CONTROLLING RECIPROCITY IN BINARY AND WEIGHTED NETWORKS: A NOVEL DENSITY-CONSERVING APPROACH

Fatemeh Hadaeghi[*], Kayson Fakhar[1,2], Claus C. Hilgetag[1,3]

[1]Institute of Computational Neuroscience, University Medical Center Hamburg-Eppendorf, Hamburg, Germany
[2]MRC Cognition and Brain Sciences Unit, University of Cambridge, Cambridge, UK
[3]Department of Health Sciences, Boston University, Boston, MA, USA
{f.hadaeghi, k.fakhar, c.hilgetag}@uke.de

November 24, 2024

ABSTRACT

We introduce efficient Network Reciprocity Control (NRC) algorithms for steering the degree of asymmetry and reciprocity in binary and weighted networks while preserving fundamental network properties. Our methods maintain edge density in binary networks and cumulative edge weight in weighted graphs. We test these algorithms on synthetic benchmark networks—including random, small-world, and modular structures— as well as brain connectivity maps (connectomes) from various species. We demonstrate how adjusting the asymmetry-reciprocity balance under edge density and total weight constraints influences key network features, including spectral properties, degree distributions, community structure, clustering, and path lengths. Additionally, we present a case study on the computational implications of graded reciprocity by solving a memory task within the reservoir computing framework. Furthermore, we establish the scalability of the NRC algorithms by applying them to networks of increasing size. These approaches enable systematic investigation of the relationship between directional asymmetry and network topology, with potential applications in computational and network sciences, social network analysis, and other fields studying complex network systems where the directionality of connections is essential.

Keywords Network Reciprocity Control (NRC) · Network Asymmetry · Spectral Properties · Reservoir Computing · Brain Connectomics

1 Main

The intricate balance between "reciprocity" and "asymmetry" in complex networks is fundamental to the behavior of diverse systems, ranging from neural circuits to quantum many-body systems. Reciprocity, classically defined as the proportion of directed links with a reciprocal counterpart, is not merely a structural property, but a key factor influencing system-wide behaviors [1, 2]. The balance of directed links profoundly impacts information flow, synchronization, and phase transitions across various network types [3, 4]. Moreover, the degree of reciprocity or asymmetry in the network connections plays a crucial role in determining the robustness and stability of real-world networks, governing critical phenomena observed in both natural and engineered systems [5, 6].

Despite its critical role, for analytical convenience the directed nature of connections is often simplified or overlooked in network models, neglecting the effects of varying levels of reciprocity and asymmetry. However, directionality in real-world networks profoundly impacts their behavior. In directed networks, asymmetry introduces topological features such as trophic coherence [7] and non-normality [8], which significantly shape network dynamics. These structural properties manifest in the graphs' eigenspectra, directly linking them to the stability and behavior of complex systems, including food webs, neuronal circuits, communication networks, and ecological systems [9, 7, 10, 11].

[*]Corresponding author: f.hadaeghi@uke.de

Moreover, structural asymmetry and network directionality have been identified as crucial factors in facilitating converse symmetry breaking or asymmetry-induced symmetry. These phenomena demonstrate how asymmetry, often viewed as a destabilizing feature, can paradoxically enable the stabilization of symmetric dynamical states, such as synchronization, in systems where fully symmetric coupling would otherwise fail [12, 13, 6].

Reciprocal connections play an equally crucial role, significantly affecting dynamical processes, network growth, and the formation of higher-order structures such as motifs and communities [1, 14, 15, 9, 16]. In networks that transport information or materials, such as email systems, the World Wide Web, or social media platforms, bidirectional links are fundamental drivers of network dynamics, facilitating robust mutual exchange and enhancing the efficiency and reach of propagation processes [16, 17]. This significance extends to other network types as well. In biological networks modeling dynamic processes, such as gene regulatory networks or neural circuits, reciprocal connections often represent feedback loops crucial for homeostasis and adaptive responses [18, 19]. In social networks, bidirectional links can signify mutual trust or collaboration, influencing opinion formation and social cohesion [17, 15], while in economic networks, reciprocal trade relationships can impact market stability and economic resilience [20]. Across these varied contexts, reciprocity enables immediate feedback, creating pathways for rapid signal propagation, resource sharing, and the emergence of complex behaviors, potentially leading to phenomena such as synchronization in coupled oscillators, pattern formation in reaction-diffusion systems, or cascading effects in financial networks.

Despite growing recognition of the importance of directionality and reciprocity in shaping network dynamics, studies have traditionally focused on extreme cases of complete symmetry or asymmetry, while the systematic exploration of intermediate levels has only recently gained attention. The current emphasis on extremes leaves a critical gap in understanding how the balance between reciprocity and asymmetry is optimized in real-world networks as well as computational models such as artificial neural networks. While it is clear that the link balance significantly influences stability, functionality, and information flow, models capable of systematically manipulating reciprocity while preserving other key network properties remain limited, with a few notable exceptions, such as [2, 4]. Here, we address this gap by introducing novel algorithms that precisely tune reciprocity in both binary and weighted networks, offering a new approach for directly exploring how reciprocity influences emergent behaviors in complex systems.

We demonstrate the broad applicability of our approach by analyzing how changes in reciprocity impact critical topological features of a network, including in and out-degree distributions, spectral properties, modularity, and clustering coefficients. Additionally, we present a case study on the computational implications of graded reciprocity by solving a memory task [21] within the framework of reservoir computing [22]. This application allows us to explore how varying levels of reciprocity affect a network's ability to store and retrieve information, providing further insight into the broader functional consequences of reciprocity in both structural and dynamical contexts. Furthermore, we establish the scalability of our method across networks of varying sizes and initial reciprocity levels, ensuring its utility for real-world, large-scale network analyses.

2 Defining reciprocity in networks

In a directed binary network, represented by an adjacency matrix, A, where $A_{ij} \in \{0,\ 1\}$, let L denote the total number of links in the network, and $\overleftrightarrow{L}$ the number of reciprocal links, where a reciprocal link refers to a pair of directed edges $(i \to j)$ and $(j \to i)$. The link reciprocity is then quantified as the fraction of reciprocal links relative to the total number of directed links:

$$r = \frac{\overleftrightarrow{L}}{L}. \tag{1}$$

This definition of reciprocity, however, has been criticized for having only relative significance, as the measured value, r, must be compared to the expected reciprocity of a random graph with the same size and density. This comparison complicates the ability to rank different networks in terms of reciprocity, especially, when comparing networks of varying density, because denser networks naturally exhibit more mutual connections by chance [23]. We are aware of this criticism, but since our work aims to adjust the reciprocity of a given network while preserving its size and density (see 3 for details), our baseline remains the original network. In this context, it is acceptable for the adjusted reciprocity value to have a relative meaning, as it serves the purpose of measuring changes within a controlled framework.

Weighted networks, with the adjacency matrix $W \in \mathbb{R}^{N \times N}$, generalize unweighted (binary) networks by allowing links to have different real-valued weights, which can be positive, negative, or zero. The graph can be either undirected, where $W_{ij} = W_{ji}$, or directed, where W_{ij} need not equal W_{ji}.

While the reciprocity of binary networks has been studied extensively, the reciprocity of weighted networks has received much less attention. This fact is due to the more complicated phenomenology at the dyadic level in weighted networks. In a binary graph, it is straightforward to define link reciprocation - a link from node i to node j is reciprocated, if the

link from j to i also exists. However, in a weighted network, assessing whether a link of weight $W_{ij} > 0$ from node i to node j is reciprocated based on the weight W_{ji} of the mutual link is not trivial. Consequently, quantifying reciprocity in weighted networks poses additional challenges [24].

Here, we designed our algorithms based on the formal definition of reciprocity in weighted networks proposed by Squartini et al. [24]. In this formalism, reciprocated weight between i and j (the symmetric component) is represented as

$$\overleftrightarrow{W}_{ij} := \min[W_{ij},\ W_{ji}] = \overleftrightarrow{W}_{ji}, \tag{2}$$

and the non-reciprocated weight from i to j (the asymmetric component) as

$$\overrightarrow{W_{ij}} := W_{ij} - \overleftrightarrow{W}_{ij}. \tag{3}$$

The weighted reciprocity of a weighted network can, therefore, be computed as

$$r := \frac{\overleftrightarrow{W}}{W} = \frac{\Sigma_i \Sigma j \neq i\ \overleftrightarrow{W}_{ij}}{\Sigma_i \Sigma j \neq i\ W_{ij}}. \tag{4}$$

This quantity is, however, scale-invariant, such that if all the weights are multiplied by a scale factor, $W_{ij} \to \alpha W_{ij}$, r does not change, as shown in [24]. This aspect makes the task of adjusting strength reciprocity challenging, particularly under the constraint of preserving cumulative edge weights, $E = \Sigma W_{ij}$.

