k;\ k=1,\ldots,K\}$. We consider networks that are directed and with binary edges (that is, we just consider the presence or absence of connection between two nodes). Our hypothesis is that the observed networks are topologically similar, which allows us to map each node in a network to another node in each of the other networks. The goal is then to find the most plausible mapping of nodes across networks.

We follow a probabilistic approach and surmise that there is a latent underlying blueprint $\mathbf{L}$, such that each network observation has been generated from that blueprint. Then, the network alignment problem becomes a problem of finding the blueprint $\mathbf{L}$, and the permutations $\{\pi^k,\ k=1,\ldots,K\}$ that map nodes in each network to nodes in the blueprint. This formulation naturally allows the simultaneous alignment of several networks onto a single blueprint (Fig. 1).

Formally, we consider a blueprint $\mathbf{L}$ with binary edges ($L_{ij} \in \{0,1\}$) from which networks are copied with errors. Each entry A_{ij} in an observed network is assumed to be independently generated from L_{ij} with a copying error probability that depends on L_{ij}. Edges in the blueprint ($L_{ij}=1$) are copied with error probability q, whereas non-edges ($L_{ij}=0$) are copied with error probability p (see Supplementary Material for the model with uniform copy error probability $q=p$). This approach shares some features with the approach in Ref. 22, where the goal is to align a noisy network copy with the original network (the blueprint in our case) using a similar probabilistic framework. The crucial difference lies in that, in our case, the blueprint is unknown, so that we need to infer it. This difference is precisely what makes our approach symmetric (the alignment of two networks does not require that we choose one as reference) and what allows us to align multiple networks at the same time without the need to choose an arbitrary blueprint from the set of observed networks.

For simplicity, we consider first the case in which we have a single observation. Given L_{ij}, q, and p, the probability that we observe an edge ($A_{ij}=1$) or a non-edge ($A_{ij}=0$) is then:

$$p(A_{ij}=1|L_{ij},q,p)=p^{(1-L_{ij})}(1-q)^{L_{ij}} \tag{1}$$

$$p(A_{ij}=0|L_{ij},q,p)=q^{L_{ij}}(1-p)^{(1-L_{ij})}. \tag{2}$$

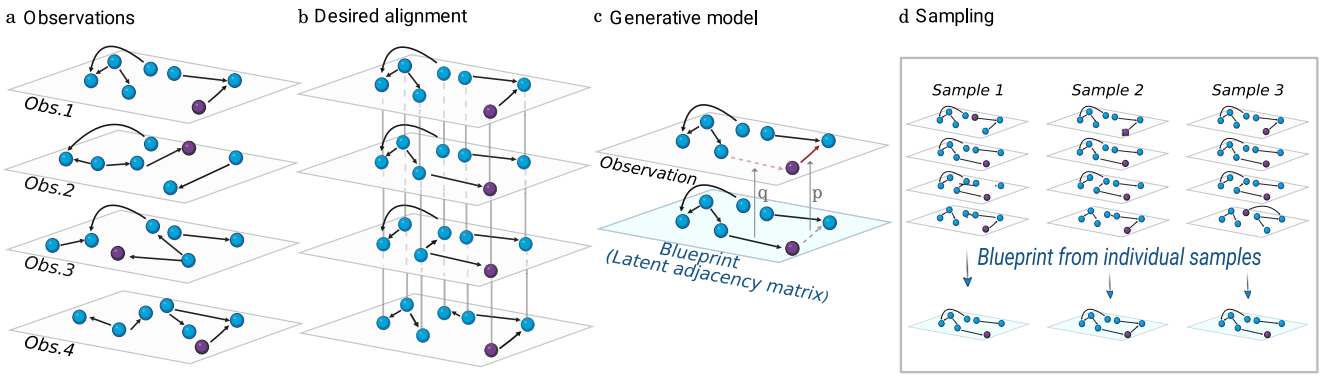

Fig. 1 | Probabilistic approach to multiple network alignment. a Our data consists of observations of topologically similar networks with unknown node identities. **b** Our objective is to align the networks, that is, to find the permutation of node identities (illustrated as a rearrangement in the plane) in each network, such that each node is mapped to its counterpart in the other networks, and edges are as similar as possible across networks. To ease visual tracking, we color in purple a node that has the same hidden identity in all networks. **c** Generative model. We assume there exists an underlying blueprint from which observations are generated by copying edges and non-edges, with copying error probabilities q and p, respectively. **d** Our probabilistic approach allows us not only to formulate the network alignment problem in terms of finding the most plausible alignment, but also to sample over the space of possible alignments (that is, permutations of node identities) and corresponding blueprints. This allows us to assign a probability to each individual node mapping, and turns out to be critical to recover ground truth alignments in noisy networks (see text and Fig. 2).

Since the probability of observing each edge is conditionally independent on the others, the total likelihood of observing an adjacency matrix $\mathbf{A}$ given a latent blueprint $\mathbf{L}$, q, and p factorizes. Additionally, note that the likelihood depends on the unknown mapping of nodes in the observed network into nodes of the blueprint, which is given by the permutation π of the nodes in the observed network. With this, the likelihood is

$$p(\mathbf{A}|\mathbf{L},q,p,\pi)=\prod_{ij}p(A_{ij}|\mathbf{L},q,p,\pi)=q^{o_{10}}p^{o_{01}}(1-q)^{o_{11}}(1-p)^{o_{00}}, \quad (3)$$

where the edge *overlaps* between the blueprint and the observations are $o_{XY}=\sum\delta_{L_{\pi(i)\pi(j)},X}\delta_{A_{ij},Y}$. For example, o_{01} is the number of entries that are 0 in $\mathbf{L}$ and 1 in $\mathbf{A}$, for mapping π.

To obtain the posterior distribution over all model parameters, we use Bayes rule

$$p(\mathbf{L},\pi,q,p|\mathbf{A})=\frac{p(\mathbf{A}|\mathbf{L},q,p,\pi)\,p(\mathbf{L},q,p,\pi)}{p(\mathbf{A})}, \quad (4)$$

where $p(\mathbf{L}, q, p, \pi)$ is the prior over model parameters. To obtain our desired posterior $p(\mathbf{L}, \pi|\mathbf{A})$, we consider a beta prior over p and q, and integrate over these variables, which yields ("Methods")

$$p(\mathbf{L},\pi|\mathbf{A})\propto\frac{\Gamma(o_{11}+\beta_q)\Gamma\left(o_{10}+\alpha_q\right)}{\Gamma\left(n_1+\alpha_q+\beta_q\right)}\frac{\Gamma(o_{00}+\beta_p)\Gamma\left(o_{01}+\alpha_p\right)}{\Gamma\left(n_0+\alpha_p+\beta_p\right)}, \quad (5)$$

where $n_1=\sum_i\delta_{L_{ij},1}$, $n_0=\sum_i\delta_{L_{ij},0}$ are the number of edges and non-edges in the latent blueprint $\mathbf{L}$, respectively, and (α_q, β_q) and (α_p, β_p) are the hyper-parameters of the prior distributions for q and p, respectively (Methods). Note that one expects $\beta_{q/p}>\alpha_{q/p}\geq 1$ so that priors favor small copy error probabilities, although the effect of the prior becomes negligible for even modest edge overlaps.

In the case in which our observation comprises K networks, Eq. (5) generalizes to ("Methods")

$$p(\mathbf{L},\{\pi^k\}|\{\mathbf{A}^k\})\propto\frac{\Gamma(O_{11}+\beta_q)\Gamma\left(O_{10}+\alpha_q\right)}{\Gamma\left(Kn_1+\alpha_q+\beta_q\right)}\frac{\Gamma(O_{00}+\beta_p)\Gamma\left(O_{01}+\alpha_p\right)}{\Gamma\left(Kn_0+\alpha_p+\beta_p\right)}, \quad (6)$$

where $O_{XY}=\sum_k o^k_{XY}$, so that the posterior depends on the overall overlap of edges and non-edges of all the networks with the blueprint. Note that because $\beta>\alpha$, the $(\mathbf{L}, \{\pi^k\})$ that maximize the posterior are the ones that maximize O_{11} and O_{00}, and minimize O_{10} and O_{01}. Also note that for a fixed permutation choice $\{\pi^k\}$, the $\mathbf{L}^\star$ that maximizes the posterior is such that $L^\star_{ij}$ is equal to the majority of all $A^k_{\tilde\pi^k(i),\tilde\pi^k(j)}$, where $\tilde\pi^k(i)$ is the inverse of $\pi^k(i)$, so that $\pi^k(\tilde\pi^k(i))=i$ ("Methods").

