Exploiting symmetries in network analysis

Ruben Sanchez-Garcia

Mathematical Sciences, University of Southampton

Seventeenth Mathematics of Networks meeting (MoN17)

University of Sheffield

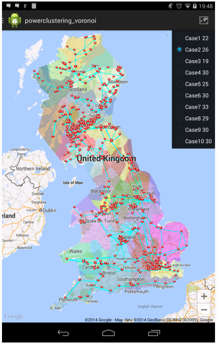
20 September 2018

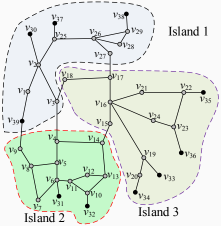
A brief introduction...

- background in pure mathematics (algebraic topology)

A brief introduction. . .

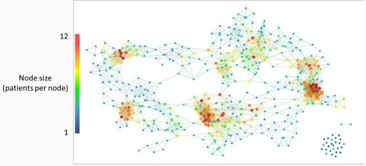
- background in pure mathematics (algebraic topology)
- past & current projects in networks:
  - cascading failures in power grid (spectral clustering)

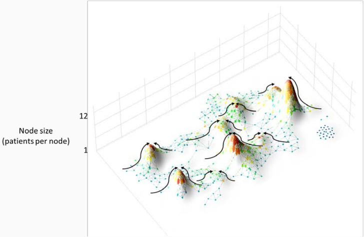




A brief introduction. . .

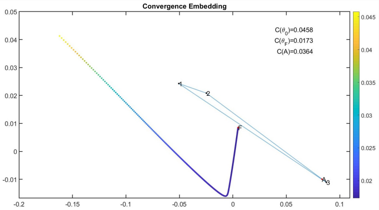
- background in pure mathematics (algebraic topology)
- past & current projects in networks:
  - cascading failures in power grid (spectral clustering)
  - Morse-theoretic clustering of annotated network data [asthma phenotypes]

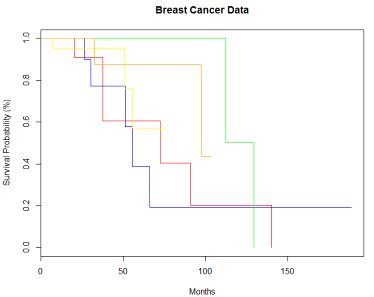




A brief introduction. . .

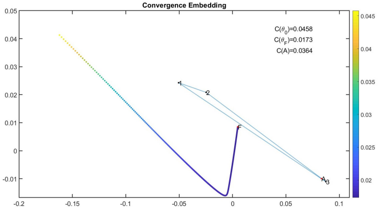
- background in pure mathematics (algebraic topology)
- past & current projects in networks:
  - cascading failures in power grid (spectral clustering)
  - Morse-theoretic clustering of annotated network data [asthma phenotypes]
  - information-theoretic network data integration ['omics cancer data]

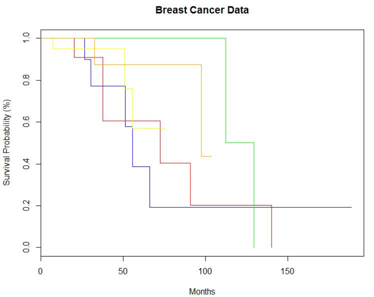




A brief introduction. . .

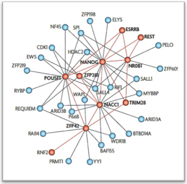
- background in pure mathematics (algebraic topology)
- past & current projects in networks:
  - cascading failures in power grid (spectral clustering)
  - Morse-theoretic clustering of annotated network data [asthma phenotypes]
  - information-theoretic network data integration ['omics cancer data]
  - network symmetry. . .

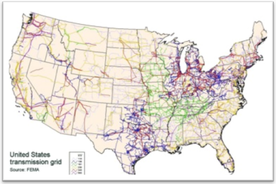


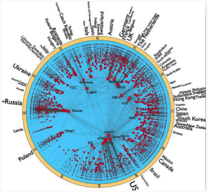


Real-world complex networks

- Network models of real-world complex systems (biological, social, technological, . . .)



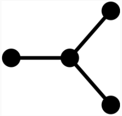




- Successful approach
  - simple & versatile
  - reveals structural and dynamical properties
  - common properties/principles e.g. symmetry

Network symmetry

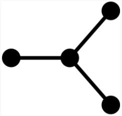
- Network symmetries identify structurally equivalent nodes (and links)



- Mathematically, network symmetries are *graph automorphisms*: permutations of the vertices (nodes) $\sigma\colon V \to V$ preserving adjacency

Network symmetry

- Network symmetries identify structurally equivalent nodes (and links)



- Mathematically, network symmetries are *graph automorphisms*: permutations of the vertices (nodes) $\sigma \colon V \to V$ preserving adjacency

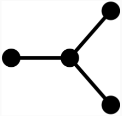
$$(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \quad \forall i,j \in V$$

(Here $\mathcal{G} = (V, E)$ graph with vertex set V and edge set $E \subseteq V \times V$)

- They form a group $\mathrm{Aut}(\mathcal{G})$

Network symmetry

- Network symmetries identify structurally equivalent nodes (and links)



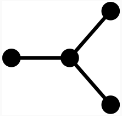
- Mathematically, network symmetries are *graph automorphisms*: permutations of the vertices (nodes) $\sigma\colon V \to V$ preserving adjacency

$$(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \quad \forall i,j \in V$$

  (Here $\mathcal{G} = (V, E)$ graph with vertex set V and edge set $E \subseteq V \times V$)
- They form a group $\mathrm{Aut}(\mathcal{G})$
- They reflect structural redundancies on the underlying system, thus relate to system robustness

Network symmetry

- Network symmetries identify structurally equivalent nodes (and links)



- Mathematically, network symmetries are *graph automorphisms*: permutations of the vertices (nodes) $\sigma \colon V \to V$ preserving adjacency

$$(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \quad \forall i,j \in V$$

  (Here $\mathcal{G} = (V, E)$ graph with vertex set V and edge set $E \subseteq V \times V$)
- They form a group $\mathrm{Aut}(\mathcal{G})$
- They reflect structural redundancies on the underlying system, thus relate to system robustness
- They may arise from replicative growth processes, evolution from basic principles or functional optimisation

Are real-world networks symmetric?