3 Novel algorithms for modulating network reciprocity

The proposed Network Reciprocity Control (NRC) algorithms operate under several key assumptions. First, they assume that unidirectional links have the highest potential to become reciprocal when adjusting the network towards the desired reciprocity, reflecting the idea that existing connections may be easily converted into bidirectional ones. Additionally, the algorithms presume that converting reciprocal links into unidirectional links does not significantly disrupt the overall network structure, allowing for flexible reciprocity adjustments. Throughout the process, the network density in binary graphs and the total strength in weighed networks are maintained. In the binary case, the network is treated as maximally random while adhering to its specified link density and desired reciprocity, thereby minimizing systematic biases during the adjustment process. This approach contrasts with algorithms that impose strict constraints on the correlation between in- and out-degree sequences, such as those described in [2, 4, 25]. While the density-preserving algorithm minimizes biases in terms of randomness and structural freedom, algorithms that preserve correlations between in- and out-degree sequences introduce specific structural constraints that might introduce biases related to those correlations, such as in the degree distribution or network modularity.

The algorithm designed to adjust reciprocity in the weighted network uses a deterministic strategy. Both algorithms assume symmetric treatment of link or weight modifications, meaning that changes are balanced to preserve the network's topology. Local adjustments in link or weight configurations are considered to have global effects on the network's reciprocity, and the algorithms are designed to approximate the target reciprocity within a small tolerance interval, acknowledging that perfect alignment with the desired value may not always be feasible.

3.1 Binary networks

3.1.1 Algorithm design and implementation

Given a directed binary network represented by an adjacency matrix, A, where $A_{ij} \in \{0,\ 1\}$, the algorithm, detailed in **Algorithm 1**, aims to adjust the network's reciprocity to a specified target value, r_d while maintaining the overall structure and the number of links, i.e, $L = \Sigma_{i \neq j} A_{ij}$.

Step 1: Initial setup
The current reciprocity, r_c, is calculated as the ratio of reciprocal links to the total number of links. If r_c is already within a small tolerance interval of the desired reciprocity, r_d, the algorithm terminates without making changes.

Step 2: Identification of symmetric and asymmetric links
Symmetric (reciprocal) links, denoted by S, are identified where $A_{ij} = 1$ and $A_{ji} = 1$, with $i < j$ to prevent double counting. Asymmetric ($\overrightarrow{L}$) links are identified where $A_{ij} = 1$ and $A_{ji} = 0$.
Step 3: Determine target number of reciprocal links
The target number of reciprocal links, $\overleftrightarrow{L}_r$, is estimated using

$$\overleftrightarrow{L}_r = r_d \cdot L \tag{5}$$

where L represents the total number of links.
Step 4: Reciprocity adjustment

- **Case 1: Reducing reciprocity** ($r_d < r_c$)
  If the number of reciprocal links, $|S|$, is found to exceed the target value $\frac{\overleftrightarrow{L}_r}{2}$, the difference is computed as
  $$\Delta = |S| - \frac{\overleftrightarrow{L}_r}{2}. \tag{6}$$
  To reduce reciprocity, Δ reciprocal links are removed from S by converting them randomly into unidirectional links. To ensure structural symmetry, pruned links are selected from the upper and lower triangles with equal probability. To maintain network density, Δ unidirectional links are distributed to Δ non-existing connections, ensuring that they are not already reciprocal.
- **Case 2: Increasing reciprocity** ($r_d > r_c$)
  When increasing reciprocity, the difference Δ, as computed in Eq. 6, indicates the number of unidirectional links that should be converted to reciprocal links. If sufficient unidirectional links are available ($\overrightarrow{L} \geq 2\Delta$), Δ asymmetric links are converted to reciprocal links to increase reciprocity, and Δ are removed from other unidirectional links to preserve density. If unidirectional links are insufficient ($\overrightarrow{L} < 2\Delta$), all available unidirectional links are converted to reciprocal links. Excess reciprocal links are then removed, and asymmetric links are added to randomly chosen non-existing connections to achieve the desired reciprocity while maintaining network density.

3.1.2 Bounds of reciprocity: Theoretical limits and practical implications

For a directed binary network characterized by an adjacency matrix A, with the total number of links denoted as $L = \sum_{i \neq j} A_{ij}$, our algorithm does not impose constraints on the correlation between in-degrees and out-degrees. This flexibility allows the network to attain full reciprocity, where the reciprocity index $r = 1.0$. Such behavior is consistent with prior findings in network theory [9, 26, 4].

The lower bound of reciprocity, however, is determined by the size of the network and its link density. This section derives the mathematical lower bound of reciprocity for a network with N nodes. The total number of links, L, can be decomposed into two categories: reciprocal links, $\overleftrightarrow{L}$, and asymmetric links, $\overrightarrow{L}$, such that:
$$L = \overleftrightarrow{L} + \overrightarrow{L}. \tag{7}$$
For a sparse network with $L \leq \frac{N(N-1)}{2}$, it is possible to generate a maximally random configuration of the adjacency matrix A where all links are asymmetric ($L = \overrightarrow{L}$) and no reciprocal pairs are present ($\overleftrightarrow{L} = 0$). In this scenario, the lower bound of reciprocity is $r_{\text{min}} = \frac{\overleftrightarrow{L}}{L} = 0$.

However, for networks where $L > \frac{N(N-1)}{2}$, such that $L = \frac{N(N-1)}{2} + \delta$ with $\delta > 0$, at least $\frac{\delta}{2}$ reciprocal link pairs must exist to maintain the link density. In this case, the lower bound of reciprocity is given by:
$$r_{\text{min}} = \frac{\overleftrightarrow{L}}{L} = \frac{\delta/2}{L} = \frac{L/2 - N(N-1)}{L} = \frac{\delta - N(N-1)/2}{N(N-1) + 2\delta}. \tag{8}$$

This formalism defines the limits for asymmetry and emphasizes how network size and density influence the achievable levels of reciprocity in our algorithm.

3.2 Weighted networks

3.2.1 Algorithm design and implementation

Given a weighted adjacency matrix $W \in \mathbb{R}^{N \times N}$, the following algorithm, detailed in **Algorithm 2**, is proposed to adjust strength reciprocity while preserving the sum of connection strengths, $E = \sum W_{ij}$. It should be noted that weights are restricted to non-negative values, i.e., $W_{ij} \geq 0$ for $i = 1, 2, ..., N$.

Step 1: Initial setup

The current reciprocity, r_c, is computed using Eq. 4. If r_c is found to be close to the desired weighted reciprocity, r_d (within a small tolerance), the algorithm is terminated without making any adjustments.

Algorithm 1 Adjust Reciprocity of Binary Network

Require: Binary adjacency matrix A, desired reciprocity r_d
Ensure: Adjusted adjacency matrix A'

```
Remove self-loops: Set A_ii = 0 for all i
Compute current reciprocity r_c and total links L

if r_c ≈ r_d then
    return A
end if
Compute number of desired reciprocal links: n_r = r_d · L
Identify symmetric link pairs S and their count n_s
Compute difference: d = 2n_s − n_r
if d ≥ 0 then
    Remove n_remove = n_s − n_r/2 reciprocal links:
        Randomly select n_remove pairs from S
        For each pair, randomly set one direction to 0
    Identify symmetric zero pairs Z and their count n_z
    if n_z < d then
        return error (density too high for this desired reciprocity)
        return A
    else
        Add n_remove asymmetric links to non-connected pairs:
            Randomly select n_remove pairs from Z
            For each pair, randomly set one direction to 1
    end if
else
    Compute n_add = |d|
    Identify asymmetric link pairs A_s
    if |A_s| ≥ n_add then
        Randomly select n_add pairs from A_s to make reciprocal
        Randomly select n_add pairs from A_s to remove
    else
        Make all A_s reciprocal
        Compute n_remove = |A_s|/2
        Randomly select n_remove reciprocal pairs to remove
    end if
end if
return adjusted matrix A'
```

Step 2: Identification of symmetric and asymmetric network components

The symmetric component of the adjacency matrix, denoted by $\overleftrightarrow{W}$, is identified where $W_{ij} = W_{ji} = \min[W_{ij}, W_{ji}]$ (see Eq. 2). The asymmetric component, $\overrightarrow{W}$, is then identified as the difference $\overrightarrow{W} = W - \overleftrightarrow{W}$.

