Chapter 12
On the Communities of a Population-based SHM Network

Daniel S. Brennan and Keith Worden

Abstract The aspiration of Population-based Structural Health Monitoring (PBSHM) is to accumulate a deeper knowledge of a structure's health, by harvesting available data across a range of similar structures (or substructures). Whilst each structure is represented via a unified language called an Irreducible Element (IE) model, the comparison and similarity metrics are computed within a graph space. As such, each model is subsequently converted into an Attributed Graph (AG) before being placed into the PBSHM Framework's comparison graph space; *the network*.

This paper looks into the application of using the PBSHM Framework's comparison graph space as a complex network and determining if the theories of communities and community structures from network science, can be applied to the PBSHM Framework's network, to determine similarity and define yet-unknown populations.

Keywords Population-based structural health monitoring · Graph neural network · Irreducible element model

Introduction

The intention of Population-based Structural Health Monitoring (PBSHM) [1, 2, 3, 4, 5] is to accumulate additional knowledge on a given structure, by harvesting available data across a range of similar structures — the population. Whilst there are inevitable a myriad of possible methods to establish the aforementioned similarity between real-world structures, this paper will focus on the *Network of Structures* approach defined in the first five papers in the 'Foundations of PBSHM' series [1, 2, 3, 4, 5]. Each structure is represented within the network as a node, weighted links are then established between the structure nodes using the graph-based similarity metrics in the PBSHM Framework.

Previous work in the field of PBSHM has often used a defining feature of the structure as the delimiter for stating what the populations within the network should be — say the number of spans within a bridge. These drawn populations are then used as the judgement for benchmarking if the similarity metric is performing as desired or not. This paper explores the idea of flipping this notion on the head and instead, letting the similarity metrics themselves decide what the populations are within our network via the vehicle of network communities.

The layout of this paper is as follows. The background theory to understanding how structures are represented within the network is set out in the first section. The second section explains how hierarchical clustering and the Ravasz algorithm is used within the PBSHM Network to generate communities — and thus sub-populations — within the network. Lastly a conclusion is given with the next steps to realising this algorithm as a method for generating populations.

Background

Before one can start to compare structures via any form of a similarity metric, one must first have an equal and fair procedure for describing the structures in question — a universal language for capturing the structurally-significant information. The aforementioned vehicle for accomplishing this embedding of information in PBSHM is an Irreducible Element (IE) model [5]. Whilst the scope of the IE-model language is too large to be included within this paper, an abridged recap is included;

Daniel S. Brennan · Keith Worden
Dynamics Research Group, School of Mechanical, Aerospace, and Civil Engineering, University of Sheffield,
Mappin Street, Sheffield, S1 3TD, UK
e-mail: d.s.brennan@sheffield.ac.uk; k.worden@sheffield.ac.uk

© The Author(s), under exclusive license to River Publishers 2026

Kristin B. Zimmerman, PhD (eds.), *Data Science in Engineering, 2026, Vol. 9*, Conference Proceedings
of the Society for Experimental Mechanics Series, https://doi.org/10.13052/rp-9788743814207

however, the interested reader is encouraged to read [5] for a full explanation of the depths of data that can be encapsulated within an IE model.

To generate an IE model of a structure, one must first break down the structure into a set of structurally-significant components — the *elements* of the structure. Once one has a set of elements, these can then be further divided into *regular* elements, which belong to the structure in question, and *ground* elements, which don't belong to the structure, but instead denote the presence of another structure adjacent to the structure one is modelling. One could oversimplify this process to imagine a magic hammer that would instantly reverse any welds, rivets, adhesive, etc that hold a structure together. The set of *element*s would be the components left over after hitting the structure with the aforementioned hammer.