Note that, in the limit $p\to 0$ or, equivalently, $\beta_p\to\infty$, our generative model is equivalent to generating networks from the blueprint by just sampling on the edges in the blueprint (that is $L_{ij}=1$). The limits of recovery of node identities and group labels of these nodes in terms of the sampling probability have been extensively studied for correlated pairs of random graphs[31–33] and for correlated pairs of graphs with group structure[34]. In this probabilistic framework, the best alignment corresponds to the blueprint-permutation pair that maximizes the posterior probability $p(\mathbf{L}, \{\pi^k\}|\{\mathbf{A}^k\})$. In the limit $p\to 0$, the posterior reduces to $p(\mathbf{L}, \{\pi^k\}|\{\mathbf{A}^k\})\propto\Gamma(O_{11}+\beta_q)\Gamma(O_{10}+\alpha_q)$, so that the most plausible alignment corresponds to the blueprint-permutation pair that maximizes O_{11}, that is, the total number of edges in the graph that overlap with the blueprint. Formally, for the alignment of a pair of networks with binary adjacency matrices $(\mathbf{A}, \mathbf{B})$, if we assume that one of the two networks is equal to the blueprint as in Ref. 22, then maximizing O_{11} is equivalent to maximizing the product $\sum_{ij}A_{ij}B_{\pi(i)\pi(j)}$ as in the Koopmans-Beckmann QAP formulation of the graph matching problem[11], and, in general, to minimizing the Frobenius norm between two adjacency matrices with binary entries. When $p\neq 0$, our approach considers the contribution of both aligned edges and aligned non-edges. Therefore, the most plausible alignment in our case does not necessarily coincide with that in the Koopmans-Beckmann QAP formulation of the network alignment problem.

This observation also highlights the fact that current structural pairwise approaches cannot be easily modified to allow for the alignment of multiple networks, since, in order to do that, we would have to decide which of the networks is the generating blueprint against which all other networks need to be aligned. Unfortunately, if all copies are subject to the same copy error, there is no reason to expect that one of the observations is closer to the blueprint than another a priori. In fact, even if we had that prior information, pairwise approaches would only consider independent alignments of each network against the selected blueprint, so that the information coming from the overlap between all other pairs of networks other than the blueprint would be lost, and we would have to resort to adding a method to assess consistency of pair alignments[29,30]. Our probabilistic approach naturally circumvents these issues.

Sampling the space of plausible alignments

By analogy to statistical mechanics, we can associate an "energy" $\mathcal{H}=-\log\left(p(\mathbf{L},\{\pi^k\},\{\mathbf{A}^k\})\right)$ to each blueprint-permutation pair $(\mathbf{L}, \{\pi^k\})$, so that their posterior is written

$$p(\mathbf{L},\{\pi^k\}|\{\mathbf{A}^k\})=\frac{\exp(-\mathcal{H})}{p(\{\mathbf{A}^k\})}, \quad (7)$$

and the energy $\mathcal{H}$ can be obtained directly from Eq. (6). In this interpretation, the best alignment is the blueprint-permutation pair that minimizes $\mathcal{H}$. Importantly, this equivalence enables us to sample over the space of alignments $(\mathbf{L}, \{\pi^k\})$ using Markov chain Monte Carlo[35] ("Methods"; see Supplementary Material and Fig. fig:time_size for the algorithmic complexity of our tool). The sampling of equilibrium alignments from this space allows us to approach the network alignment problem in ways other than just finding the single best alignment.

In particular, by sampling alignments from the posterior distribution, we can estimate the probability $p(\pi^k(i_k)=i_{\mathbf{L}}|\{\mathbf{A}^k\})$ that node i_k in network A^k is mapped to node $i_{\mathbf{L}}$ in the blueprint as the fraction of sampled permutations in which $\pi^k(i_k)=i_{\mathbf{L}}$. Then, we can estimate the most probable mapping for each node $\pi^{k,\star}(i_k)$ as the one that maximizes $p(\pi(i_k)=i_{\mathbf{L}}|\{\mathbf{A}^k\})$ (Methods). As we show below, in noisy observations, the most plausible alignment does not necessarily recover the ground truth mapping of nodes. This is because for nodes with few connections, copy error can introduce a degree of ambiguity and degeneration in the mappings of node identities. However, we find that the most probable mapping of *each individual node* recovers the ground truth for their mapping more reliably (Fig. 2). This is, in fact, an expected result from using a probabilistic approach: averages over the ensemble of possible alignments are more accurate at predicting hidden information (in this case, node mappings to the blueprint) than single point estimates (that is, the best alignment) (see for instance Ref. 36 for a discussion in the context of link prediction).

Additionally, MCMC sampling allows us to leverage relevant information about the nodes. For instance, if we have access to node attributes such as group labels, we can constrain permutations to those in which only nodes with the same group labels can be mapped to the same node in the blueprint (Methods); or if we have information about the precise identity of some of the nodes (often called 'anchors' or 'seeds'[12]; see Methods for details), our algorithm can be forced to only sample permutations that have a fixed mapping for these nodes.