Step 3: Determination of reciprocity scaling factor

The reciprocity scaling factor, β, is computed based on the desired strength reciprocity, r_d, as:

$$\beta = \frac{r_d}{r_c}. \tag{9}$$

Step 4: Reciprocity adjustment

- **Case 1: Reducing reciprocity** ($\beta \leq 1$)
  If a reduction in strength reciprocity is required, the symmetric component of the adjacency matrix is scaled down by a factor of β, such that $\overleftrightarrow{W}_{ij} \to \beta \overleftrightarrow{W}_{ij}$. To preserve the total strength, E, the asymmetric component,

$\overrightarrow{W}$, is scaled up accordingly. The decrease in the symmetric component reduces the total sum by:

$$\varepsilon = (1-\beta)\sum_{i}\sum_{j\neq i}\overleftrightarrow{W}_{ij}. \tag{10}$$

To compensate for this reduction, the positive elements of the asymmetric component are adjusted. First, the indices of positive elements in $\overrightarrow{W}$ are identified as:

$$I^{+} = \{(i,j)|\overrightarrow{W}_{ij} > 0\}.$$

The number of such elements is counted as $n^{+} = |I^{+}|$. If $n^{+} > 0$, the increment to be added to each positive element is computed as:

$$\delta = \frac{\varepsilon}{n^{+}}.$$

Finally, each positive element in $\overrightarrow{W}$ is updated as:

$$\overrightarrow{W}_{ij} \rightarrow \overrightarrow{W}_{ij} + \delta \quad \text{for all} \quad (i,j) \in I^{+}.$$

This procedure ensures that the total deviation, ε, is evenly distributed while maintaining the total weight sum and preserving the sign of each matrix element.

- **Case 2: Increasing reciprocity** ($\beta > 1$)
  If an increase in strength reciprocity is needed, a direct scaling of the symmetric component is not feasible, as it may affect the values of $\min[W_{ij}, W_{ji}]$ and alter the symmetric part of the matrix. Therefore, an iterative procedure is followed. At each iteration, the scaling factor is recalculated as the minimum of the ratio of the target to the current reciprocity, β, and $r_c + \delta$ where δ is a small step size. This factor controls the gradual scaling of elements in $\overleftrightarrow{W}$. As the symmetric component is adjusted, the asymmetric component is reduced to maintain the total sum of strengths, E, ensuring that no values fall below a small, positive non-zero threshold to satisfy $W_{ij} \geq 0$.

3.2.2 Handling edge cases and exceptional scenarios

In this section, we elaborate on two extreme cases of network reciprocity: complete symmetry ($r = 1.0$) and complete asymmetry ($r = 0$), as handled by our algorithm. These cases allow us to examine how the network evolves under maximum structural constraints while preserving key properties such as total connection strength.

Symmetry Breaking in Initially Symmetric Networks: For weighted graphs $W \in \mathbb{R}^{N\times N}$ with initial reciprocity $r_c = 1.0$, we introduce a small perturbation to break structural symmetry. This is achieved by applying a random mask $P \sim \mathcal{N}(0,\delta)$, where δ is chosen to be smaller than the minimum non-zero element in W. This approach ensures that the perturbation is subtle enough to maintain the overall network structure while introducing the desired asymmetry.

Achieving Full Symmetry: While a fully symmetric network can be trivially obtained as $W^{'} = \frac{(W+W^{T})}{2}$, this method may introduce spurious connections, altering the network's fundamental structure. To maintain consistency in our analysis of network features (detailed in Section 5), we instead employ our iterative algorithm to adjust the strength reciprocity to $r_d = 1.0$. This approach preserves the original connection patterns while changing the strengths to achieve the desired symmetry.

Maximum Asymmetry: To explore the opposite extreme, we develop a method to create fully asymmetric networks. Given a weighted network $W \in \mathbb{R}^{N\times N}$, we set all reciprocal weights to zero ($\overleftrightarrow{W} \rightarrow 0$) and up-scale the remaining non-zero elements to preserve the total edge weight E, while ensuring $W_{ij} \geq 0$. It is important to note that this is the only scenario in which our algorithm removes existing links; in all other cases, unlike its binary counterpart, our method preserves the original network connectivity.

4 Experimental setup and data

The algorithms were applied to a range of synthetic networks—including random, small-world, and modular topologies—as well as empirical connectomes from multiple species. The analysis reveals how modifying the asymmetry-reciprocity balance impacts key network characteristics, including spectral properties, degree distributions, community structures, clustering coefficients, and shortest path lengths.

Algorithm 2 Adjust Strength Reciprocity While Preserving Total Connection Strengths

Require: Adjacency matrix W, target reciprocity r_d, tolerance ϵ
Ensure: Adjusted adjacency matrix W'

Compute current reciprocity r_c using Eq. 4 and total strength $E = \sum W_{ij}$
if $r_c \approx r_d$ **then**
    return A
end if
Compute symmetric component: $\overleftrightarrow{W}_{ij} = \min(W_{ij}, W_{ji})$
Compute asymmetric component: $\overrightarrow{W} = W - \overleftrightarrow{W}$
Compute scaling factor: $\beta = \frac{r_d}{r_c}$
if $\beta \leq 1$ **then**
    Scale down symmetric component: $\overleftrightarrow{W}_{ij} \rightarrow \beta \overleftrightarrow{W}_{ij}$
    Adjust asymmetric component to preserve total strength
        Compute deviation in E: $\varepsilon = (1-\beta) \sum_i \sum_{j \neq i} \overleftrightarrow{W}_{ij}$
        Identify positive elements in $\overrightarrow{W}$: $I^+ = \{(i,j) | \overrightarrow{W}_{ij} > 0\}$
        Calculate increment $\delta = \frac{\varepsilon}{|I^+|}$
    for all $(i,j) \in I^+$ **do**
            Update positive elements in asymmetric component: $\overrightarrow{W}_{ij} \rightarrow \overrightarrow{W}_{ij} + \delta$
    end for
else
    Set iteration limit: max_iter and step size δ
    for $iter = 1$ to max_iter **do**
        Scale symmetric component by factor, $\min(\beta,\ 1+\delta)$
        Update symmetric and asymmetric components
        Adjust asymmetric component to preserve E without violating $W_{ij} \geq 0$
        update r_c
        Check if $|r_c - r_d| \leq \epsilon$ and terminate if condition is met
    end for
end if
return Adjusted matrix W'

Each network type was generated over 50 trials. Random networks were constructed using undirected binary Erdös-Rényi graphs [27]. Undirected small-world networks were modeled as connected Watts-Strogatz graphs [28], while modular networks were generated using stochastic block models [29].

For weighted networks, random non-negative weights ($W_{ij} > 0$) sampled from a uniform distribution over $[0, 1)$ were assigned to the existing links in these previously generated binary networks.

The algorithms were further tested on empirical connectomes from various species, which exhibit diverse sizes, densities, and initial binary or weighted reciprocity values [30, 31, 32, 33, 34, 35]. These key properties across datasets are summarized in Table 1. Representative synthetic and empirical connectomes are illustrated in Figure 1.

5 Impact on network properties

5.1 Spectral analysis

This investigation reveals a clear connection between network reciprocity and the eigenvalue spectrum of the adjacency matrix, a critical measure of global network properties. Through systematic adjustment of reciprocity using the density-preserving algorithm, distinct modifications in the spectral characteristics of both random and structured networks are observed.

As illustrated in Figure 2, an increase in reciprocity is associated with a broadening of the distribution of eigenvalue magnitudes across 50 experimental trials. The symmetry of the real components of the eigenvalues is universally preserved. However, in structured networks, a subtle translation of the symmetry axis is detected when applying the binary algorithm—a phenomenon absent in the weighted algorithm implementation shown in Figure S1.