The *relationship*s are the interactions between the components. The welds, rivets, adhesive, etc that the previous metaphor had removed from the structure. Akin to *element*s, the *relationship* objects within the model have four further categorisations. A *perfect* relationship, is when a single larger *element* has been broken down into smaller *element*s for either damage localisation or embedding complex geometries within the model. A *connection* relationship, is where several *element*s are held together by an un-modelled *element* which has been omitted from the model as it has been determined to be structurally non-significant. A *joint* relationship is where the physics between the two *element*s interacting is desired to be modelled. And finally a *boundary* relationship denotes the interaction between a *regular* element and a *ground* element. Each categorisation of both *element*s and *relationship*s has a set of potential information that can be embedded within that type of object in the model.

Figure 1 shows the above process for an example wind turbine. The structure is first broken down into its structurally-significant components: three blades, one rotor hub, one nacelle, three tower parts, and one foundation. These *element*s — and associated *relationship*s — are then converted into a graph domain so as to be compatible with the PBSHM Framework similarity space. It is important to note that the associated graph from an IE model, is not just a purely topological representation, the available model object knowledge is encoded into attributes on the associated vertex or edge.

Once the structure is within the PBSHM Framework's network space, similarity metrics can be computed using the available algorithms in the PBSHM Framework. To date there are four published algorithms that can operate within the PBSHM Network: the Jaccard index with Maximum Common Subgraph (MCS) [2, 6, 7], Graph Kernels [8], Graph Matching Network (GMN) [9], and NetPair [10]. Whilst the middle two algorithms use Machine Learning (ML) [11] to help compute the similarity between two graphs, the first and last algorithms — the Jaccard index with Maximum Common Subgraph and the NetPair algorithm — are graph theory methods [12, 13] for computing similarity.

As the purview of this paper is not to evaluate the available similarity metrics within the framework, the NetPair [10] algorithm has been chosen as the base similarity metric for this paper. Figure 2 shows the standard NetPair similarity metrics against the first five bridges of each span in the PEAR beam-and-slab [14] dataset. As one can see, even though the NetPair algorithm is able to find similarities within the network, the similarity is not confined to the bridges with the same number

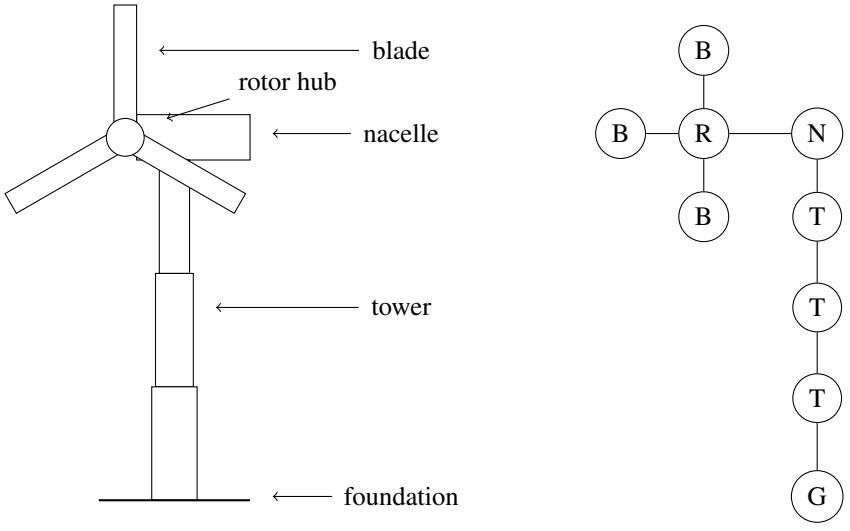

Fig. 1 The process of breaking down a structure (a wind turbine in this example) into its Irreducible Elements and then into its final form as an Attributed Graph (AG) for comparison in the *Network of Structures*