https://doi.org/10.1038/s41467-025-59077-7

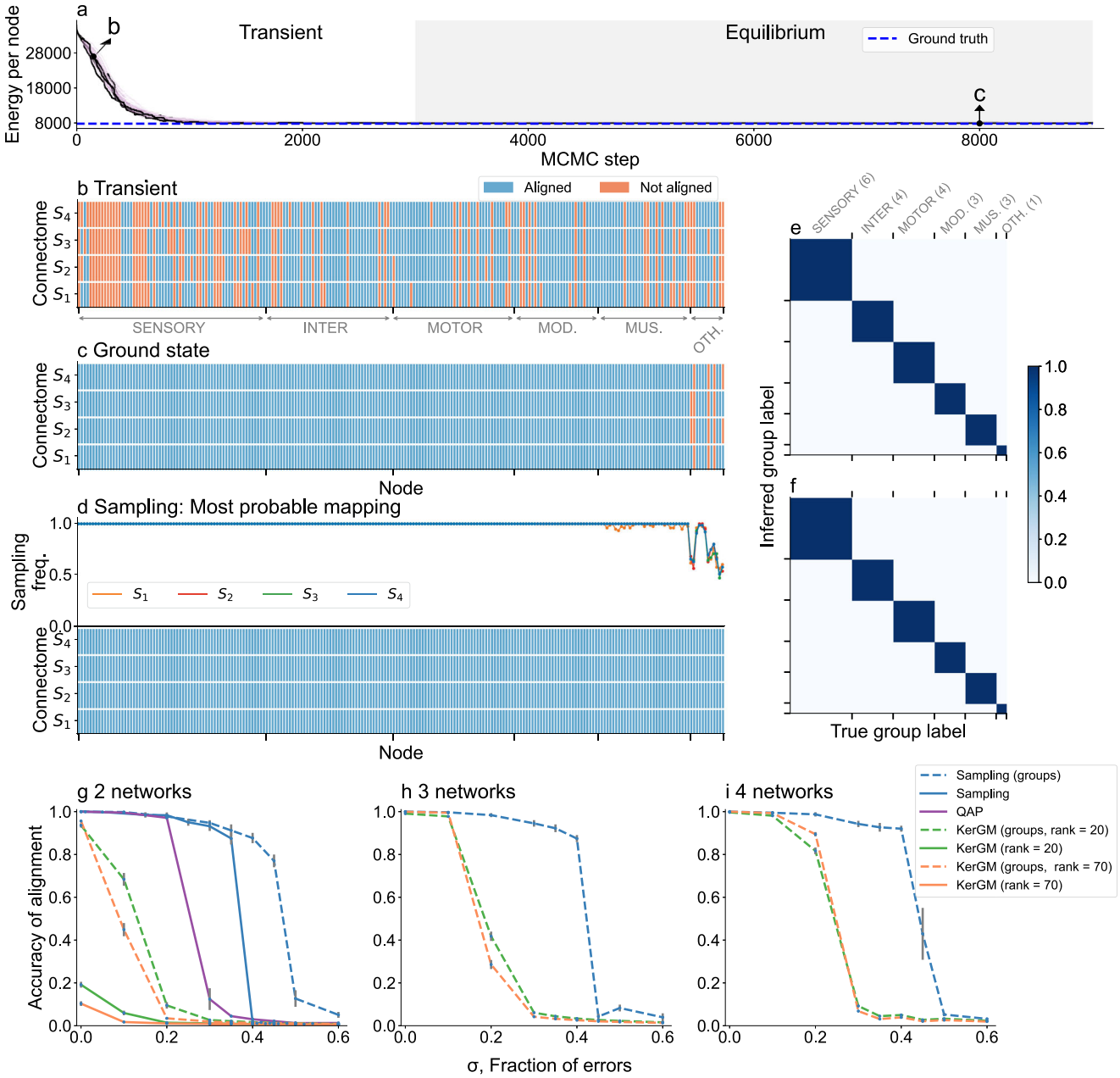

Fig. 2 | Alignment of synthetic connectomes. We consider *C. elegans* connectome A2 in Ref. 25 as a blueprint to generate noisy synthetic connectomes. **a**–**d** Alignment of 4 synthetic connectomes using group labels (neuron type) of each node. **a** Energy $\mathcal{H}$ as a function of time (Methods). Each line corresponds to a replica running at a different temperature. We sample alignments in the equilibrium zone (gray) at temperature $T = 1$ (black line). The blue dashed line shows the energy of the ground truth alignment. **b**–**d**. Alignment of individual nodes for each synthetic connectome S_1, ..., S_4, ordered by neuron type. Blue nodes are correctly aligned, i.e., mapped to their ground truth identity (blue); red if misaligned ("Methods"). **b** Example of alignment in the transient regime—point *b* in (**a**). **c**. alignment with minimum energy (ground state)—point *c* in (**a**, **d**). Most probable mapping for each node sampled from the posterior distribution at equilibrium (text and Methods). The top panel shows the frequency with which each node (in each connectome) is assigned to its most probable mapping. **e**, **f** Node label inference. We assume that neuron types are unknown for some nodes in networks S_2, S_3, and S_4 (total number for each type in parenthesis). We estimate the probability that a node has label X as the fraction of the sampled alignments in which that node is assigned that label (text and Methods). Each matrix element (X, Y) shows the probability that a node with unknown label Y is assigned label X. **e** Unlabeled nodes are the same across the synthetic connectomes S_2–S_4. **f** Unlabeled nodes are chosen at random in each of the connectome networks $S_2 - S_4$. See supplementary Figs. S5, S6 for their corresponding alignments. **g**–**i** Accuracy ar recovering the ground truth alignment for K noisy connectome copies with different fractions of errors, for three alignment methods: Fast QAP (QAP)[38], multi-way (KerGM)[23], and our sampling method (Sampling). **g** $K = 2$; **h** $K = 3$; **i** $K = 4$. Points show the mean accuracy for 10 sets of networks; error bars show the standard error of the mean.

Validation of the probabilistic approach on synthetic networks

We start by validating our approach on synthetic, but realistic, settings of increasing difficulty. Because network alignment is particularly relevant in the context of connectome analysis, we consider the connectome of the nematode *C. elegans*[25,37] as our benchmark, and perform two initial sets of experiments: (i) the alignment of multiple identical networks in which node identities have been shuffled; (ii) the alignment of several noisy copies of the same network.

In the first experiment, we use multiple copies of the connectome network provided in ref. 11 (279 neurons), with node identities

randomly shuffled in each copy. Because the networks are identical, the ground state of the energy in Eq. (7) coincides with the ground truth alignment, that is, with the desired mapping of node identities. However, the ground state is degenerate because any permutation of the nodes in the latent blueprint leaves the energy invariant. To break this symmetry, which greatly complicates the search in the space of possible alignments, we find that it is more efficient to first align two networks (chosen at random), and then add the remaining networks one by one to eventually obtain the desired global alignment of all the networks (Supplementary Fig. S2). We find that this approach enables us to perfectly align as many networks as desired, always converging to the ground state.

Except for the fact that we consider an arbitrary number of networks, as opposed to just two, the previous experiment is the standard validation test in the literature on network alignment and graph matching. In the second experiment, we move to a more realistic and challenging scenario. In particular, we use the adult A2 connectome of *C. elega ns* from ref. 25 as a blueprint (224 neurons and 2,186 connections), and generate four noisy synthetic connectomes by swapping 327 edges with non-edges from the original adjacency matrix.

Using a maximum likelihood estimate, this process would be equivalent to generating connectomes using our copying mechanism with $q \approx 0.15$ and $p \approx 0.007$.

Since the four observations are not identical, now the global ground state may not correspond to that of the ground truth. Additionally, if due to observational noise there are neurons with few connections, new degeneracies or quasi-degeneracies may appear. Therefore, the energy landscape is likely to become rougher in this realistic scenario. Indeed, we find that, whereas alignment is still possible and most nodes are correctly matched, the MCMC finds alignments with lower energy than the ground truth, indicating that the exact ground truth has become undiscoverable. (Since in this experiment the synthetic networks are generated by the generative model underlying our approach, our approach is Bayes optimal and no other method would, in general, be able to identify the ground truth alignment in this case.)

Given the added difficulty of this scenario, we take advantage of the fact that network datasets often contain useful information about nodes, such as group labels; in the specific case of connectomes, one could typically have information about neuron types (depending on their neuroblast lineage). Therefore, we can restrict our Markov chain to permutations where each neuron is mapped to a neuron of the same type in the blueprint. This drastically reduces the space of possible alignment permutations, and allows us to sample the alignment space more thoroughly and efficiently (Fig. 2). (Alternatively, we can apply the same successive alignment strategy as before, wherein we start by aligning two networks add layers successively; Supplementary Fig. S3.) Our results confirm that there exist multiple alignment permutations (and the corresponding blueprints) that have energies lower than that of the ground truth alignment (Fig. 2a, c).

This observation is key for real-world scenarios, and shows that the ground-state alignment should not be taken as the best estimate for the identity of individual nodes. Rather, as previously discussed, for individual nodes one should rather use the mapping $\pi^{k,\star}(i_k)$ that maximizes the probability $p(\pi^k(i_k) = i_{\mathbf{L}}|\{\mathbf{A}^k\})$ that node i_k in network A^k is mapped to node $i_{\mathbf{L}}$ in the blueprint (Methods). As we show in Fig. 2d, the most probable mapping for each node is more accurate at recovering the ground truth node identities. In fact, we find that some neurons are easy to align, so that the majority of sampled alignments have the same node mappings for those neurons; but there are other neurons, typically with few connections (see Supplementary Fig. S4), for which there are several possible alignments and only through averaging can we recover the underlying node identity.

To benchmark the performance of our approach, we compare it to other methods for network alignment: the Fast QAP[11,38] and the multi-way Kernel Graph Matching (KerGM)[23]. Fast QAP is a method to align pairs of networks and offers the possibility to use anchors in the alignment but does not allow to use node attributes such as group labels; KerGM can align simultaneously multiple networks and also incorporate node attributes in the alignment process (see Methods for details). For a systematic comparison, we generate sets of $K \in \{2, 3, 4\}$ noisy copies of the A2 connectome by swapping a fraction σ of the edges in the network with non-edges, with $\sigma \in [0, 0.6]$. For $K = 2$ we compare our approach without using group labels to the Fast QAP and KerGM, and using group labels against KerGM (Fig. 2g; Supplementary Table S1). For $K = 3, 4$, we compare our approach to KerGM, using group labels as node attributes in both cases (Fig. 2h, i; Supplementary Table S1; Methods). Our approach outperforms both QAP and KerGM, since it is able to recover accurate alignments for larger fractions of errors. Importantly, our approach has superior accuracy even when we reduce the number of sampled alignments with run-times comparable to those of KerGM (Supplementary Table S1).

Finally, to further showcase the potential uses of our approach, we note that, in some real-world scenarios, node group labels are only known for some of the nodes. For instance, in large neuronal connectomes[27,28,39,40] neuron type is not always straightforward to establish, and some neurons are left without annotation. In such cases, our probabilistic approach can be used to infer the labels of unannotated nodes. Specifically, for each alignment sampled from the posterior, each node i_k in network k is mapped to a labeled node in the blueprint $i_{\mathbf{L}} = \pi^k(i_k)$; and a node i_k in network k with unknown group label $g(i_k)$ is automatically assigned the label of the node in the blueprint to which it is mapped $g(i_k) = g(i_{\mathbf{L}})$. Therefore, by sampling alignments from the posterior, we can estimate the probability $p(g(i_k) = g)$ that an unannotated node has label g as the fraction of sampled alignments in which $g(i_k) = g$. To assess the performance of our approach in this task, we consider the same noisy connectome networks as in Fig. 2a–d, and we assume that one of the networks is fully annotated, but the others have a fraction of unlabeled nodes. In Fig. 2e, f, we show that our approach is able to recover ground truth annotation for all nodes with unknown labels, both when unlabeled nodes are randomly chosen in each observed connectome, and in the harder case in which unlabeled neurons are the same across connectomes (Supplementary Figs. S5, S6).

The probabilistic approach correctly aligns real networks

Having validated our method on synthetic networks for which the underlying assumptions are fulfilled by construction, we now explore its performance on sets of real networks, for which the assumptions may not be fulfilled. We consider three real network datasets (see Data for details): (i) four neural connectomes of the *C. elegans* nematode corresponding to four different developmental stages[25]; (ii) the left and right hemispheres of the brain of the larva of *D. melanogaster*[26]; (iii) the yearly e-mail communication networks within an academic institution for four consecutive years[41]. In each of these cases, we make use of whatever information is available from the nodes, and incorporate it to to the MCMC sampling to make it more efficient; we compare our results to those of Fast QAP and KerGM under the same conditions. In Table 1, we show a summary of the comparison, and in what follows we discuss each experiment in detail.