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

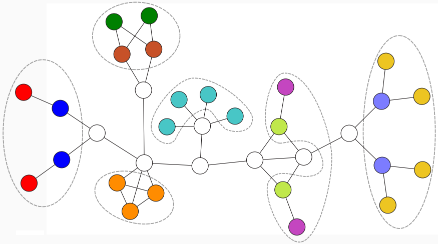
Are real-world networks symmetric?

- Large automorphism group (10^{407} to $10^{197,552}$ in our test networks)

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Are real-world networks symmetric?

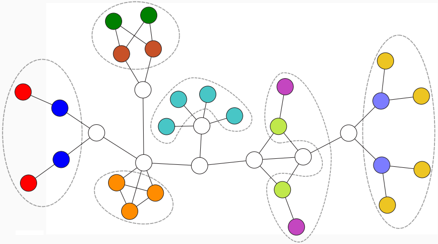
- Large automorphism group (10^{407} to $10^{197,552}$ in our test networks)
- Generated at small subgraphs (*symmetric motifs*) made of orbits of structurally equivalent nodes



[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Are real-world networks symmetric?

- Large automorphism group (10^{407} to $10^{197,552}$ in our test networks)
- Generated at small subgraphs (*symmetric motifs*) made of orbits of structurally equivalent nodes

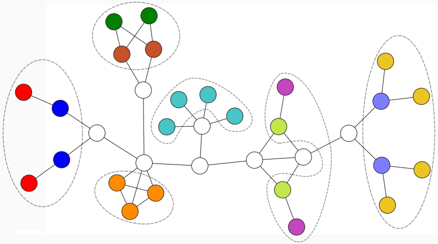


$|\mathrm{Aut}(\mathcal{G})| = 4{,}608$

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Are real-world networks symmetric?

- Large automorphism group (10^{407} to $10^{197{,}552}$ in our test networks)
- Generated at small subgraphs (*symmetric motifs*) made of orbits of structurally equivalent nodes

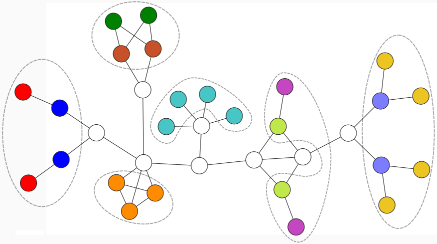


$|\mathrm{Aut}(\mathcal{G})| = 4{,}608$ but 11 generating permutations

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Are real-world networks symmetric?

- Large automorphism group (10^{407} to $10^{197,552}$ in our test networks)
- Generated at small subgraphs (*symmetric motifs*) made of orbits of structurally equivalent nodes



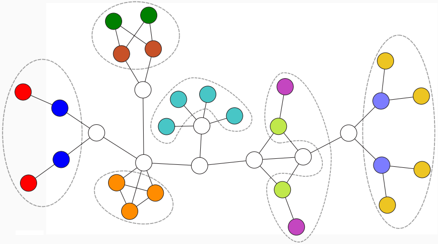
$|\mathrm{Aut}(\mathcal{G})| = 4{,}608$ but 11 generating permutations

- Typical symmetric motif ($>90\%$): orbits of same size with every permutation in each orbit realised — *basic* symmetric motifs (BSMs)

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Are real-world networks symmetric?

- Large automorphism group (10^{407} to $10^{197,552}$ in our test networks)
- Generated at small subgraphs (*symmetric motifs*) made of orbits of structurally equivalent nodes



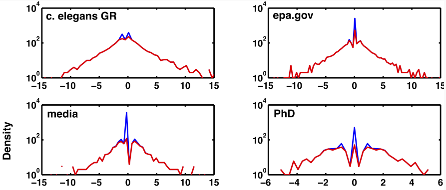
$$|\mathrm{Aut}(\mathcal{G})| = 4{,}608 \text{ but } 11 \text{ generating permutations}$$

- Typical symmetric motif ($>90\%$): orbits of same size with every permutation in each orbit realised — *basic* symmetric motifs (BSMs)
- Non-basic symmetric motifs typically branched trees

[MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)]

Redundant spectrum

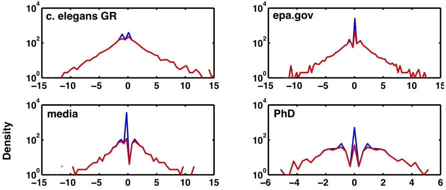
- Symmetry explains most of the discrete part of the spectrum ('peaks' in the spectral density) of the adjacency matrix



[MacArthur, Sanchez-Garcia *Spectral characteristics of network redundancy* **Phys. Rev. E** (2009)]

Redundant spectrum

- Symmetry explains most of the discrete part of the spectrum ('peaks' in the spectral density) of the adjacency matrix