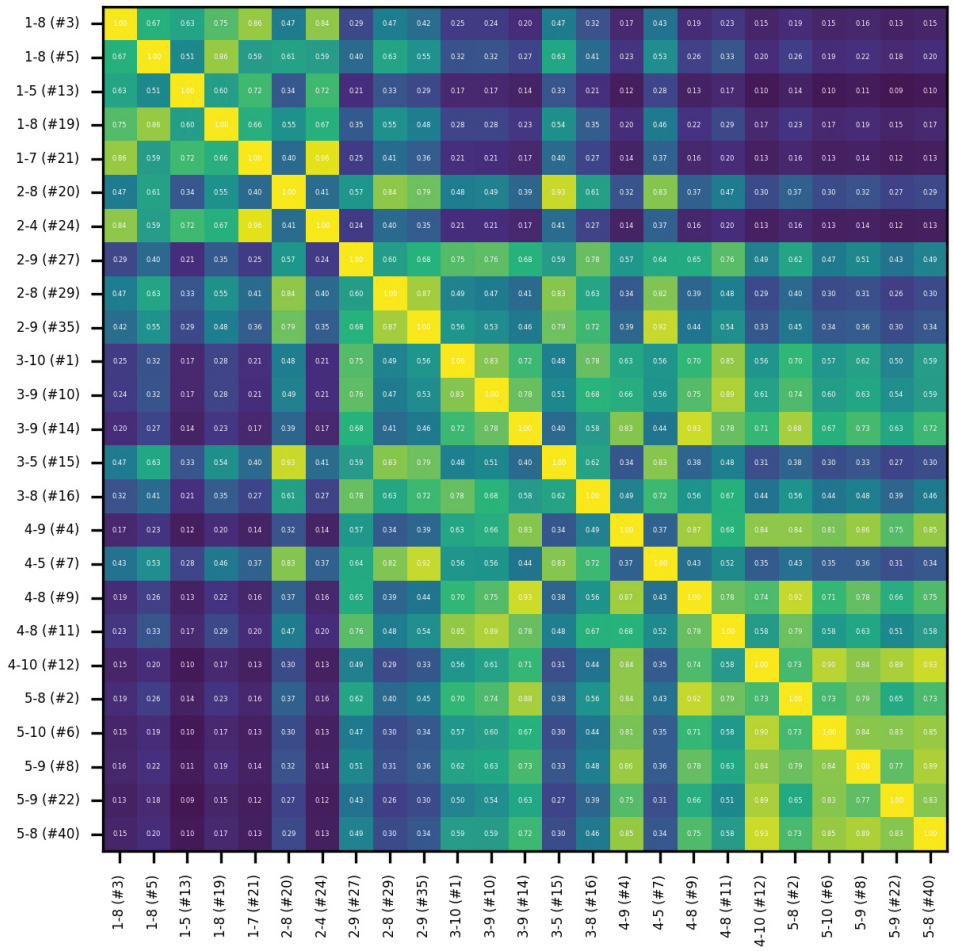

Fig. 2 NetPair similarity heatmap using the first five bridges from each span in the beam-and-slab PEAR dataset

of spans or even neighbouring number of spans. Therefore, one asks; can network communities provide an answer to what populations exist within the network?

Communities

Communities are a method within Network Science [12, 13] of clustering similar nodes together. One of the methods for achieving community discovery is hierarchical clustering. The starting point for hierarchical clustering is a similarity matrix that illustrates the distance between each node within the network. This similarity matrix could be as trivial as calculating the Euclidean distance between the nodes within the network, or it could be as complicated as the output from the PBSHM Framework's similarity comparisons.

Hierarchical clustering can be split into two types of algorithms: *agglomerative* and *divisive*. Agglomerative algorithms merge nodes/communities with a high similarity into a single community, whereas divisive algorithms remove nodes from a community with a low similarity. One example of an agglomerative algorithm is the Razasz algorithm [15]. The Razasz algorithm can be summarised in four steps:

1. Calculate a similarity matrix for the network.
2. Determine a metric for deciding the similarity of a community.
3. Apply hierarchical clustering.
4. Create a dendrogram.