Table 1: Summary of synthetic and empirical networks analyzed in this study. The networks vary in size, density, and original reciprocity. For synthetic networks, different structures include random, small-world, and modular types. For a network with N nodes and L links, density was computed for directed networks as $L/N(N-1)$. For empirical networks, both weighted and binary representations are considered. The notation r_w denotes weighted reciprocity, while r_b refers to binary reciprocity. For empirical networks originally weighted, a binary representation is generated by establishing a link where $W_{ij} > 0$.

network	**#nodes**	**density**	r_w	r_b
		Synthetic networks		
Random	256	0.05	0.032	0.07
Small-world	256	0.1	0.67	1.0
Modular	256	0.1	0.67	1.0
		Connectomes		
C. elegans	279	0.03	0.12	0.21
Drosophila	49	0.83	0.67	0.92
Human-struct-scale033	83	0.31	1.0	1.0
Human-struct-scale060	129	0.23	1.0	1.0
Human-struct-scale125	234	0.13	1.0	1.0
Human-struct-scale250	463	0.06	1.0	1.0
Human-struct-scale500	1015	0.02	1.0	1.0
Macaque	242	0.07	0.51	0.51
Mouse	112	0.53	0.52	0.67
Rat	73	0.37	0.45	0.66

To quantify the impact of reciprocity on matrix normality, Henrici's index of departure from normality was employed [8]. This index, detailed in Supplementary Note 1, reaches its minimum for normal matrices and increases as matrices deviate from normality. The findings demonstrate that elevated reciprocity consistently enhances normality. This observation holds true for both binary and weighted algorithmic approaches as shown in Figure S2 and aligns with the results reported in [8].

These results provide crucial insights into the interplay between local dyadic relationships and global network topology, unencumbered by confounding density effects. The ability to fine-tune reciprocity while maintaining network density opens new avenues for investigating dynamical processes on directed networks.

5.2 Degree distribution

The algorithm for adjusting network reciprocity presents a novel approach compared to previous methods that operate under prescribed degree correlation constraints [25, 4]. By relaxing these constraints, a more comprehensive exploration of how correlation patterns evolve with changing reciprocity values is enabled. This approach reveals several intriguing phenomena across different network topologies.

Consistent with theoretical predictions [4], it is observed that full symmetry ($r = 1.0$) coincides with a perfect correlation between in- and out-degree sequences. As reciprocity decreases, this correlation diminishes, albeit to varying extents across network topologies. As depicted in Figure 3, the effect is more pronounced in random graphs and less noticeable in structured networks, highlighting the role of underlying topology in reciprocity-induced changes.

Interestingly, while reciprocity variations in random networks do not alter the distributions of in- and out-degrees, structured networks exhibit subtle but important changes. In small-world networks, starting from initial reciprocity of 1.0, we observe a slight broadening of both in- and out-degree distributions as unidirectional links increase. Conversely, networks with modular structures display an opposite trend: the introduction of asymmetric links leads to a sharpening of both distributions, despite preserving their overall shape.

In weighted networks, the sensitivity of in- and out-strength distributions to reciprocity adjustments appears to be more topology-dependent. Networks with modular structures demonstrate more robust distributions, while random networks exhibit greater susceptibility to changes. Specifically, in random weighted networks, adjusting reciprocity to higher values (i.e., $r_d > r_c = 0.07$) leads to a noticeable broadening of both in- and out-strength distributions, reflecting the lack of inherent structural constraints that might otherwise stabilize these distributions against perturbations.

These findings illuminate the intricate interplay between reciprocity, network structure, and degree correlations. The development of an algorithm that enables systematic manipulation of reciprocity provides a powerful tool for comprehensive numerical exploration of these relationships.

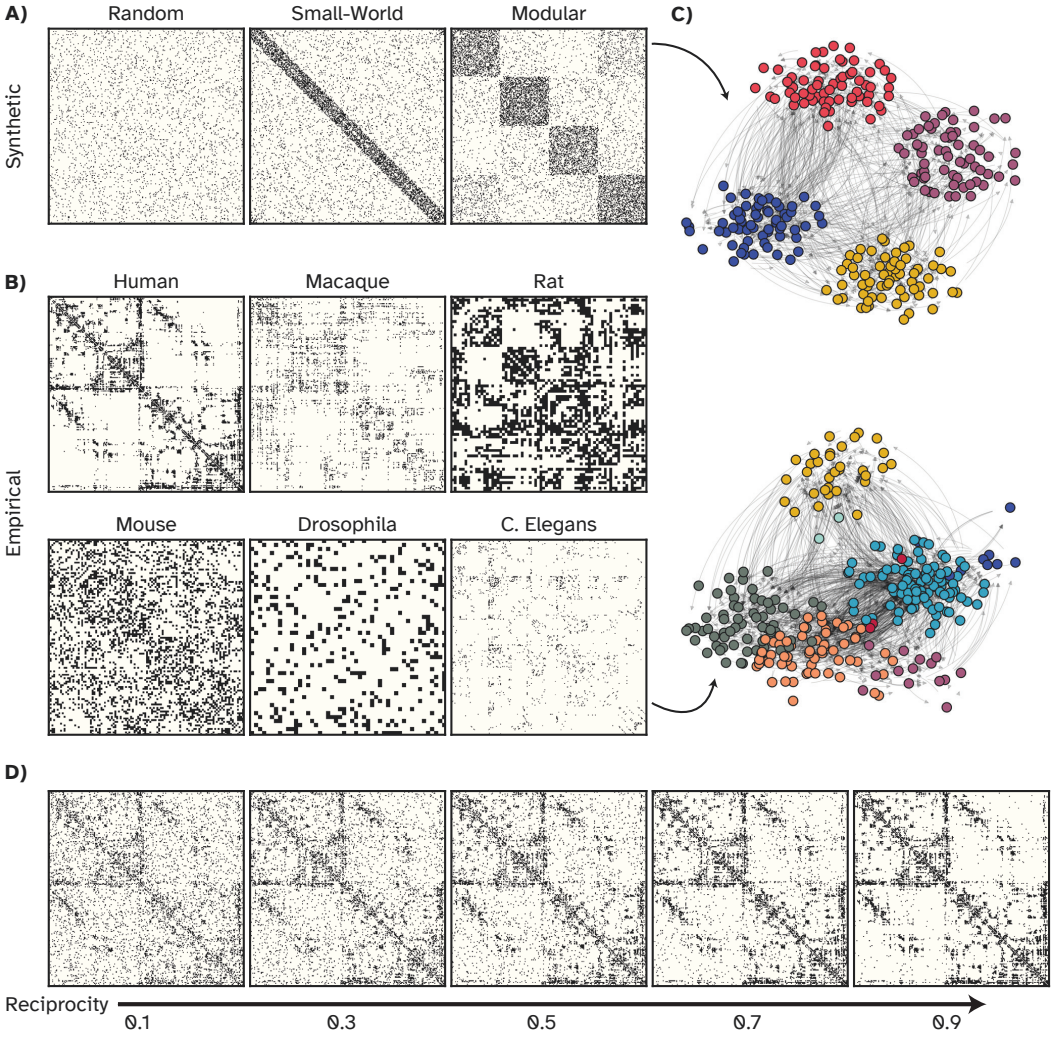

Figure 1: Representative synthetic (A) and empirical (B) connectomes used for algorithm testing. (A) The random network, initially with reciprocity $r_b = 0.07$ is shown after adjusting its reciprocity to $r = 1.0$. Small-world and modular networks are originally symmetric. Illustrated connectomes display the adjacency matrices after adjusting their reciprocity to $r_b = 1.0$ using Algorithm 1. (C) Community structures identified in two networks with clear modular organization. (D) Human connectome (Human-struct-scale125) with initial reciprocity, $r_b = 1.0$ (fully symmetric), shown alongside representations with adjusted reciprocity values, $r_b = [0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9]$, from left to right.

5.3 Alterations in path lengths and network efficiency

The impact of reciprocity variation on inter-nodal distances within the network models was investigated. This analysis involved the transformation of weight matrices into length matrices through normalization and inversion of non-zero weights, followed by the application of the Floyd-Warshall algorithm to compute shortest paths. From these computations, global efficiency metrics [36, 37] were derived by averaging inverse shortest paths, and changes in network diameters were examined.

In binary networks, the results, depicted in Figure 4, reveal a striking resilience of these metrics to reciprocity alterations, with one notable exception: the *C. elegan*s connectome. Due to its extreme sparsity and initial reciprocity of 0.21, both increases and decreases in reciprocity led to significant enhancements in global efficiency.