***C. elegans* connectome at different stages of development.** We consider four connectomes of *C. elegans* at different stages of development[25]: two late larval samples (L2, L3) and two adult samples (A1, A2). In this dataset, each connectome comprises 224 neurons and each neuron is associated to one of six different neuron types, so we can use this information to constrain the search of alignment permutations. Yet, the alignment of these connectomes is challenging because the density of neuron-to-neuron connections increases throughout development, so that L2 and L3 connectomes are sparser

Table 1 | Comparison of accuracy and run-times for real-world networks

	Sampling			QAP			KerGM			
Dataset (#nets, #nodes)	Acc.(%)	t(s)	Algs. (#runs)	Acc.(%)	t(s)	Algs.	Acc.(%)	t(s)	Algs.	Rank
C. elegans	**88.4**	902	200 (10)	86.4 ± 0.4	5.4	40	39 ± 3	50	40	20
(2, 224)	87.5	222	40 (5)				34 ± 4	62	40	70
C. elegans	**94.0**	19,297	400 (50)	–	–	–	56 ± 1	286	40	20
(4, 224)	91.6	2088	40 (5)				56 ± 2	302	40	70
D. melanogaster	**79.2**	20,471	105 (15)	77.2	146	40	59.63 ± 0.05	2615	40	20
(2, 1,235)	78.6	9538	42 (6)				69.48 ± 0.07	3031	40	70
Emails	**95.9**	5113	300 (20)	–	–	–	67 ± 1	162	40	20
(4, 170)	95.1	142	50 (5)				80 ± 0.5	125	40	70
Emails	**96.1**	24,379	280 (20)	–	–	–	25 ± 1	1943	40	20
(4, 356)	94.0	1074	50 (5)				59 ± 2	1789	40	70

We show results for our approach (Sampling), the Fast QAP method (QAP) and the multi-way Kernel Graph Matching Method (KerGM). The column **Algs**. indicates the number of alignments obtained with each method for the reported accuracy (**Acc**.). For our method, we also report in parenthesis the number of different initializations (#runs)(Methods). The column (**t**) shows run-times for in seconds. For QAP, if we find several alignments with the same score, we report the mean and standard deviation of the accuracy of those alignments. For KerGM, we report the mean accuracy against the ground truth as well as the standard error of the mean for different dimensions of the kernel (**Rank** = {20, 70}) (Methods). Numbers in bold indicate the best alignment accuracy for each set of networks.

than A1 and A2 connectomes (Supplementary Fig. S4). This implies that the network observations are not fully consistent with our assumption that they have been generated from the blueprint with the same copy error probability. Nonetheless, we find that the most probable mapping of each neuron (the mapping that maximizes the probability $p(\pi^k(i_k) = i_{\mathbf{L}}|\{\mathbf{A}^k\})$, obtained by averaging over alignments sampled in the equilibrium zone) recovers the ground truth identity for 94% of the neurons (Fig. 3a–b), compared to 56% identities for KerGM (Table 1). By contrast, as for the synthetic connectomes, the ground-state alignment is significantly further from the ground truth, as it misaligns 33.5% of the neurons due to the left-right symmetry in the data (an issue that is also apparent in the typical alignments obtained using KerGM; Supplementary Fig. S7). Therefore, in this case, it is critical to average over the ensemble of alignments in which different nodes in different connectomes are misaligned, to recover an alignment closer to the biological ground truth. Importantly, the advantage of sampling is already apparent even if we substantially reduce the number of samples and run times (Supplementary Fig. S7; Table 1).

To further benchmark the performance of our approach, we also looked at the alignment of two adult connectomes (A1, A2), in which case we can also compare to Fast QAP. In general, aligning pairs of networks happens to be a harder problem (because less information is available about nodes and their connections) but, again, the alignment obtained using our approach is better than those obtained using any of the other methods (Table 1; Supplementary Fig. S8). However, in contrast to the alignment of four connectomes, the most probable mapping does not improve over the ground state (Supplementary Fig. S8). This is because for only a pair of networks the alignment landscape is dominated by one deep minimum.

Finally, we investigate the ability of our approach to infer unknown group labels (that is, neuron types), assuming that A2 is fully annotated and that L2, L3, and A1 have 15% of the nodes without annotation (Fig. 3c–e). This is a harder problem, since unannotated nodes are unconstrained by group label and the number of possible permutations is, thus, larger. Nonetheless, we find that, again by sampling from the posterior distribution of alignments, the most probable mapping of each neuron recovers the ground truth identity in 92.8% of cases, and that our approach is able to fully recover unknown labels. Qualitatively, these results are robust to changes in the selection of the fully annotated connectome and the unannotated nodes, and to other variations in the experiment (Supplementary Figs. S9–S11).

Left and right brain hemispheres for the *Drosophila melanogaster* larva. Next, we consider the connectome for the full brain of the larva of the *Drosophila melanogaster* fly[26]. Neurons in the fly brain can be classified into hemispheres that are assumed to be mirror copies of one another, which is consistent with our hypothesis of an underlying blueprint. Our goal is thus to use our probabilistic approach to align the two hemispheres and show that our method is valid to align connectomes comprising over a thousand neurons. For this task, we constrain the search space of permutations by using two pieces of information (Data): (i) the neuroblast lineage of each neuron; and (ii) a set of 292 anchors, that is, pairs of neurons for which the precise mapping is well established (also called "seeds" in the literature[12]). In Fig. 4, we show that the most probable mapping of each neuron recovers the ground truth mapping in 79.2% of the cases (or 78.6% if we consider a smaller set of alignments). This is higher than just looking at the ground state (77.1%), the best alignment we obtained using a seeded QAP approach[12] (76.9%), and also the mean accuracy obtained using KerGM (69.5 %) (Table 1 and Supplementary Fig. S12).

Note also that neither anchors nor mismatched neurons are homogeneously distributed. For example, some groups are hard to align because the topological overlap between connections of ground truth pairs and those with other neurons of the same type is often large, such as for Kenyon cells (KC; neurons in the mushroom body of the brain) or gustatory external neurons (Supplementary Figs. S13, S14). For KC neurons misalignment between neurons is to be expected, since these neurons establish many connections following a random pattern with well defined pre-synaptic and post-synaptic partners, and therefore they are topologically hard to distinguish[42]. Nonetheless, this is not necessarily the case for other neuron types such as gustatory neurons, for which there are clear topological partners. For some of these misaligned neuron pairs, we have checked that our tool finds the topologically optimal solution that does not correspond to the biological ground truth, suggesting that either the annotation is incorrect or that there is further biological information the we would need to incorporate into the tool to be able to recover the biological ground truth mapping.

E-mail communication networks. As a final case study, we consider a completely different network representing the emails exchanged between individuals within an academic institution during four consecutive years (2007-2010)[41]. For each year, we define a directed network where the nodes are individuals and links represent the existence of stable e-mail communication between them during the years we