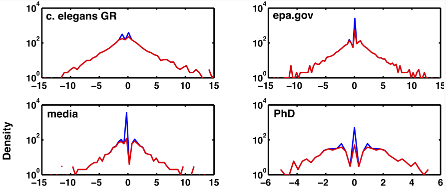


- The redundant spectrum is generated at the symmetric motifs (*localised* eigenvectors)

[MacArthur, Sanchez-Garcia *Spectral characteristics of network redundancy* **Phys. Rev. E** (2009)]

Redundant spectrum

- Symmetry explains most of the discrete part of the spectrum ('peaks' in the spectral density) of the adjacency matrix



- The redundant spectrum is generated at the symmetric motifs (*localised* eigenvectors)
- Most symmetric motifs are basic and we can predict their contribution to the discrete spectrum e.g.

$$\mathrm{RSpec}_1 = \{-1, 0\} \qquad \mathrm{RSpec}_2 = \{-2, -\varphi, -1, 0, \varphi - 1, 1\}$$

[MacArthur, Sanchez-Garcia *Spectral characteristics of network redundancy* **Phys. Rev. E** (2009)]

Graph automorphisms

- Let $\mathcal{G} = (V, E)$ be a graph with adjacency matrix $A = (a_{ij})$.
- A *graph automorphisms* is a permutation of the vertices (nodes) $\sigma\colon V \to V$ preserving adjacency

$$(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \qquad \forall i,j \in V$$

Graph automorphisms

- Let $\mathcal{G} = (V, E)$ be a graph with adjacency matrix $A = (a_{ij})$.
- A *graph automorphisms* is a permutation of the vertices (nodes) $\sigma\colon V \to V$ preserving adjacency

$$(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \qquad \forall i,j \in V$$
$$AP = PA$$

where P is the permutation matrix representing σ ($[P]_{ij} = 1$ if $\sigma(i) = j$ and 0 otherwise).

Graph automorphisms

- Let $\mathcal{G} = (V, E)$ be a graph with adjacency matrix $A = (a_{ij})$.
- A *graph automorphisms* is a permutation of the vertices (nodes) $\sigma\colon V \to V$ preserving adjacency

$$
\begin{aligned}
(i,j) \in E \iff (\sigma(i), \sigma(j)) \in E \qquad\qquad & \forall i,j \in V \\
AP = PA &
\end{aligned}
$$

  where P is the permutation matrix representing σ ($[P]_{ij} = 1$ if $\sigma(i) = j$ and 0 otherwise).
- In particular, automorphisms generate high-multiplicity eigenvalues: $Av = \lambda v$ implies $APv = PAv = \lambda Pv$.

Network measures

Crucial observation

Network symmetries are inherited by any measure on the network that depends on structure alone

Crucial observation

Network symmetries are inherited by any measure on the network that depends on structure alone

- A pairwise network measure $F\colon V \times V \to \mathbf{R}$ is *structural* if $F(i,j) = F(\sigma(i),\sigma(j))$ for all $\sigma \in \mathrm{Aut}(\mathcal{G})$ and all $i,j \in V$.
  (That is, F depends on the network combinatorial structure alone.)

Network measures

Crucial observation

Network symmetries are inherited by any measure on the network that depends on structure alone

- A pairwise network measure $F \colon V \times V \to \mathbf{R}$ is *structural* if $F(i,j) = F(\sigma(i), \sigma(j))$ for all $\sigma \in \mathrm{Aut}(\mathcal{G})$ and all $i, j \in V$.
  (That is, F depends on the network combinatorial structure alone.)
- Examples: network metrics (e.g. shortest path, resistance)

Network measures

Crucial observation

Network symmetries are inherited by any measure on the network that depends on structure alone

- A pairwise network measure $F\colon V \times V \to \mathbf{R}$ is *structural* if $F(i,j) = F(\sigma(i), \sigma(j))$ for all $\sigma \in \mathrm{Aut}(\mathcal{G})$ and all $i, j \in V$.
  (That is, F depends on the network combinatorial structure alone.)
- Examples: network metrics (e.g. shortest path, resistance), matrices derived from the adjacency matrix (e.g. graph Laplacian)
  (We identify pairwise functions F with matrices $[M]_{ij} = F(i,j)$)

Network measures

Crucial observation

Network symmetries are inherited by any measure on the network that depends on structure alone

- A pairwise network measure $F \colon V \times V \to \mathbf{R}$ is *structural* if $F(i,j) = F(\sigma(i), \sigma(j))$ for all $\sigma \in \mathrm{Aut}(\mathcal{G})$ and all $i, j \in V$.
  (That is, F depends on the network combinatorial structure alone.)
- Examples: network metrics (e.g. shortest path, resistance), matrices derived from the adjacency matrix (e.g. graph Laplacian)
  (We identify pairwise functions F with matrices $[M]_{ij} = F(i,j)$)

Note Our results can be adapted to the presence of vertex/edge labels by restricting to automorphism preserving the additional structure

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i, j)$.

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i,j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i,j)$.
- This is (typically) an all-to-all weighted network on the same vertex set.

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i, j)$.
- This is (typically) an all-to-all weighted network on the same vertex set. However, $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$

  $$F(i, j) = F(\sigma(i), \sigma(j)) \quad \forall i, j \in V \iff F(A)P = PF(A)$$

  ($\sigma \in \mathrm{Aut}(\mathcal{G})$, P permutation matrix representing σ)

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i, j)$.
- This is (typically) an all-to-all weighted network on the same vertex set. However, $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$

$$F(i, j) = F(\sigma(i), \sigma(j)) \quad \forall i, j \in V \iff F(A)P = PF(A)$$

($\sigma \in \mathrm{Aut}(\mathcal{G})$, P permutation matrix representing σ)

- In particular, $F(\mathcal{G})$ has the same symmetric motifs and orbits as $\mathcal{G}$.