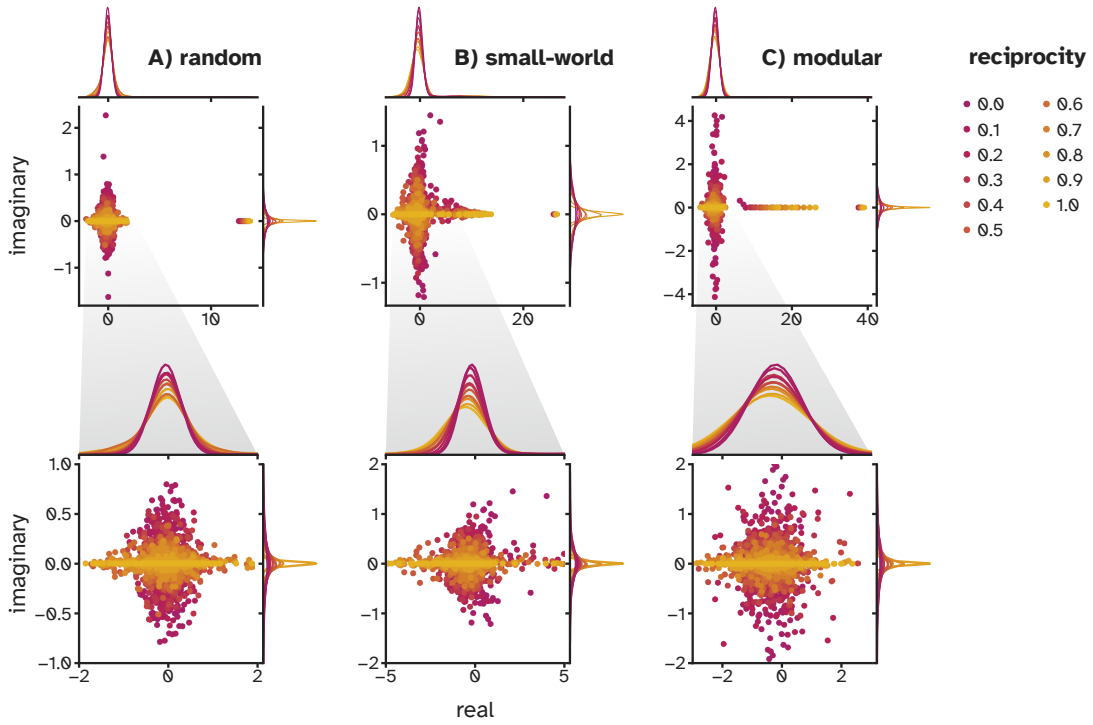

Figure 2: Change in eigenspectrum of adjacency matrices as their reciprocity is adjusted from $r_b = 0$ to $r_b = 1.0$ using Algorithm 1. The top panel shows how eigenvalues are arranged in the complex plane, while the bottom panel provides a zoomed-in view. Distributions of real and imaginary components at various degrees of reciprocity were calculated from data collected in 50 trials for each setting.

Upon application of the weighted variant of the algorithm, an increase in global efficiency was observed as the asymmetric components of human connectomes were enhanced, as shown in Figure S3. This effect was not limited to biological networks, but was also observed in randomly generated network structures.

5.4 Evolution of community structure

In this analysis, the influence of reciprocity on network community structure was investigated using the modularity index [38] as the primary metric. As shown in Figure 4, the modularity index increases consistently with rising network reciprocity, particularly in networks with an initially modular structure. The modular networks were initially constructed with complete symmetry (i.e., $r_b = 1.0$), promoting a well-defined modular arrangement. When asymmetry was introduced—by pruning reciprocal links and adding unidirectional ones—the cohesiveness of the network's community structure diminished. In contrast, random networks with low initial reciprocity (i.e., $r_b = 0.07$) exhibited a notable enhancement in the modularity index as reciprocity increased.

The effect of varying reciprocity on network community structure was further evident when the algorithm was applied to weighted random networks and empirical connectomes (see Figure S3). This effect was most pronounced in random networks, where algorithmic adjustments significantly altered the modularity index. In contrast, weighted networks with small-world and modular binary backbones showed less pronounced changes in modularity following reciprocity adjustments.

Community structures are crucial for the specificity and stability of real-world network topologies [39]. The NRC algorithms developed here systematically manipulate reciprocity, offering a valuable tool for examining how these structural features contribute to the overall dynamics and functionality of complex networks.

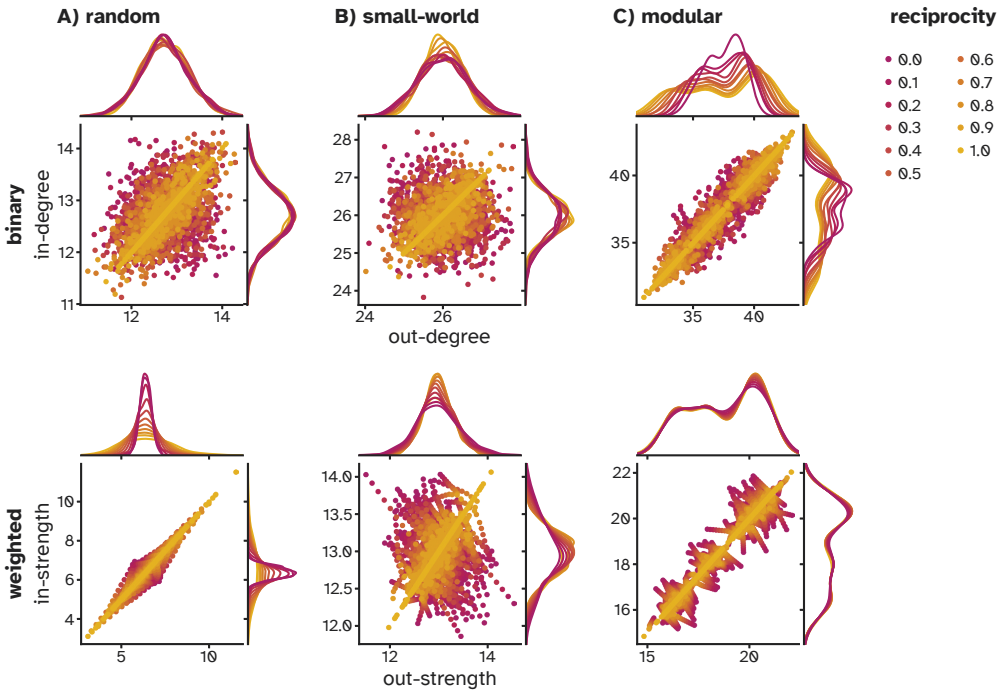

Figure 3: Evolution of degree distributions and correlation in binary (top) and weighted (bottom) adjacency matrices as their reciprocity is adjusted from $r = 0$ to $r = 1.0$ using Algorithm 1 and Algorithm 2.

5.5 Clustering coefficient variations

The average clustering coefficients [40, 41] were examined across various networks by systematically varying reciprocity. The observed trends were strongly influenced by the networks' initial reciprocity levels, indicating that the algorithm affects the relationship between reciprocity and clustering differently depending on the network configuration.

In synthetic binary networks, random graphs begin with near-zero reciprocity, while small-world and modular networks are fully symmetric. As shown in Figure 4, increasing reciprocity consistently decreases the clustering coefficient. This behavior is intuitive, as redistributing links to favor reciprocal connections reduces the probability of forming triangles, which are central for clustering coefficient calculations. A similar trend is observed in rat, mouse, and Drosophila connectomes, where the initial binary reciprocity values range between $0.5 < r_b < 1.0$. Conversely, in large human connectomes ($r_b = 1.0$) with sparse connections, redistributing links to unidirectional connections decreases triangle density. However, smaller human connectomes, with higher connection densities, show a different pattern, reflecting the variable impact of reciprocity on clustering.

In weighted networks, both synthetic and empirical (see Figure S3), the clustering coefficient appears less sensitive to reciprocity adjustments. An exception is the macaque monkey connectome, which starts with a sparse connectivity profile and initial reciprocity of $r_b = r_w = 0.51$. Here, adjustments to both increase and decrease reciprocity, as implemented in Algorithm 2, led to reductions in the clustering coefficient (Figures 4 and S3).