https://doi.org/10.1038/s41467-025-59077-7

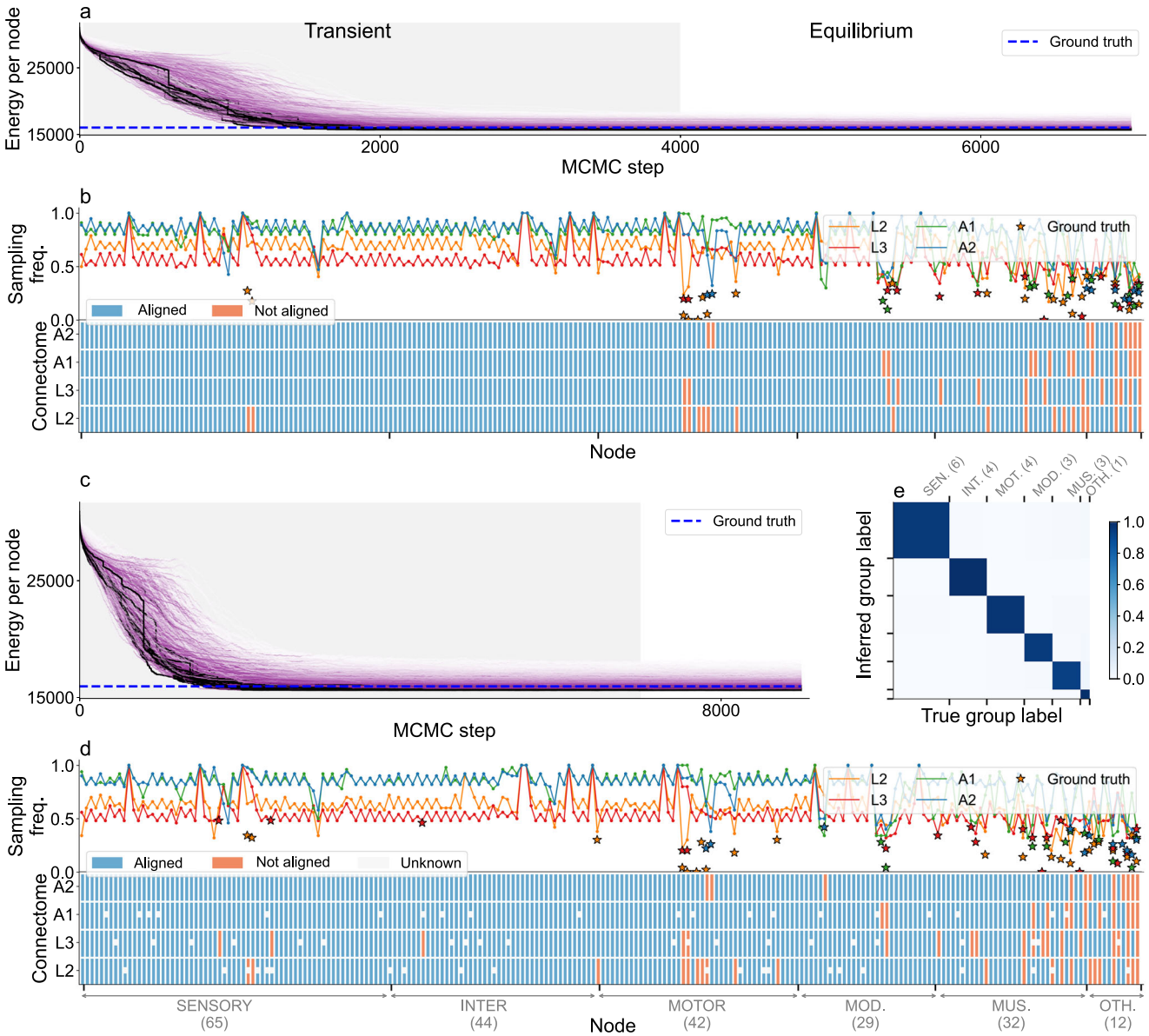

Fig. 3 | Alignment of four real *C.elegans* connectomes corresponding to different developmental stages. a, **b** Network alignment using group labels (neuron types). **a** Same as in Fig. 2a, **b** same as Fig. 2d.In this case the ground truth alignment (blue dashed line) is provided in ref. 25. Additionally, in (**b**) stars indicate the probability with which misaligned neurons are mapped to the ground truth identity.**c**–**e** Network alignment with some nodes with missing group labels. We assume that, for some nodes chosen at random in networks *L*2, *L*3 an *A*1, we do not have information about their group label (neuron type), while neurons in A2 are fully annotated. **c**, **d** same as **a**, **b** respectively.In the bottom panel of (**d**) nodes whose group label is assumed to be unknown are indicated by middle gray rectangles. **e** Probability matrix for the inferred group labels of the selected nodes, as in 2e, f. See Supplementary Figs. S9–S11 for results in slightly different experimental conditions.

consider (Data). In particular, we consider two cases of stable yearly communication between individuals that result in two networks of 170 nodes ($A_{ij} = 1$ if i sends at least 25 emails to j in a year) and 356 nodes ($A_{ij} = 1$ if i sends at least 12 emails to j in a year), respectively. E-mail communication between two individuals can vary significantly from year to year[41], which makes this a harder problem and serves as a benchmark for networks with large copy error.

The data includes the organizational unit to which each individual is affiliated, so we use this information to constrain the possible alignment permutations. In Fig. 5, we show that for both networks most of the alignment permutations we sample have an energy which is lower than that of the ground truth alignment. As in the case of *C. elegans*, we find that sampling over alignments increases significantly the ability to recover ground truth alignment. Using sampling we recover the ground truth identity for 95.9% and 96.1% of the nodes for the 170 and 356-node network, respectively (95.1% and 94% if we use a reduced sampling strategy), which is substantially higher than looking at the ground state alignment (92.8% and 89.3%) or using KerGM (average 80% and 59%) (Table 1 and Supplementary Fig. S15).

Discussion

We have introduced a probabilistic approach to the problem of network alignment. In our approach, we assume that topologically correlated network observations are generated from the same underlying blueprint via a noisy copying mechanism with errors. By contrast to other approaches, the assumptions of the underlying generative model are explicit and interpretable, and general enough to be applied to a wide range of contexts. In particular, it opens the door to

https://doi.org/10.1038/s41467-025-59077-7

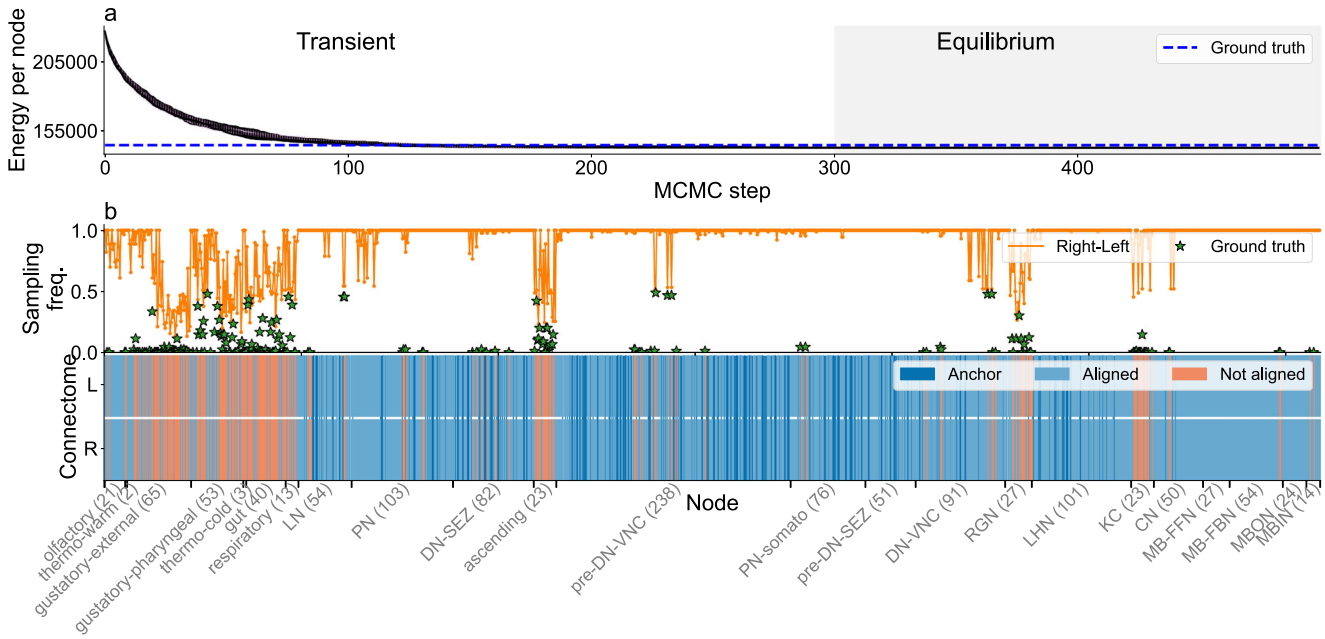

Fig. 4 | Alignment of the right and left hemispheres of the brain of the *Drosophila melanogaster* larva. a Same as in Fig. 2a, b same as Fig. 2d. In this case the ground truth alignment (blue dashed line) is provided in ref. 26. In the top panel of (**b**) stars indicate the probability with which misaligned neurons are mapped to the ground truth identity. In the bottom panel apart from the aligned and misaligned nodes, there are nodes set as anchors (dark blue) as indicated in the dataset (292 out of 1235 in each hemisphere).