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i, j)$.
- This is (typically) an all-to-all weighted network on the same vertex set. However, $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$

$$F(i, j) = F(\sigma(i), \sigma(j)) \quad \forall i, j \in V \iff F(A)P = PF(A)$$

  ($\sigma \in \mathrm{Aut}(\mathcal{G})$, P permutation matrix representing σ)
- In particular, $F(\mathcal{G})$ has the same symmetric motifs and orbits as $\mathcal{G}$.
- This can be exploited in practice, for an arbitrary F,
  - quantify & eliminate redundancies (compression) *up to 74%*
  - computational reduction *up to 89%*
  - spectral signatures of symmetry in $F(A)$ *redundant eigenvalues*

Network representation of a network measure

- Any $n \times n$ real matrix $A = (a_{ij})$ is the adjacency matrix of a weighted graph: edge (i, j) weighted $a_{ij} \neq 0$, no such edge if $a_{ij} = 0$.
- Hence we can encode a network measure F on $\mathcal{G}$ as a graph $F(\mathcal{G})$ with adjacency matrix $[F(A)]_{ij} = F(i, j)$.
- This is (typically) an all-to-all weighted network on the same vertex set. However, $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$

$$F(i,j) = F(\sigma(i), \sigma(j)) \quad \forall i, j \in V \iff F(A)P = PF(A)$$

  ($\sigma \in \mathrm{Aut}(\mathcal{G})$, P permutation matrix representing σ)
- In particular, $F(\mathcal{G})$ has the same symmetric motifs and orbits as $\mathcal{G}$.
- This can be exploited in practice, for an arbitrary F,
  - quantify & eliminate redundancies (compression) *up to 74%*
  - computational reduction *up to 89%*
  - spectral signatures of symmetry in $F(A)$ *redundant eigenvalues*

[Sanchez-Garcia *Exploiting symmetries in network analysis* arxiv preprint 1803.06915]

Full and sparse network measures

Our results apply to arbitrary network measures, however some will be more useful for either *full* or *sparse* measures.

full measure $F(i,j) \neq 0$ for most $i,j \in V$ (e.g. graph metric)

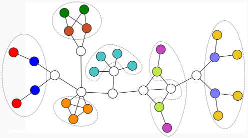
sparse measure $F(i,j) = 0$ if $a_{ij} = 0$ for most $i,j \in V$ (e.g. graph Laplacian)

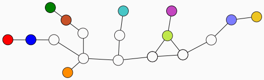
Network quotient

- Formal procedure to eliminate redundancies/symmetries.

Network quotient

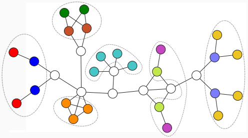
- Formal procedure to eliminate redundancies/symmetries.

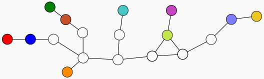




Network quotient

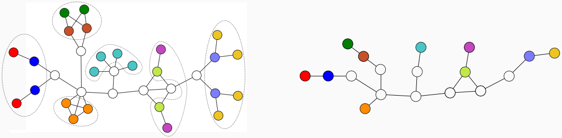
- Formal procedure to eliminate redundancies/symmetries.
- Let $\mathcal{G} = (V, E)$ with $n \times n$ adjacency matrix $A = (a_{ij})$ and orbit partition $V = V_1 \cup \ldots \cup V_m$.





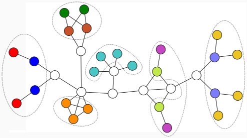
Network quotient

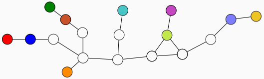
- Formal procedure to eliminate redundancies/symmetries.
- Let $\mathcal{G} = (V, E)$ with $n \times n$ adjacency matrix $A = (a_{ij})$ and orbit partition $V = V_1 \cup \ldots \cup V_m$.
- The *quotient graph* $\mathcal{Q}$ is the graph with adjacency matrix $Q(A) = (b_{kl})$ given by $b_{kl} = \frac{1}{|V_k|} \sum_{i \in V_k, j \in V_l} a_{ij}$ (average connectivity from a vertex in V_k to vertices in V_l).



Network quotient

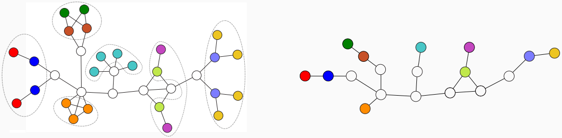
- Formal procedure to eliminate redundancies/symmetries.
- Let $\mathcal{G} = (V, E)$ with $n \times n$ adjacency matrix $A = (a_{ij})$ and orbit partition $V = V_1 \cup \ldots \cup V_m$.
- The *quotient graph* $\mathcal{Q}$ is the graph with adjacency matrix $Q(A) = (b_{kl})$ given by $b_{kl} = \frac{1}{|V_k|} \sum_{i \in V_k, j \in V_l} a_{ij}$ (average connectivity from a vertex in V_k to vertices in V_l).
- Matrix equation: $Q(A) = \Lambda^{-1} S^T A S$ where S is the $n \times m$ *characteristic matrix* of the partition ($[S]_{ik} = 1$ if $i \in V_k$) and $\Lambda = \mathrm{diag}(n_1, \ldots, n_m)$, $n_k = |V_k|$.