6 Computational implications of graded reciprocity

The computational implications of graded reciprocity were explored through the memory capacity [21] of reservoir computing models [22]. To investigate the effects of graded reciprocity, a series of experiments were conducted on binary and weighted random networks with adjusted reciprocity values. The performance of these networks was then assessed in a memory task, focusing on their ability to store and recall information. As shown in Figure 5, the influence of graded reciprocity on memory capacity (MC) was found noticeable in both binary and weighted networks, though the

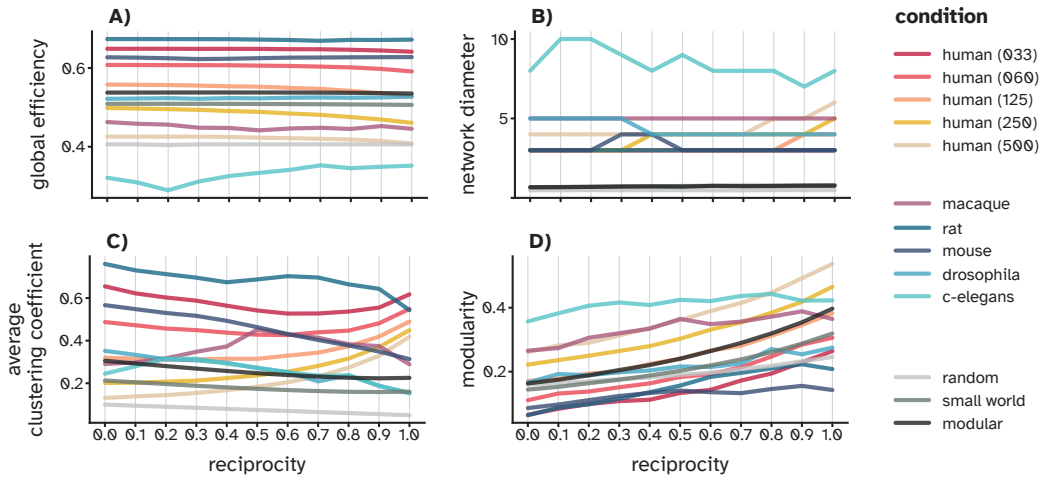

Figure 4: Trends in key network metrics—global efficiency, network diameter, modularity index, and average clustering coefficient—in binary adjacency matrices as their reciprocity is adjusted from $r = 0$ to $r = 1.0$ using Algorithm 1.

patterns observed differ. In binary networks, increasing reciprocity leads to a reduction in memory capacity, whereas in weighted networks, the opposite trend is observed, with memory capacity improving as reciprocity increases.

In binary networks, increasing reciprocity and the resulting feedback loops between nodes reduce the overall diversity of the network's states, limiting its ability to store and recall complex patterns. In fact, reciprocal connections tend to reinforce the same directions of information flow, potentially causing the recurrent network to become more deterministic and generating a poor repertoire of representations of the input stimulus. This limitation can undermine the ability of reservoir computing to create the diverse and rich representations necessary for effective computation.

On the other hand, increasing weighted reciprocity leads to the modulation of a small subset of weights, causing them to have larger values while suppressing the smaller weights even further. As a result, increasing the reciprocity of a weighted network expands the range of weight values. While the total sum of the weights remains constant, their distribution shifts, resulting in a greater disparity among individual weights. This redistribution enhances the network's memory capacity by making the system more selective in its interactions. The left panel of Figure 6 illustrates the average weight values at different levels of weighted reciprocity, computed over 50 random matrix trials. Since weight modulation introduces a confounding effect when analyzing memory capacity, MC was normalized through division by the standard deviation of non-zero weights. This normalization, as shown in the right panel of Figure 6, allows the impact of reciprocity on memory performance to be isolated. After normalization, the trend observed in the memory capacity closely resembles the one seen in the binary networks.

7 Performance and scalability

To evaluate the scalability of the reciprocity adjustment algorithm, tests were conducted on random networks with up to $2,048$ nodes and densities of $0.1,\ 0.2,\ 0.3,\ 0.4$, and 0.5. Both binary and weighted networks were initialized with reciprocity values matching respective densities.

The binary algorithm consistently completed execution in under 1 second, with computation time primarily influenced by network size and the magnitude of reciprocity adjustment, as illustrated in Figure 7. The estimated time complexity of Algorithm 1 is $\mathcal{O}(n^{2.28})$. In contrast, the weighted network algorithm, described in Algorithm 2, incorporates an iterative procedure that significantly affects performance. For $\frac{r_d}{r_c} < 1$, where r_d and r_c represent the desired and current reciprocity, the algorithm maintained high efficiency. However, when $\frac{r_d}{r_c} > 1$, a time complexity of $\mathcal{O}(e^n)$ was observed, reflecting the iterative nature of Algorithm 2. This disparity in performance between binary and weighted networks highlights the added complexity introduced by weighted networks.

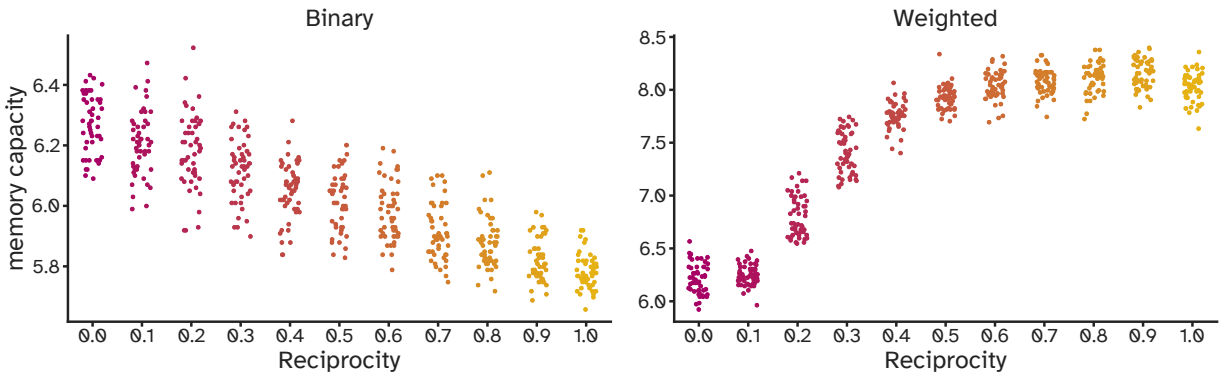

Figure 5: Trends in memory capacity of reservoir computing models with random network structures as their reciprocity is adjusted from $r = 0$ to $r = 1.0$ using Algorithm 1 and Algorithm 2. To assess the memory of a given reservoir computing model, a sequence of white noise is generated by sampling from a normal distribution with mean 0 and variance 1.0 (i.e., $u(t) \sim \mathcal{N}(0,\ 1)$). Then k independent output units are trained to predict delayed versions of the input, i.e., $d_k(t) = u(t-k)$ where in our investigations, $k = 1, 2, ..., 20$. On the test dataset, all the predictions, $d_k(t)$, are collected and the memory capacity (MC) is calculated using the squared Pearson correlation coefficient, ρ^2 as $MC = \Sigma_k \rho^2(d_k, \hat{d}_k)$. In the reservoir computing (RC) models used in this study, the hyperbolic tangent (tanh) function is employed as the transfer function, while a linear readout layer is used to map the reservoir states to the output. The input-to-reservoir weights are randomly sampled from a uniform distribution, ensuring a broad range of input connections. To satisfy the echo state property (ESP), the adjacency matrices are normalized such that their spectral radius is set to 1.0, thereby ensuring network stability and appropriate dynamics for computation over time [42].