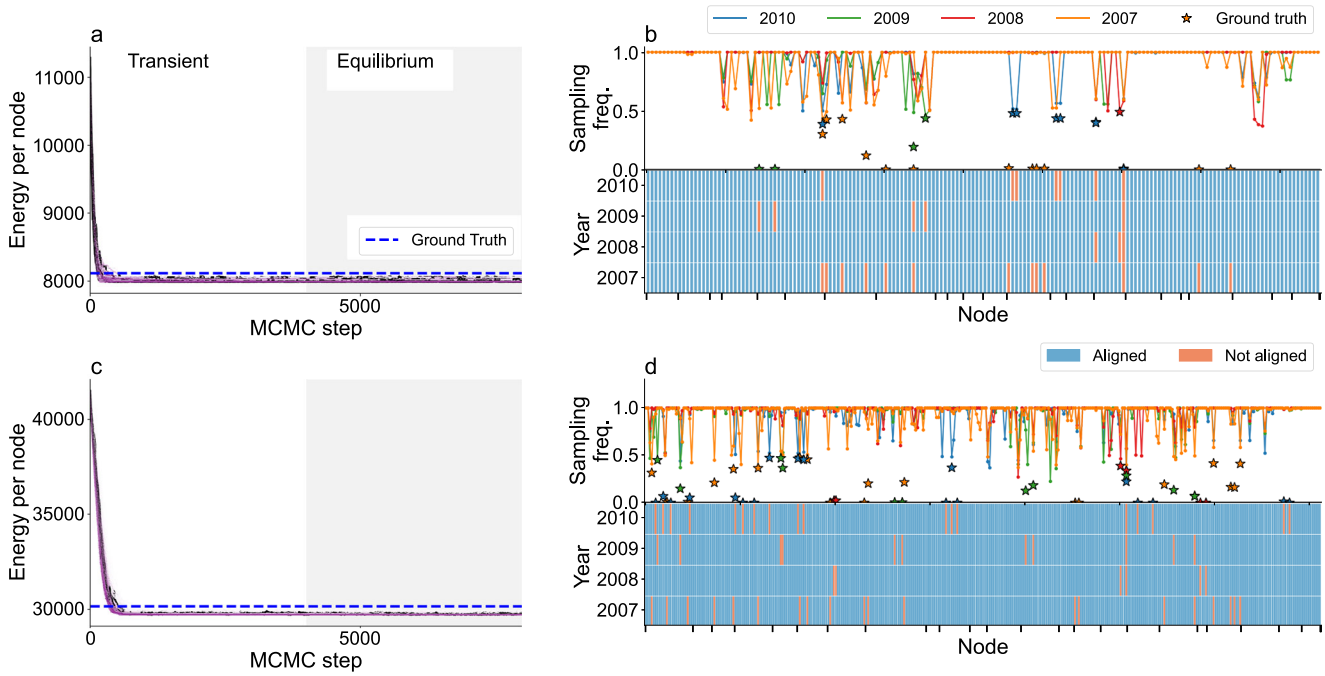

Fig. 5 | Alignment of university email networks corresponding to four consecutive years[41]. Multiple network alignment using the information about the group label (organizational unit) of each node, for different constrained set of users, (**a**, **b**) for 170 users and (**c**, **d**) for 356 users in 23 organizational units (see Data). **a**, **c** Same as in Fig. 2a, b, d same as Fig. 2d. In the top panels of (**b**, **d**) stars indicate the probability with which misaligned neurons are mapped to the ground truth identity.

developing methods to align time-evolving networks by incorporating a time-dependent generative mechanism or to align networks with a different number of nodes by incorporating node copying mechanisms into the generative process.

Our approach also allows us to easily incorporate information on node attributes such as group labels or known identity mappings across networks to constrain the space of plausible alignments efficiently. Such modifications of the basic underlying model are often hard to implement in heuristic methods. Also in contrast to the majority of approaches, our probabilistic approach enables us to address the problem of the simultaneous alignment of multiple networks.

Additionally, within our probabilistic framework, one is not limited to finding a single alignment that optimizes a specific quality metric. Rather, the method yields the full posterior distribution of network alignments, which leads to more complete answers to any question related to or dependent on network alignment. In particular, averaging over possible alignments and obtaining the most probable mapping for each individual node enables us to obtain accurate mappings of node identities, even in situations where the single most plausible alignment fails to correctly match many of the nodes. Our results show that our sampling approach provides a major advantage with respect to recent methods such as KerGM[23] that allow for the simultaneous alignment of several networks using node attributes and other fast, reliable methods for the alignment of pairs of networks such as Fast QAP[38] (Supplementary Table S1 and Table 1).

The sampling of the alignment space, in principle, comes with the price of longer computational times than heuristic optimization approaches. Nonetheless, we find that, with a few exceptions, if we reduce the number of sampled alignments, the accuracy we obtain is still superior to that of other methods with run-times that are comparable to those of KerGM (See Supplementary Table S1 and Table 1). In harder problems, for which it takes a longer time for our tool to equilibrate, the improved accuracy justifies a more computationally costly approach. Future research will likely need to combine our approach with more scalable approaches to easily find the region in alignment space that has a large contribution to the posterior. All in all, our approach is a qualitative step forward in the problem of network alignment that opens the possibility to develop a new generation of powerful algorithms to solve context-dependent network alignment problems in the biological and social sciences.

Methods

Data

C. elegans. We consider the connectomes of *C. elegans* available for larval (L) and adult (A) stages of development in ref. 25. This dataset comprises 8 connectomes in total, for stages L1(4 samples), L2, L3, L4, A1, A2. Each connectome includes 224 neurons and provides information about the number of synapses, w, between neurons. The dataset also includes the mapping of neuron identities across the connectomes, which we use as the ground truth in our analysis. In our study, we represent these networks with binary adjacency matrices, A_k, where each edge indicates the presence or not of a synaptic connection, thus $A_{i,j} = 1$ if $w_{i,j} \geq 1$, and $A_{i,j} = 0$ if $w_{i,j} = 0$. The dataset also includes information about the neuroblast lineage of each neuron, which divide them into six categories: Sensory neurons(65 neurons), interneuron (44), motor neuron (42), modulatory neuron (29), muscle (32), and others (12). Supplementary Fig. S4 illustrates the in-out degree of connections for neurons in each of the connectome networks, ordered according their neuroblast lineage.

D. melanogaster. We consider the connectome for the entire brain for the larva of *Drosophila melanogaster* available in ref. 26 which comprises 2,952 neurons. This dataset includes the neuron locations within the right and left hemispheres, which are nearly mirror images of each other. Additionally, it also provides the mapping of neuron identities between the two hemispheres, which we use as ground truth in our analysis. We construct two binary connectomes networks with 1,235 neurons each, corresponding to the right and left hemispheres. To build these connectomes, we only consider the presence or absence of axon-dendrite connections between neurons within the same hemisphere (thus only including neurons for which their hemisphere location is known), using binary adjacency matrices, as in the *C.elegans* dataset. This dataset also includes the neuroblast lineage of each neuron, which divide them into 17 categories (sensory, LN, PN, DN-SEZ, ascending, pre-DN-VNC, PN-somato,pre-DN-SEZ, DN-VNC, RGN, LHN, KC, CN, MB-FFN, MB-FBN, MBON, MBIN). Supplementary Fig. S16 illustrates the in-out degree of connections for neurons in each of the hemispheres, ordered according their neuroblast lineage. For sensory neurons, we use additional annotation to subdivide them into seven subgroups (visual, olfactory, thermo-warm, gustatory-external, gustatory-pharyngeal, thermo-cold, gut, respiratory). Finally, we incorporate information about certain neurons with condently identified mapping, referred to as anchors or seeds: a set of 292 neurons in each hemispheres (Supplementary Table S2; obtained from https://l1em.catmaid.virtualflybrain.org).

E-mail network data. We consider a dataset of e-mails exchanged among individuals workers within an academic institution over four consecutive years (2007, 2008, 2009, 2010)[41]. The number of users is 1514, 1608, 1878, and 2066, respectively. For each user pair (i, j), we collect the total number of e-mails w_{ij} that i sends to j. To work with alignable networks, we are interested in constructing suitable,stable e-mail connections. In particular, we construct two set of networks with different constraint conditions. The first, common step is to include only the individuals/users that are present across all four years (1282 nodes in total) and consider the communications between them. The second step is to include only stable communications: we define there exists a stable communication between users $i - j$ when individual i sends a minimum of emails per year to individual j (or vice-versa). For the first set of networks, we choose a tighter constraint of sending a minimum of 25 emails per year(approximately ~2 emails/month), while for the second only 12 emails/year (approximately ~1 email/month). Thus, we construct binary directed networks such where $A_{ij} = 1$ if $w_{i,j} \geq 25$, and $A_{i,j} = 0$, otherwise. We further constrain our networks to only those individuals that have communicated each year at least with 5 other users, resulting in networks with 170 individuals in the first set, and 356 in the second case. These individuals are distributed into 23 organizational units, which we use to constrain the space of plausible alignments. Supplementary Fig. S17 illustrates the in-out degree of connections for neurons in each of the connectome networks, ordered according their organizational units.

Generative model details

The likelihood of observing a network with adjacency matrix $\mathbf{A}$ according to our generative model is given by Eq. (3). If we have k independent network observations with adjacency matrices $\{\mathbf{A}^k\}$ the likelihood is the product of likelihoods for each observation

$$\begin{aligned} p(\{\mathbf{A}^k\}|\mathbf{L}, q, p, \{\pi^k\}) &= \prod_k q^{o^k_{10}} p^{o^k_{01}} (1-q)^{o^k_{11}} (1-p)^{o^k_{00}} \\ &= q^{O_{10}} p^{O_{01}} (1-q)^{O_{11}} (1-p)^{O_{00}} \end{aligned} \tag{8}$$

where $O_{XY} = \sum_k o^k_{XY}$.