Network quotient

- Formal procedure to eliminate redundancies/symmetries.
- Let $\mathcal{G} = (V, E)$ with $n \times n$ adjacency matrix $A = (a_{ij})$ and orbit partition $V = V_1 \cup \ldots \cup V_m$.
- The *quotient graph* $\mathcal{Q}$ is the graph with adjacency matrix $Q(A) = (b_{kl})$ given by $b_{kl} = \frac{1}{|V_k|} \sum_{i \in V_k, j \in V_l} a_{ij}$ (average connectivity from a vertex in V_k to vertices in V_l).
- Matrix equation: $Q(A) = \Lambda^{-1} S^T A S$ where S is the $n \times m$ *characteristic matrix* of the partition ($[S]_{ik} = 1$ if $i \in V_k$) and $\Lambda = \mathrm{diag}(n_1, \ldots, n_m)$, $n_k = |V_k|$.
- The graph $\mathcal{Q}$ is directed, weighted with loops, but spectrally equivalent to an undirected graph, if $\mathcal{G}$ undirected too.



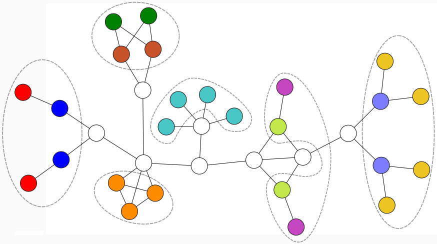
(Left) Toy network (Right) Quotient skeleton (no loops, edge directions or weights)

Internal and external edges

Internal edges between vertices in the same symmetric motif

External edges between vertices in different symmetric motifs

(fixed points treated as trivial symmetric motifs of size 1)

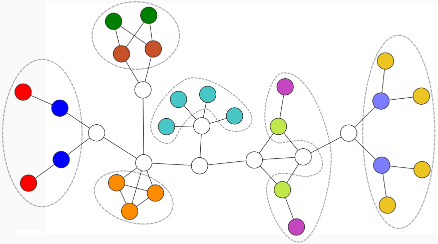


Internal and external edges

Internal edges between vertices in the same symmetric motif

External edges between vertices in different symmetric motifs

(fixed points treated as trivial symmetric motifs of size 1)



- The vast majority of edges in $F(\mathcal{G})$ are external (typically over 90% in the sparse, and 99.99% in the full, case)

Symmetry compression

- $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$ as redundancies, namely repeated values $F(\sigma(i), \sigma(j)) = F(i, j)$.
- We can use the quotient to eliminate the symmetry-induced redundancies.
- Compression ratios between $m_{\mathcal{Q}}/m_{\mathcal{G}}$ (sparse) and $(n_{\mathcal{Q}}/n_{\mathcal{G}})^2$ (full).

Symmetry compression

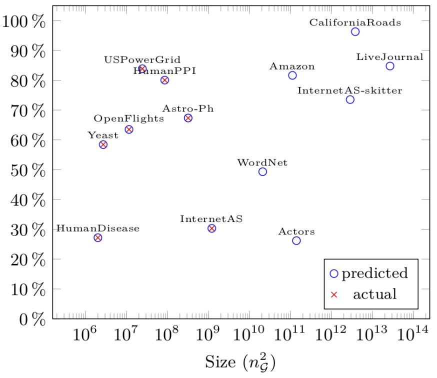
- $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$ as redundancies, namely repeated values $F(\sigma(i), \sigma(j)) = F(i,j)$.
- We can use the quotient to eliminate the symmetry-induced redundancies.
- Compression ratios between $m_{\mathcal{Q}}/m_{\mathcal{G}}$ (sparse) and $(n_{\mathcal{Q}}/n_{\mathcal{G}})^2$ (full).

Table 2. Symmetry in some real-world networks.

Name	$n_{\mathcal{G}}$	$m_{\mathcal{G}}$	gen	t_1	t_2	sm	bsm	mv	$\tilde{n}_{\mathcal{Q}}$
HumanDisease	1,419	2,738	713	0.00	0.16	272	96.0	71.0	48.3
Yeast	1,647	2,736	380	0.00	0.01	149	99.3	33.3	76.3
OpenFlights	3,397	19,230	732	0.00	0.11	321	93.5	32.4	77.3
USPowerGrid	4,941	6,594	414	0.00	0.09	302	97.4	16.7	90.2
HumanPPI	9,270	36,918	972	0.00	0.12	437	100	15.3	89.5
Astro-Ph	17,903	196,972	3,232	0.01	0.21	1,682	99.4	27.5	81.9
InternetAS	34,761	107,720	15,587	0.03	0.29	3,189	99.9	54.3	55.0
WordNet	145,145	656,230	52,152	0.18	0.62	28,456	92.0	60.0	60.1
Amazon	334,863	925,872	32,098	0.20	0.39	23,302	99.8	16.8	90.3
Actors	374,511	15,014,839	182,803	0.95	1.38	36,703	99.9	58.6	51.2
InternetAS-skitter	1,694,616	11,094,209	319,738	1.71	4.17	84,675	99.1	19.7	85.4
CaliforniaRoads	1,957,027	2,760,388	36,430	0.47	0.16	35,210	98.8	4.0	97.9
LiveJournal	5,189,808	48,687,945	410,575	8.02	3.59	245,211	99.9	12.7	92.1

Symmetry compression

- $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$ as redundancies, namely repeated values $F(\sigma(i), \sigma(j)) = F(i,j)$.
- We can use the quotient to eliminate the symmetry-induced redundancies.
- Compression ratios between $m_{\mathcal{Q}}/m_{\mathcal{G}}$ (sparse) and $(n_{\mathcal{Q}}/n_{\mathcal{G}})^2$ (full).