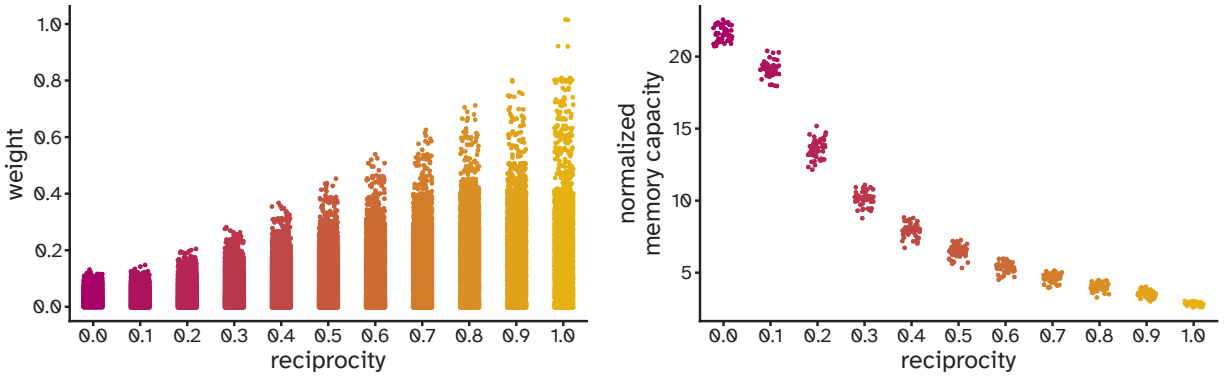

Figure 6: Weight modulation and its effect on memory capacity. Left: Average weight values at different levels of weighted reciprocity, calculated over 50 random matrix trials. Right: Memory capacity of reservoir computing models with random network structures, as reciprocity is varied from $r = 0$ to $r = 1.0$ using Algorithm 2, and subsequently normalized by dividing by the standard deviation of non-zero weights.

8 Outlook

In complex systems, directed networks are prevalent, as they effectively capture the inherent asymmetry in both natural and artificial interactions. From neural pathways to social media, these networks rely on a delicate balance between reciprocity and asymmetry, which influences performance, adaptability, and resource utilization across various domains. Despite its significance, the role of graded reciprocity is often overlooked, due to the challenge of modifying it without affecting other network properties.

We have developed novel network reciprocity control algorithms to adjust reciprocity in both binary and weighted directed networks while maintaining key characteristics such as density and total strength. This innovation provides

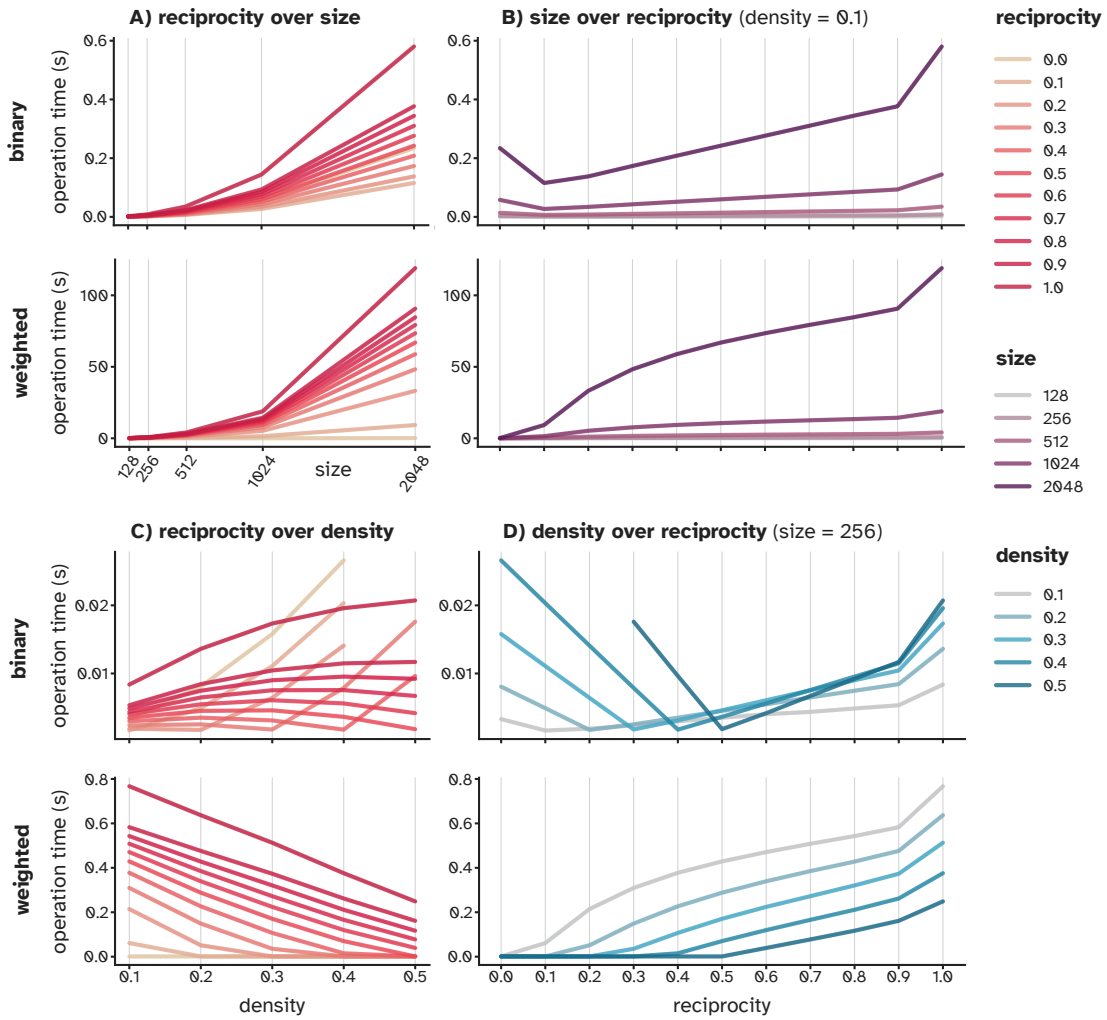

Figure 7: Scalability of Algorithm 1 and Algorithm 2 as a function of network size, density, and initial link/strength reciprocity. A, B) Computation time for both algorithms primarily depends on the network size and the extent of reciprocity adjustment. The initial random network has low link/strength reciprocity (< 0.1), requiring the redistribution of unidirectional links in Algorithm 1 and an iterative procedure in Algorithm 2. As network size increases, the number of links and elements requiring redistribution (Algorithm 1) or weight manipulation (Algorithm 2) grows, leading to longer computation times. C) In both binary and weighted networks initialized with reciprocity matching their density, this reciprocity(density) value represents a breakpoint in the piecewise linear trends of operation time. For a fixed number of nodes ($N = 256$), increasing density raises the number of links to be redistributed for higher reciprocity, thus increasing operation time in Algorithm 1. In weighted networks, higher density results in a greater number of non-zero elements in the adjacency matrix, allowing Algorithm 2 to operate on more values and converge faster. D) Computation time follows a piecewise relationship with the magnitude of reciprocity change. Breakpoints correspond to the original reciprocity, with segments on the left representing the time required to reduce reciprocity and those on the right showing the time needed to increase it in both algorithms.

a practical tool for exploring how reciprocity impacts network dynamics in fields such as complex systems physics, neuroscience, and bio-inspired computing.

Our method allows researchers to use reciprocity as a control parameter in the investigation of criticality, phase transitions, and self-organized behavior in networked systems. The approach also generates null models for more robust statistical analyses, particularly in computational neuroscience. In neuroscience, where symmetric connectome data

may obscure underlying asymmetries [43], our approach facilitates the exploration of synthetic reciprocity, offering insights into the optimal balance for neural processing.

In artificial intelligence, our algorithm could enhance recurrent neural networks (RNNs), potentially improving training efficiency by fine-tuning the reciprocity-asymmetry balance. We demonstrated its broad applicability by examining how reciprocity influences network features, such as degree distributions, spectral properties, modularity, and clustering. The scalability of the NRC approach ensures its relevance to large-scale networks, providing deeper insights into the organization and dynamics of complex systems.

In sum, this work provides a tool for network hypothesis testing and exploration, offering new insights into understanding and optimizing binary and weighted directed network structures across various scientific fields.

9 Data and code availability

Data supporting the findings of this study are available publicly. The connectome dataset is available at netneurotools `https://netneurotools.readthedocs.io`. Synthetic data and codes that were used to implement the algorithm and perform network analysis in this manuscript are available via GitHub: `https://github.com/m00rcheh/NRC_binary_and_weighted_Network_Reciprocity_Control`.

10 Acknowledgement

The funding is gratefully acknowledged: F.H: DFG TRR169-A2, K.F: German Research Foundation (DFG)-SFB 936-178316478-A1; TRR169-A2; SPP 2041/GO 2888/2-2 and the Templeton World Charity Foundation, Inc. (funder DOI 501100011730) under grant TWCF-2022-30510. C.H: SFB 936-178316478-A1; TRR169-A2; SFB 1461/A4; SPP 1212 2041/HI 1286/7-1, the Human Brain Project, EU (SGA2, SGA3). We thank Arnaud Messé for valuable feedback and for reviewing the manuscript.