To compute $p(\mathbf{L}, \{\pi^k\}, q, p|\{\mathbf{A}^k\})$, we use Bayes theorem, and consider a Beta prior for both q and p, with hyper-parameters (α_q, β_q) and (α_p, β_p), respectively:

$$p(\mathbf{L}, \{\pi^k\}, q, p|\{\mathbf{A}^k\}) = \frac{p(\{\mathbf{A}^k\}|\mathbf{L}, q, p, \{\pi^k\})\, p(\mathbf{L}, q, p, \{\pi^k\})}{p(\{\mathbf{A}^k\})} \tag{9}$$

$$= \frac{q^{O_{10}+\alpha_q-1} p^{O_{01}+\alpha_p-1} (1-q)^{O_{11}+\beta_q-1} (1-p)^{O_{00}+\beta_p-1}}{B(\alpha_q, \beta_q) B(\alpha_p, \beta_p)} \frac{p(\mathbf{L}, \{\pi^k\})}{p(\{\mathbf{A}^k\})}. \tag{10}$$

Finally, to obtain $p(\mathbf{L}, \{\pi^k\}|\{\mathbf{A}^k\})$, we integrate over both p and q to obtain the following expression for the posterior

$$p(\mathbf{L}, \{\pi^k\}|\{\mathbf{A}^k\}) = \frac{p(\mathbf{L})p(\{\pi^k\})}{p(\{\mathbf{A}^k\}))\mathrm{B}(\alpha_q,\beta_q)\mathrm{B}(\alpha_p,\beta_p)} \frac{\Gamma(O_{11}+\beta_q)\Gamma\left(O_{10}+\alpha_q\right)}{\Gamma\left(Kn_1+\alpha_q+\beta_q\right)} \frac{\Gamma(O_{00}+\beta_p)\Gamma\left(O_{01}+\alpha_p\right)}{\Gamma\left(Kn_0+\alpha_p+\beta_p\right)}, \tag{11}$$

where n_1 and n_0 are the number of edges and non-edges in the latent blueprint $\mathbf{L}$, respectively.

Monte Carlo simulation and graphical representation

The optimization of the posterior distribution, or equivalently, the minimization of the proposed energy function, was conducted employing a Markov Chain Monte Carlo (MCMC) algorithm using parallel tempering.

MCMC movements. We propose identity swaps between pairs of nodes within a specific network, i.e., we select a network k, choose a pair of nodes (i, j), and propose the swap $\pi^k_{\mathrm{new}}(i) = \pi^k_{\mathrm{old}}(j)$, $\pi^k_{\mathrm{new}}(j) = \pi^k_{\mathrm{old}}(i)$, accepting the change based on Metropolis acceptance.

When nodes have group labels, identity swaps are only allowed between nodes with the same group label. For nodes with an unknown group label, we proceed as follows: at each step, these unlabeled nodes are temporarily assigned to a group based on the node in the blueprint to which they are mapped (that always have a group label since one of the networks is fully labeled). In this case, two types of swaps are allowed: (i) swaps between pair of nodes within the same group (where the unlabeled nodes participate within their current group); and (ii) swaps among only unlabeled nodes. Additionally, we introduce an extra movement when aligning four networks (where three of the networks have some nodes unlabeled and one is fully labeled): we propose a collective swap between a group of three unlabeled nodes (from three different networks) that are mapped to the same node in the blueprint, and another group of three unlabeled nodes mapped to a different node in the blueprint.

Parallel tempering. Monte Carlo Parallel Tempering algorithms[43] enhance sampling efficiency by simultaneously running several replicas of the system at different temperatures.

In this study, we use 17 replicas with energies $\{\mathcal{H}_i\}$ at different inverse temperatures $\{\beta_i\}$. After every four Monte Carlo steps (where 1 MCS represents an attempt to swap each node in each network) across all the replicas, we propose an exchange between a pair replicas (1, 2) at adjacency temperatures. The acceptance probability for this exchange is then:

$$acc_{12} = \min\{1, \exp(-(\beta_1-\beta_2)(\mathcal{H}_2-\mathcal{H}_1)\}. \tag{12}$$

To guarantee a sustained acceptance for these exchanges, we set the following schedule for $\{\beta_i\}$:

$$\beta_{s_i} = \beta_0^{\alpha_{s_i}}, \qquad \text{where} \qquad \alpha \in [-40, 0],\ \beta_0 = 1.03 \tag{13}$$

This broad range is chosen to avoid convergence issues in the presence of multiple minima due to increased noise.

Sampling. Monte Carlo methods allow us to sample the space of alignment permutations, providing probabilistic insights. After an initial thermalization stage, during which the system reaches equilibrium (indicated by a gray zone in each plot), we begin storing alignments $(\mathbf{L}, \{\pi^k\})$ of the replica at $T = 1$ every 50 MCMC steps, except for the case of the four real *C.elegans* connectomes and e-mail networks; for these, we sample every 400 and 500 MCMC steps, respectively, to ensure uncorrelated alignments. In our analysis, the most efficient sampling of the alignment landscape comes from performing several runs, that is by performing different initializations of the alignment algorithm. For the results presented in the main text, the number of runs used for synthetic and real-world networks is reported in Supplementary Table S1 and Table 1, respectively.

Initial configuration. We start the algorithm by sorting the nodes in each network by degree, with the additional constraint of the group labels. Nodes are initially matched based on this degree ranking, except for the alignment of identical network identical copies, where we initialize the algorithm with a random permutation, as degree-based ranking alone would nearly produce a perfect alignment.

Once the initial permutation is established, the blueprint is defined as the majority configuration of all $\mathbf{A}^k$ matrices under this permutation. That is, $L^\star_{ij} = \theta\left(\left(\frac{1}{K}\sum_k A^k_{\pi^k(i),\pi^k(j)}\right) - 0.5\right)$, where θ is the step function. For an even number of networks K, if there is no majority, we set $L^\star_{ij} = 0$ because in sparse networks this is the most likely value. However, performance remains unaffected if we set $L^\star_{ij} = 1$.

Most probable mapping for each node. Our probabilistic approach and corresponding MCMC algorithm allows us to sample from the energy landscape permutations and their associated blueprints, enabling us to estimate the probability $p(\pi(i_k) = i_{\mathbf{L}})$ that node i_k in network A^k is mapped to node $i_{\mathbf{L}}$ in the blueprint. This probability is calculated as the fraction of sampled alignment permutations in which $\pi^k(i_k) = i_{\mathbf{L}}$. We then identify the most probable mapping for each node as the one that maximizes $p(\pi(i_k) = i_{\mathbf{L}})$.

Comparison of multiple network alignments with the ground truth. In all cases studied in the paper, the ground truth is known, enabling us to assess the performance of our method by comparing our results against it. For network pair alignments, the comparison and its graphical representation are straightforward: if two nodes, that are paired according to the ground truth, are mapped to the same node in the blueprint, then both nodes are correctly mapped (represented in blue in visualizations). Nonetheless, comparing alignments across more than two networks is more complex. In our approach, we handle this as follows: for each $(\{\pi^k\}, \mathbf{L})$ tuple, we assign a label to each node $i_{\mathbf{L}}$ in the blueprint based on the most frequent id among the nodes mapped to $i_{\mathbf{L}}$, i.e., nodes for which $\pi^k(i) = i_{\mathbf{L}}$. For instance, if $\pi(a)^1 = \pi(a)^2 = \pi(m)^3 = \pi(a)^4 = i_{\mathbf{L}}$, we assign label a to node $i_{\mathbf{L}}$. If there is not a majority consensus among the nodes ids, we assign label NaN to node $i_{\mathbf{L}}$. Using this labeling, we assess each alignment $(\{\pi^k\}, \mathbf{L})$ by checking whether each node in each network is correctly mapped or not, comparing node id to the label of its corresponding node in the blueprint. For an ensemble of alignments $\{(\{\pi^k\}, \mathbf{L})_i\}$, we determined the most probable alignment of node i as the label of the blueprint node to which it is most frequently mapped.

Comparison with other tools

Fast QAP. We use the implementation provided in the Graspy Python package[38]. For the synthetic copies, we use the unseeded version. For the alignment of the right and left hemispheres of the *D. melanogaster* larva, we use the seeded version also provided in the Graspy package.

KerGM. We use the multi-way kernel method in https://github.com/fxdupe/graphmatchingtools[23]. This method offers the possibility to simultaneously align multiple networks using node and edge attributes. In our analysis, the only group attributes we use are group labels, which can be represented as a 1-dimensional attribute in the KerGM formalism (i.e., all nodes with the same group label have the same node attribute), so we set the parameter dim = 1. The other main

parameter is the rank, that is, the dimension of the Kernel space. In our analysis, we show the results for two values of this parameter rank = 20 – which is the default value in the analysis of random graph alignment in[23], and rank = 70–which we find works better for real networks. The other parameters, such as the number of iterations, had little effect on the accuracy of the alignment, so we used the default value niter = 100. Note that the KerGM implementation produces multi-network alignments without assigning a cost function, therefore as in ref. 23, for each network we report the mean accuracy and the standard error of the mean of a set on nruns = 40 different alignments obtained for different random initial conditions.

For the case of the alignment of the left and right hemispheres of the larval brain of *D. melanogaster*, we use both group labels and anchors (i.e., known mapping of pairs of neurons). To do this, we use different node attributes for each pair of anchor nodes.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The data that support the findings of this study are available from published references:[11,25,26,41].