To apply the Razasz algorithm to the PBSHM Network, one first takes the similarity matrix generated by the PBSHM Framework — in the figures included, this is the default NetPair algorithm with all the features weighted equally. Each node — structure — becomes its own community. The similarity matrix is then scanned to find the highest similarity within the network which is not a match to itself — similarity scores from and to the same structure should have the maximum similarity, which is unity. The two associated communities are then merged together and the average of the community member's similarities are stored back within the similarity matrix. The algorithm then goes back to scanning the matrix for the highest similarity score and merging the associated communities until there is only one community left.

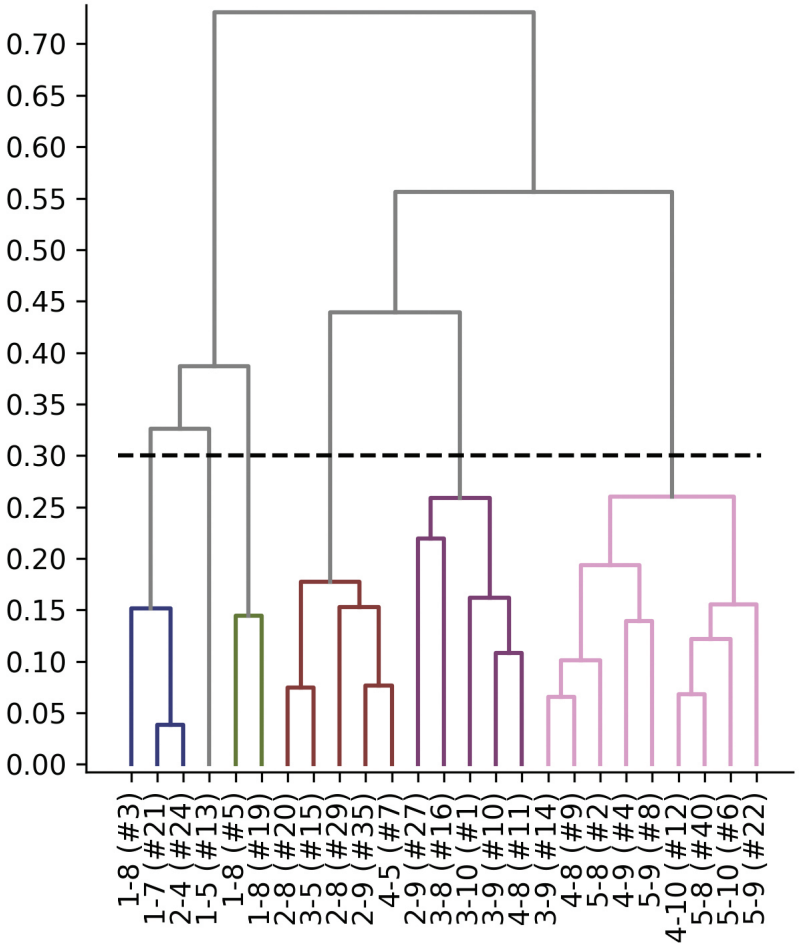

Fig. 3 The dendrogram of an agglomerative hierarchical clustering algorithm on the similarity matrix in Figure 2 with a threshold set of 0.7 similarity

The hierarchical clustering algorithm will always converge the network down to a single community. To be able to pull out N communities from the hierarchical clustering algorithm, one needs to use a dendrogram. A dendrogram is a way of pictorially viewing the merges that occurred during the hierarchical clustering algorithm and inspecting how the merges affect the similarity score of the community. Each horseshoe on the dendrogram represents a merger between the communities attached to the horseshoe. The height of the horseshoe depicts the increase in distance between the members of the community — the decrease in similarity. To determine the communities within the dendrogram, a line is draw at a given distance threshold. Each horizontal line that is cut through by the distance threshold is a community within the network. Figure 3 shows the dendrogram of using the above method on the similarity matrix from Figure 2. Based upon the work by Hughes et al. [16] an approximate threshold has been set of 0.7 similarity — 0.3 distance — to determine the communities within the network.