References

[1] Marián Boguñá and M Ángeles Serrano. Generalized percolation in random directed networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 72(1):016106, 2005.

[2] Diego Garlaschelli and Maria I Loffredo. Multispecies grand-canonical models for networks with reciprocity. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 73(1):015101, 2006.

[3] Louis M Pecora, Francesco Sorrentino, Aaron M Hagerstrom, Thomas E Murphy, and Rajarshi Roy. Cluster synchronization and isolated desynchronization in complex networks with symmetries. *Nature communications*, 5(1):4079, 2014.

[4] Antoine Allard, M Ángeles Serrano, and Marián Boguñá. Geometric description of clustering in directed networks. *Nature Physics*, 20(1):150–156, 2024.

[5] Ueli Rutishauser, Jean-Jacques Slotine, and Rodney Douglas. Computation in dynamically bounded asymmetric systems. *PLoS computational biology*, 11(1):e1004039, 2015.

[6] Everton S Medeiros, Ulrike Feudel, and Anna Zakharova. Asymmetry-induced order in multilayer networks. *Physical Review E*, 104(2):024302, 2021.

[7] Samuel Johnson. Digraphs are different: Why directionality matters in complex systems. *Journal of Physics: Complexity*, 1(1):015003, 2020.

[8] Malbor Asllani, Renaud Lambiotte, and Timoteo Carletti. Structure and dynamical behavior of non-normal networks. *Science advances*, 4(12):eaau9403, 2018.

[9] Gorka Zamora-López, Vinko Zlatić, Changsong Zhou, Hrvoje Štefančić, and Jürgen Kurths. Reciprocity of networks with degree correlations and arbitrary degree sequences. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 77(1):016106, 2008.

[10] Samuel Johnson, Virginia Domínguez-García, Luca Donetti, and Miguel A Munoz. Trophic coherence determines food-web stability. *Proceedings of the National Academy of Sciences*, 111(50):17923–17928, 2014.

[11] Sebastian Ocklenburg and Annakarina Mundorf. Symmetry and asymmetry in biological structures. *Proceedings of the National Academy of Sciences*, 119(28):e2204881119, 2022.

[12] Takashi Nishikawa and Adilson E Motter. Symmetric states requiring system asymmetry. *Physical review letters*, 117(11):114101, 2016.

[13] Joseph D Hart, Yuanzhao Zhang, Rajarshi Roy, and Adilson E Motter. Topological control of synchronization patterns: Trading symmetry for stability. *Physical review letters*, 122(5):058301, 2019.

[14] Vinko Zlatic and Hrvoje Štefancic. Model of wikipedia growth based on information exchange via reciprocal arcs. *Europhysics letters*, 93(5):58005, 2011.

[15] Ronghua Xu, Qingpeng Zhang, and Sidie Tan. The formation of reciprocal social support in online support groups: a network modeling approach. *IEEE Transactions on Computational Social Systems*, 10(6):3370–3384, 2022.

[16] Vinko Zlatić and Hrvoje Štefančić. Influence of reciprocal edges on degree distribution and degree correlations. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 80(1):016117, 2009.

[17] Yu-Xiao Zhu, Xiao-Guang Zhang, Gui-Quan Sun, Ming Tang, Tao Zhou, and Zi-Ke Zhang. Influence of reciprocal links in social networks. *PloS one*, 9(7):e103007, 2014.

[18] Natalia J Martinez, Maria C Ow, M Inmaculada Barrasa, Molly Hammell, Reynaldo Sequerra, Lynn Doucette-Stamm, Frederick P Roth, Victor R Ambros, and Albertha JM Walhout. A c. elegans genome-scale microrna network contains composite feedback motifs with high flux capacity. *Genes & development*, 22(18):2535–2549, 2008.

[19] Wenzhe Ma, Ala Trusina, Hana El-Samad, Wendell A Lim, and Chao Tang. Defining network topologies that can achieve biochemical adaptation. *Cell*, 138(4):760–773, 2009.

[20] Xing Yao, Yongzhong Zhang, Rizwana Yasmeen, and Zhen Cai. The impact of preferential trade agreements on bilateral trade: A structural gravity model analysis. *PloS one*, 16(3):e0249118, 2021.

[21] Fabrizio Damicelli, Claus C Hilgetag, and Alexandros Goulas. Brain connectivity meets reservoir computing. *PLoS computational biology*, 18(11):e1010639, 2022.

[22] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science*, 304(5667):78–80, 2004.

[23] Diego Garlaschelli and Maria I Loffredo. Patterns of link reciprocity in directed networks. *Physical review letters*, 93(26):268701, 2004.

[24] Tiziano Squartini, Francesco Picciolo, Franco Ruzzenenti, and Diego Garlaschelli. Reciprocity of weighted networks. *Scientific reports*, 3(1):2729, 2013.

[25] Gorka Zamora-López, Changsong Zhou, Vinko Zlatić, and Jürgen Kurths. The generation of random directed networks with prescribed 1-node and 2-node degree correlations. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224006, 2008.

[26] M Ángeles Serrano, Ana Maguitman, Marián Boguñá, Santo Fortunato, and Alessandro Vespignani. Decoding the structure of the www: A comparative analysis of web crawls. *ACM Transactions on the Web (TWEB)*, 1(2):10–es, 2007.

[27] Vladimir Batagelj and Ulrik Brandes. Efficient generation of large random networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 71(3):036113, 2005.

[28] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[29] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

[30] Lav R Varshney, Beth L Chen, Eric Paniagua, David H Hall, and Dmitri B Chklovskii. Structural properties of the caenorhabditis elegans neuronal network. *PLoS computational biology*, 7(2):e1001066, 2011.

[31] Ann-Shyn Chiang, Chih-Yung Lin, Chao-Chun Chuang, Hsiu-Ming Chang, Chang-Huain Hsieh, Chang-Wei Yeh, Chi-Tin Shih, Jian-Jheng Wu, Guo-Tzau Wang, Yung-Chang Chen, et al. Three-dimensional reconstruction of brain-wide wiring networks in drosophila at single-cell resolution. *Current biology*, 21(1):1–11, 2011.

[32] Alessandra Griffa, Yasser Alemán-Gómez, and Patric Hagmann. Structural and functional connectome from 70 young healthy adults [data set]. *Zenodo*, 2019.

[33] Dharmendra S Modha and Raghavendra Singh. Network architecture of the long-distance pathways in the macaque brain. *Proceedings of the National Academy of Sciences*, 107(30):13485–13490, 2010.

[34] Mikail Rubinov, Rolf JF Ypma, Charles Watson, and Edward T Bullmore. Wiring cost and topological participation of the mouse brain connectome. *Proceedings of the National Academy of Sciences*, 112(32):10032–10037, 2015.

[35] Mihail Bota, Olaf Sporns, and Larry W Swanson. Architecture of the cerebral cortical association connectome underlying cognition. *Proceedings of the National Academy of Sciences*, 112(16):E2093–E2101, 2015.

[36] Caio Seguin, Martijn P Van Den Heuvel, and Andrew Zalesky. Navigation of brain networks. *Proceedings of the National Academy of Sciences*, 115(24):6297–6302, 2018.

[37] Vito Latora and Massimo Marchiori. Efficient behavior of small-world networks. *Physical review letters*, 87(19):198701, 2001.

[38] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 70(6):066111, 2004.

[39] Sergei Maslov and Kim Sneppen. Specificity and stability in topology of protein networks. *Science*, 296(5569):910–913, 2002.

[40] Jari Saramäki, Mikko Kivelä, Jukka-Pekka Onnela, Kimmo Kaski, and Janos Kertesz. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 75(2):027105, 2007.

[41] Marcus Kaiser. Mean clustering coefficients: the role of isolated nodes and leafs on clustering measures for small-world networks. *New Journal of Physics*, 10(8):083042, 2008.

[42] Herbert Jaeger. Adaptive nonlinear system identification with echo state networks. *Advances in neural information processing systems*, 15, 2002.

[43] Laura E Suarez, Yossi Yovel, Martijn P van den Heuvel, Olaf Sporns, Yaniv Assaf, Guillaume Lajoie, and Bratislav Misic. A connectomics-based taxonomy of mammals. *Elife*, 11:e78635, 2022.