Code availability

Code is available at https://github.com/teelasa/Connectome_alignment.git. https://doi.org/10.5281/zenodo.14946049[44]

References

1. Emmert-Streib, F., Dehmer, M. & Shi, Y. Fifty years of graph matching, network alignment and network comparison. *Inf. Sci.* **346-347**, 180–197 (2016).
2. Milano, M. et al. An extensive assessment of network alignment algorithms for comparison of brain connectomes. *BMC Bioinforma.* **18**, 235 (2017).
3. Li, Z., Arroyo, J., Pantazis, K. & Lyzinski, V. Clustered graph matching for label recovery and graph classification (2023). 2205.03486.
4. Vijayan, V. & Milenković, T. Multiple network alignment via multimagna++. *IEEE/ACM Trans. Comput. Biol. Bioinforma.* **15**, 1669–1682 (2018).
5. Haller, S. et al. A comparative study of graph matching algorithms in computer vision. In Avidan, S., Brostow, G., Cissé, M., Farinella, G. M. & Hassner, T. (eds.) *Computer Vision – ECCV 2022*, 636–653 (Springer Nature Switzerland, 2022).
6. Zanfir, A. & Sminchisescu, C. Deep learning of graph matching. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2684–2693, https://doi.org/10.1109/CVPR.2018.00284 (2018).
7. Gao, Q., Wang, F., Xue, N., Yu, J.-G. & Xia, G.-S. Deep graph matching under quadratic constraint. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5067–5074, https://doi.org/10.1109/CVPR46437.2021.00503 (2021).
8. Wang, R., Yan, J. & Yang, X. Learning combinatorial embedding networks for deep graph matching. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3056–3065, https://doi.org/10.1109/ICCV.2019.00315 (IEEE Computer Society, 2019).
9. Calissano, A., Maignant, E. & Pennec, X. Towards Quotient Barycentric Subspaces. In *GSI 2023: Geometric Science of Information*, **14071** of *Lecture Notes in Computer Science*, 366–374, https://doi.org/10.1007/978-3-031-38271-0_36 (Springer Nature Switzerland, 2023).
10. Xu, T. et al. Cross-species functional alignment reveals evolutionary hierarchy within the connectome. *NeuroImage* **223**, 117346 (2020).
11. Vogelstein, J. et al. Fast approximate quadratic programming for graph matching. *PLoS ONE* **10**, e0121002 (2015).
12. Fishkind, D. E. et al. Seeded graph matching. *Pattern Recognit.* **87**, 203–215 (2019).
13. Pedigo, B. D., Winding, M., Priebe, C. E. & Vogelstein, J. T. Bisected graph matching improves automated pairing of bilaterally homologous neurons from connectomes. *Netw. Neurosci.* **7**, 522–538 (2023).
14. Trisedya, B. D., Qi, J. & Zhang, R. Entity alignment between knowledge graphs using attribute embeddings. *Proc. AAAI Conf. Artif. Intell.* **33**, 297–304 (2019).
15. Zeng, K., Li, C., Hou, L., Li, J. & g, L. F. A comprehensive survey of entity alignment for knowledge graphs. *AI Open* **2**, 1–13 (2021).
16. Wang, C. et al. Deepmatching: A structural seed identification framework for social network alignment. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, 600–610, https://doi.org/10.1109/ICDCS.2018.00065 (2018).
17. Xiaokai, C. et al. Cross-network embedding for multi-network alignment. 273–284, https://doi.org/10.1145/3308558.3313499 (2019).
18. Le, V.-V., Tran, T. K., Nguyen, B.-N. T., Nguyen, Q. u.-D. & Snasel, V. Network alignment across social networks using multiple embedding techniques. *Mathematics* **10**, https://doi.org/10.3390/math10213972 (2022).
19. Bayati, M., Gleich, D. F., Saberi, A. & Wang, Y. Message-passing algorithms for sparse network alignment. *ACM Trans. Knowl. Discov. Data* **7**, https://doi.org/10.1145/2435209.2435212 (2013).
20. Feizi, S. et al. Spectral alignment of graphs. *IEEE Trans. Netw. Sci. Eng.* **7**, 1182–1197 (2020).
21. Zhang, Z., Xiang, Y., Wu, L. & Xue, B. and Nehorai, A. Kergm: Kernelized graph matching. 32 (2019).
22. Jesús, A., Sussman, D. L., Priebe, C. E. & Lyzinski, V. Maximum likelihood estimation and graph matching in errorfully observed networks. *J. Comput. Graph. Stat.* **30**, 1111–1123 (2021).
23. Dupé, F.-X., Yadav, R., Auzias, G. & Takerkart, S. Kernelized multigraph matching. In Khan, E. & Gonen, M. (eds.) *Proceedings of The 14th Asian Conference on Machine Learning*, **189** of *Proceedings of Machine Learning Research*, 311–326 (PMLR, 2023).
24. Cinaglia, P., Milano, M. & Cannataro, M. Multilayer network alignment based on topological assessment via embeddings. *BMC Bioinforma.* **24**, 416 (2023).
25. Witvliet, D. et al. Connectomes across development reveal principles of brain maturation. *Nature* **596**, 257 – 261 (2021).
26. Winding, M. et al. The connectome of an insect brain. *Science* **379**, eadd9330 (2023).
27. Schlegel, P. et al. Whole-brain annotation and multi-connectome cell typing of Drosophila. *Nature* **634**, 139–152 (2024).
28. Dorkenwald, S. et al. Neuronal wiring diagram of an adult brain. *Nature* **634**, 124–138 (2024).
29. Williams, M. L., Wilson, R. C. & Hancock, E. R. Multiple graph matching with Bayesian inference. *Pattern Recognit. Lett.* **18**, 1275–1281 (1997).
30. Park, H.-M. & Yoon, K.-J. Consistent multiple graph matching with multi-layer random walks synchr onization. *Pattern Recognit. Lett.* **127**, 76–84 (2019).
31. Cullina, D. & Kiyavash, N. Exact alignment recovery for correlated erdos-rényi graphs (2018). 1711.06783.
32. Wu, Y., Xu, J. & Yu, S. H. Settling the sharp reconstruction thresholds of random graph matching. *IEEE Trans. Inf. Theory* **68**, 5391–5417 (2022).
33. Ding, J. & Du, H. Matching recovery threshold for correlated random graphs. *Ann. Stat.* **51**, 1718 – 1743 (2023).
34. Racz, M. Z. & Sridhar, A. Correlated stochastic block models: Exact graph matching with applications to recovering communities. In *Advances in Neural Information Processing Systems* (NeurIPS, 2021).
35. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092 (1953).

36. Vallès-Català, T., Peixoto, T. P., Sales-Pardo, M. & Guimerà, R. Consistencies and inconsistencies between model selection and link prediction in networks. *Phys. Rev. E* **97**, 062316 (2018).
37. White, J., Southgate, E. L., Thomson, J. N. & Brenner, S. The structure of the nervous system of the nematode Caenorhabditis elegans. *Philos. Trans. R. Soc. Lond. Ser. B, Biol. Sci.* **314 1165**, 1–340 (1986).
38. Chung, J. et al. Graspy: Graph statistics in Python. *J. Mach. Learn. Res.* **20**, 1–7 (2019).
39. Scheffer, L. K. et al. A connectome and analysis of the adult *Drosophila* central brain. *eLife* **9**, e57443 (2020).
40. Marin, E. C. et al. Systematic annotation of a complete adult male Drosophila nerve cord connectome reveals principles of functional organisation. *eLife* **13**, RP97766 (2024).
41. Godoy-Lorite, A., Guimerà, R. & Sales-Pardo, M. Long-term evolution of email networks: Statistical regularities, predictability and stability of social behaviors. *PLoS ONE* **11**, e0146113 (2016).
42. Caron, S., Ruta, V., Abbot, L. & Axel, R. Random convergence of olfactory inputs in the Drosophila mushroom body. *Nature* **497**, 113–117, https://doi.org/10.1038/nature12063
43. Earl, D. J. & Deem, M. W. Parallel tempering: Theory, applications, and new perspectives, https://doi.org/10.1039/b509983h (2005).
44. Lázaro, T. Probabilistic alignment of multiple networks, https://doi.org/10.5281/zenodo.14946049 (2024).

Acknowledgements

This research was funded by project PID2022-142600NB-I00 from MCIN/ AEI/10.13039/501100011033 FEDER, UE, and by the Government of Catalonia (2021SGR-633, and 2022FI_B 00516 - TL).

Author contributions

T.L. obtained data, wrote code, and performed experiments. All authors (T.L., R.G., and M.S.) designed research, analyzed results, and wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-025-59077-7.

Correspondence and requests for materials should be addressed to Roger Guimerà or Marta Sales-Pardo.

Peer review information *Nature Communications* thanks the anonymous reviewer(s) for their contribution to the peer review of this work. A peer review file is available.