The six populations — from the associated communities — calculated from the dendrogram, break out of the prescribed populations based around a dominant feature of the structure and instead show previously unknown populations within the

network. Figure 4 shows the same heatmap as in Figure 2; however, in Figure 4 the structures are ordered in the x and y axis in community order. As one can see, once the ordering is changed to be by communities derived from the similarity metrics themselves, the similarity between the populations is clear.

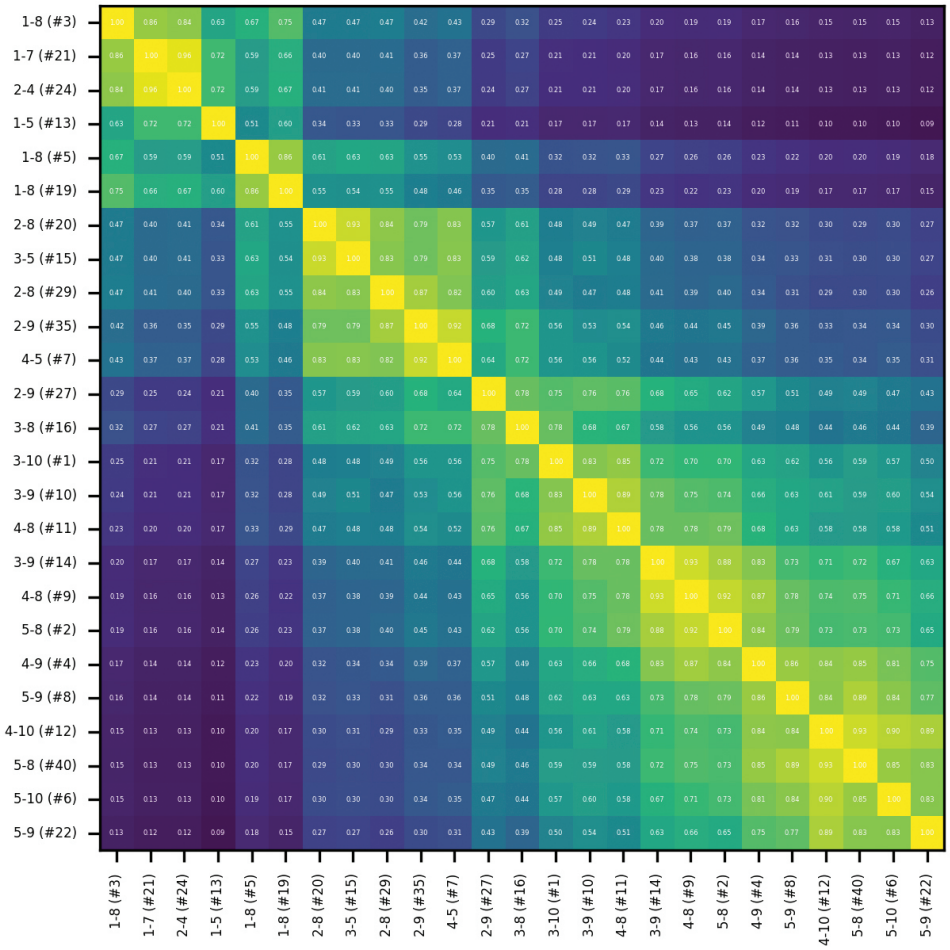

Fig. 4 NetPair similarity heatmap after reordering via the dendrogram using the first five bridges from each span in the beam-and-slab PEAR dataset

Conclusion

In conclusion, this paper has explored how to find network communities from a similarity metric generated by the PBSHM Framework. This paper has shown that our previous methods of using a distinct shared feature on the structures in question — to identify what a sub-population should be — is omitting hidden clusters of similar structures. Instead, letting the network of structures itself inform us on what communities exist, facilitates previously unknown clusters of similar structures being brought into focus. The hope is that by highlighting these unknown clusters of structures, one can potentially perform transfer learning on previously unexplored structures.