Symmetry compression

- $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$ as redundancies, namely repeated values $F(\sigma(i), \sigma(j)) = F(i, j)$.
- We can use the quotient to eliminate the symmetry-induced redundancies.
- Compression ratios between $m_{\mathcal{Q}}/m_{\mathcal{G}}$ (sparse) and $(n_{\mathcal{Q}}/n_{\mathcal{G}})^2$ (full).

- A simple compression/decompression algorithm recovers external edges plus averaged internal edges (**average compression**)
- Annotating the quotient, we can achieve **lossless compression** with a (not so simple) algorithm

Symmetry compression

- $F(\mathcal{G})$ inherits all the symmetries of $\mathcal{G}$ as redundancies, namely repeated values $F(\sigma(i), \sigma(j)) = F(i, j)$.
- We can use the quotient to eliminate the symmetry-induced redundancies.
- Compression ratios between $m_{\mathcal{Q}}/m_{\mathcal{G}}$ (sparse) and $(n_{\mathcal{Q}}/n_{\mathcal{G}})^2$ (full).

- A simple compression/decompression algorithm recovers external edges plus averaged internal edges (**average compression**)
- Annotating the quotient, we can achieve **lossless compression** with a (not so simple) algorithm
- Pseudocode in the paper, and MATLAB code available in Bitbucket

Computational reduction

- (**Orbit reduction**) $F(i,j) = F(\sigma(i), \sigma(j))$ means we only need to evaluate F on $m_{\mathcal{Q}} < m_{\mathcal{G}}$ (sparse) or $n_{\mathcal{Q}}^2 \ll n_{\mathcal{G}}^2$ (full) vertex pairs

Computational reduction

- (**Orbit reduction**) $F(i,j) = F(\sigma(i), \sigma(j))$ means we only need to evaluate F on $m_{\mathcal{Q}} < m_{\mathcal{G}}$ (sparse) or $n_{\mathcal{Q}}^2 \ll n_{\mathcal{G}}^2$ (full) vertex pairs — still each value $F(i,j)$ computed on the whole network

Computational reduction

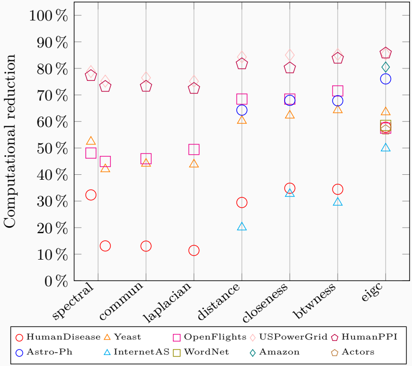
- (**Orbit reduction**) $F(i, j) = F(\sigma(i), \sigma(j))$ means we only need to evaluate F on $m_{\mathcal{Q}} < m_{\mathcal{G}}$ (sparse) or $n_{\mathcal{Q}}^2 \ll n_{\mathcal{G}}^2$ (full) vertex pairs — still each value $F(i, j)$ computed on the whole network
- (**Quotient reduction**) Perform computation on the quotient instead. We call F *quotient recoverable* if $F(\mathcal{G})$ can be obtained from $F(\mathcal{Q})$.

Computational reduction

- (**Orbit reduction**) $F(i,j) = F(\sigma(i), \sigma(j))$ means we only need to evaluate F on $m_{\mathcal{Q}} < m_{\mathcal{G}}$ (sparse) or $n_{\mathcal{Q}}^2 \ll n_{\mathcal{G}}^2$ (full) vertex pairs — still each value $F(i,j)$ computed on the whole network
- (**Quotient reduction**) Perform computation on the quotient instead. We call F *quotient recoverable* if $F(\mathcal{G})$ can be obtained from $F(\mathcal{Q})$.
- In practice we find different 'degrees' of recoverability: we call F *partially*/*average*/*fully* quotient recoverable if the external/external & averaged internal/external & internal edges of $F(\mathcal{G})$ can be obtained from $F(\mathcal{Q})$

Computational reduction

- (**Orbit reduction**) $F(i,j) = F(\sigma(i), \sigma(j))$ means we only need to evaluate F on $m_{\mathcal{Q}} < m_{\mathcal{G}}$ (sparse) or $n_{\mathcal{Q}}^2 \ll n_{\mathcal{G}}^2$ (full) vertex pairs — still each value $F(i,j)$ computed on the whole network
- (**Quotient reduction**) Perform computation on the quotient instead. We call F *quotient recoverable* if $F(\mathcal{G})$ can be obtained from $F(\mathcal{Q})$.



Redundant spectrum

For any (undirected, possibly weighted) network with symmetries (such as $F(\mathcal{G})$), we have:

Redundant spectrum

For any (undirected, possibly weighted) network with symmetries (such as $F(\mathcal{G})$), we have:

- (λ, v) quotient eigenpair then (λ, Sv) parent eigenpair (Sv means repeating values on orbits)

Redundant spectrum

For any (undirected, possibly weighted) network with symmetries (such as $F(\mathcal{G})$), we have:

- (λ, v) quotient eigenpair then (λ, Sv) parent eigenpair (Sv means repeating values on orbits)
- the remaining eigenpairs are of the form $(\lambda, \widetilde{w})$, where (λ, w) is an eigenpair of a symmetric motif $\mathcal{M}$, and $\widetilde{w}$ equals on (the vertices of) $\mathcal{M}$ and 0 elsewhere (*localised* eigenvector)

Redundant spectrum

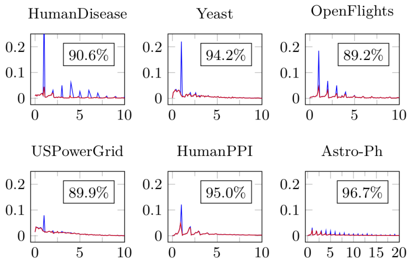
For any (undirected, possibly weighted) network with symmetries (such as $F(\mathcal{G})$), we have:

- (λ, v) quotient eigenpair then (λ, Sv) parent eigenpair (Sv means repeating values on orbits)
- the remaining eigenpairs are of the form $(\lambda, \widetilde{w})$, where (λ, w) is an eigenpair of a symmetric motif $\mathcal{M}$, and $\widetilde{w}$ equals on (the vertices of) $\mathcal{M}$ and 0 elsewhere (*localised* eigenvector)
- Symmetry induces high-multiplicity eigenvalues, and, for empirical networks, explains most of the discrete part of the spectrum

Redundant spectrum

For any (undirected, possibly weighted) network with symmetries (such as $F(\mathcal{G})$), we have:

- (λ, v) quotient eigenpair then (λ, Sv) parent eigenpair (Sv means repeating values on orbits)
- the remaining eigenpairs are of the form $(\lambda, \widetilde{w})$, where (λ, w) is an eigenpair of a symmetric motif $\mathcal{M}$, and $\widetilde{w}$ equals on (the vertices of) $\mathcal{M}$ and 0 elsewhere (*localised* eigenvector)
- Symmetry induces high-multiplicity eigenvalues, and, for empirical networks, explains most of the discrete part of the spectrum



Redundant spectrum

- Moreover, we can predict the most significant redundant eigenvalues from the structure of the BSMs with up to few orbits:

Redundant spectrum

- Moreover, we can predict the most significant redundant eigenvalues from the structure of the BSMs with up to few orbits:

Table 1: Redundant spectra of BSMs with one or two orbits

BSM	eigenvalues	mult	eigenvectors
one orbit	$-\alpha + \beta$	$n-1$	$\mathbf{e}_i$
two orbits	$-b - \kappa_1 c$	$n-1$	$(\kappa_1 \mathbf{e}_i \mid \mathbf{e}_i)$
	$-b - \kappa_2 c$	$n-1$	$(\kappa_2 \mathbf{e}_i \mid \mathbf{e}_i)$

where κ_1 and κ_2 are the two solutions of the quadratic equation $c\kappa^2 + (-a+b)\kappa - c = 0$, and a, b, c, α, β depends on F evaluated on the BSM

Redundant spectrum

- Moreover, we can predict the most significant redundant eigenvalues from the structure of the BSMs with up to few orbits:

Table 1: Redundant spectra of BSMs with one or two orbits

BSM	eigenvalues	mult	eigenvectors
one orbit	$-\alpha+\beta$	$n-1$	$\mathbf{e}_i$
two orbits	$-b-\kappa_1 c$	$n-1$	$(\kappa_1\mathbf{e}_i \mid \mathbf{e}_i)$
	$-b-\kappa_2 c$	$n-1$	$(\kappa_2\mathbf{e}_i \mid \mathbf{e}_i)$

  where κ_1 and κ_2 are the two solutions of the quadratic equation $c\kappa^2+(-a+b)\kappa-c=0$, and a,b,c,α,β depends on F evaluated on the BSM

- Example: for the graph Laplacian and 1-orbit BSMs, we obtain $\mathbb{Z}^+$

Spectral reconstruction algorithm

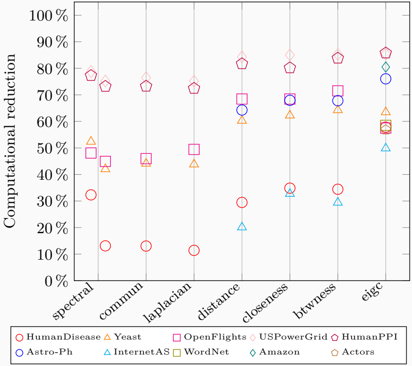
- We can compute the full eigendecomposition of A, or $F(A)$ for any measure F, from the eigendecomposition of the quotient matrix, and the symmetric motifs

Spectral reconstruction algorithm

- We can compute the full eigendecomposition of A, or $F(A)$ for any measure F, from the eigendecomposition of the quotient matrix, and the symmetric motifs
- Computational reduction up to $(n_{\mathcal{Q}}/n_{\mathcal{G}})^3$

Spectral reconstruction algorithm

- We can compute the full eigendecomposition of A, or $F(A)$ for any measure F, from the eigendecomposition of the quotient matrix, and the symmetric motifs
- Computational reduction up to $(n_{\mathcal{Q}}/n_{\mathcal{G}})^3$



Implementation

Implementation

- Calculating $\mathrm{Aut}(\mathcal{G})$ is computationally hard in general, but extremely fast in practice for the large but sparse networks typically found in applications

Implementation

- Calculating $\mathrm{Aut}(\mathcal{G})$ is computationally hard in general, but extremely fast in practice for the large but sparse networks typically found in applications

Step 1 We used `saucy` to compute generators for $\mathrm{Aut}(\mathcal{G})$ ($t_{\mathsf{max}} = 8.02$s for $n_{\mathcal{G}} = 5 \times 10^6$)

Implementation

- Calculating $\mathrm{Aut}(\mathcal{G})$ is computationally hard in general, but extremely fast in practice for the large but sparse networks typically found in applications

Step 1 We used `saucy` to compute generators for $\mathrm{Aut}(\mathcal{G})$ ($t_{\mathrm{max}} = 8.02\mathrm{s}$ for $n_{\mathcal{G}} = 5 \times 10^6$)

Step 2 Symmetric motifs can be obtained from a disjoint-support decomposition of the generators in linear time ($t_{\mathrm{max}} = 4.17\mathrm{s}$ in our test networks)