Acknowledgments The authors of this paper gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) via grant reference EP/W005816/1. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

References

1. L.A. Bull, P.A. Gardner, J. Gosliga, T.J. Rogers, N. Dervilis, E.J. Cross, E. Papatheou, A.E. Maguire, C. Campos, and K. Worden. "Foundations of Population-based SHM, Part I: Homogeneous populations and forms". *Mechanical Systems and Signal Processing*, 148:107141, 2021.
2. J. Gosliga, P.A. Gardner, L.A. Bull, N. Dervilis, and K. Worden. "Foundations of Population-based SHM, Part II: Heterogeneous populations – Graphs, networks, and communities". *Mechanical Systems and Signal Processing*, 148:107144, 2021.
3. P. Gardner, L.A. Bull, J. Gosliga, N. Dervilis, and K. Worden. "Foundations of Population-based SHM, Part III: Heterogeneous populations – Mapping and transfer". *Mechanical Systems and Signal Processing*, 149:107142, 2021.
4. George Tsialiamanis, C Mylonas, E Chatzi, Nikolaos Dervilis, David J. Wagg, and Keith Worden. "Foundations of Population-based SHM, Part IV: Structures and Feature Spaces as Geometry". *Mechanical Systems and Signal Processing*, 157:107692, 2021.
5. Daniel S. Brennan, Julian Gosliga, Elizabeth J. Cross, and Keith Worden. "Foundations of population-based SHM, Part V: Network, framework and database". *Mechanical Systems and Signal Processing*, 223:111602, 2025.
6. Paul Jaccard. "Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines". *Bull Soc Vaudoise Sci Nat*, 37:241–272, 1901.
7. Coen Bron and Joep Kerbosch. "Algorithm 457: Finding all cliques of an undirected graph". *Communications of the ACM*, 16(9):575–577, 1973.
8. Chandula T Wickramarachchi, Julian Gosliga, Andrew Bunce, Daniel S Brennan, David Hester, Elizabeth J Cross, and Keith Worden. "Similarity assessment of structures for population-based structural health monitoring via graph kernels". *Structural Health Monitoring*, page 14759217241265626, 2024.
9. Daniel S. Brennan, Timothy J. Rogers, Elizabeth J. Cross, and Keith Worden. "On calculating structural similarity metrics in population-based structural health monitoring". *Data-Centric Engineering*, 6:e24, 2025.
10. Daniel S. Brennan, Timothy J. Rogers, Keith Worden, and Brennan Klein. "Structural motifs as signatures for distinguishing critical infrastructure networks". *Submitted to Nature Communications*.
11. Simon Rogers and Mark Girolami. *A First Course in Machine Learning*. Chapman & Hall/CRC Machine Learning & Pattern Recognition Series. CRC Press, Taylor & Francis Group, Boca Raton London New York, second edition, first issued in paperback edition, 2020.
12. Albert-László Barabási and Márton Pósfai. *Network Science*. Cambridge University Press, Cambridge, United Kingdom, 2016.
13. Mark E. J. Newman. *Networks*. Oxford University Press, Oxford, second edition edition, 2018.
14. Connor O'Higgins, Daniel S. Brennan, Tristan Gowdridge, Alan J. Ferguson, David Hester, and Keith Worden. "Population-based SHM Engineered Asset Resource (PEAR) database for testing and validation of PBSHM methods". *Submitted to Structural Health Monitoring*.
15. E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A.-L. Barabási. "Hierarchical Organization of Modularity in Metabolic Networks". *Science*, 297(5586):1551–1555, 2002.
16. A. J. Hughes, J. Poole, N. Dervilis, P. Gardner, and K. Worden. "Quantifying the Value of Information Transfer in Population-Based SHM". In *Data Science in Engineering Vol. 10*, pages 19–31, New York, 2025. River Publishers.