Implementation

- Calculating Aut($\mathcal{G}$) is computationally hard in general, but extremely fast in practice for the large but sparse networks typically found in applications

Step 1 We used `saucy` to compute generators for Aut($\mathcal{G}$) ($t_{\max} = 8.02$s for $n_{\mathcal{G}} = 5 \times 10^6$)

Step 2 Symmetric motifs can be obtained from a disjoint-support decomposition of the generators in linear time ($t_{\max} = 4.17$s in our test networks)

Step 3 We used GAP^* to obtain orbits and type of each symmetric motif ($t_{\max} =$ large* but parallelizable)

Implementation

- Calculating $\mathrm{Aut}(\mathcal{G})$ is computationally hard in general, but extremely fast in practice for the large but sparse networks typically found in applications

Step 1 We used `saucy` to compute generators for $\mathrm{Aut}(\mathcal{G})$
($t_{\text{max}} = 8.02\text{s}$ for $n_{\mathcal{G}} = 5 \times 10^6$)

Step 2 Symmetric motifs can be obtained from a disjoint-support decomposition of the generators in linear time
($t_{\text{max}} = 4.17\text{s}$ in our test networks)

Step 3 We used GAP^* to obtain orbits and type of each symmetric motif
($t_{\text{max}} = \text{large}^*$ but parallelizable)

- Code available in BitBucket

Applications (examples)

- We studied symmetry compression, computational reduction, and the redundant spectrum of several well-known network measures:
  - communicability ($f(A) = \sum_{n=0}^{\infty} a_k A^k$)
  - shortest path distance
  - resistance metric (equivalent to $L^\dagger$)
  - adjacency matrix (supersedes [MSA08] and [MS09])
  - graph Laplacian

Full details in the preprint Sanchez-Garcia *Exploiting symmetries in network analysis* arxiv/1803.06915

Vertex measures

- Many important network measures are vertex-based (e.g. centrality)

Vertex measures

- Many important network measures are vertex-based (e.g. centrality)
- A *structural* vertex measure $G\colon V \to \mathbb{R}$ satisfies

$$G(i) = G(\sigma(i)) \quad \text{for all } i \in V \text{ and } \sigma \in \mathrm{Aut}(\mathcal{G})$$

(that is, it is constant on orbits).

Vertex measures

- Many important network measures are vertex-based (e.g. centrality)
- A *structural* vertex measure $G\colon V \to \mathbb{R}$ satisfies

$$G(i) = G(\sigma(i)) \quad \text{for all } i \in V \text{ and } \sigma \in \mathrm{Aut}(\mathcal{G})$$

  (that is, it is constant on orbits).
- Enough to compute/store G once per orbit

Vertex measures

- Many important network measures are vertex-based (e.g. centrality)
- A *structural* vertex measure $G\colon V \to \mathbb{R}$ satisfies

$$G(i) = G(\sigma(i)) \quad \text{for all } i \in V \text{ and } \sigma \in \mathrm{Aut}(\mathcal{G})$$

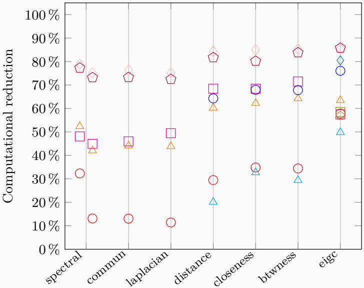
  (that is, it is constant on orbits).
- Enough to compute/store G once per orbit
- Quotient reduction often holds (e.g. degree, eccentricity, closeness and eigenvector centrality)

Vertex measures

- Many important network measures are vertex-based (e.g. centrality)
- A *structural* vertex measure $G\colon V \to \mathbb{R}$ satisfies

$$G(i) = G(\sigma(i)) \quad \text{for all } i \in V \text{ and } \sigma \in \mathrm{Aut}(\mathcal{G})$$

  (that is, it is constant on orbits).
- Enough to compute/store G once per orbit
- Quotient reduction often holds (e.g. degree, eccentricity, closeness and eigenvector centrality)



Vertex measures

- Many important network measures are vertex-based (e.g. centrality)
- A *structural* vertex measure $G\colon V \to \mathbb{R}$ satisfies

$$G(i) = G(\sigma(i)) \quad \text{for all } i \in V \text{ and } \sigma \in \mathrm{Aut}(\mathcal{G})$$

  (that is, it is constant on orbits).
- Enough to compute/store G once per orbit
- Quotient reduction often holds (e.g. degree, eccentricity, closeness and eigenvector centrality)
- Often vertex measures arise from pairwise measures
  e.g. $G(i) = F(i,i)$ or $G(i) = \frac{1}{n}\sum_j F(i,j)$.

Conclusions

For arbitrary (pairwise) structural network measures we have shown:

- a general framework to describe, manipulate and quantify the inherited symmetry and redundancy on an arbitrary network measure
- symmetry compression algorithms with average or lossless compression
- how to use the quotient network for computational reduction
- the contribution of symmetry to the discrete part of the spectrum
- how to extend the results on compression and computational reduction to arbitrary vertex-based measures
- illustrated our methods in several pairwise and vertex-based measures in empirical networks up to several million nodes

References

[MSA08] MacArthur, Sanchez-Garcia, Anderson *Symmetry in Complex Networks* **Discrete Appl. Math.** (2008)

[MS09] MacArthur, Sanchez-Garcia *Spectral characteristics of network redundancy* **Phys. Rev. E** (2009)

[preprint] Sanchez-Garcia *Exploiting symmetries in network analysis* arxiv preprint 1803.06915

[BitBucket] `https://bitbucket.org/rubenjsanchezgarcia/networksymmetry/`

Questions?