Logarithmic Coloring: How Hard is it to Determine Isomorphism?

Chris Augeri[1*], Barry Mullins[1], Leemon Baird[2], Dursun Bulutoglu[1], and Rusty Baldwin[1]
[1]Air Force Institute of Technology (AFIT), Wright-Patterson Air Force Base, Dayton, OH
[2]United States Air Force Academy (USAFA), Colorado Springs, CO

We are interested in canonically ordering multi-dimensional data such as node positions of an unmanned aerial vehicle swarm. If we assume the data is represented by a pair-wise distance matrix, this problem is equivalent to determining graph isomorphism. Our approach to finding a canonical ordering is based on the PageRank algorithm, which sorts vertices, such as web pages, on the leading eigenvector of a perturbed adjacency matrix. The algorithm we present, IsoCanon, orders vertices by lexicographically sorting on the inverse of an adjacency matrix.

Since the inverse of an adjacency matrix may not exist, we use two isomorphism-preserving perturbations that apply the Gershgorin Circle theorem to guarantee the inverse exists and so it may be efficiently obtained. To increase IsoCanon's ability to find a canonical ordering, we also apply the necessary condition that vertices in the same orbit share the same inverse entries. We have demonstrated this algorithm terminates within a logarithmic number of iterations relative to the vertex set magnitude and finds a canonical isomorph of many random and regular graphs.

We first explore how much additional information is needed to enable IsoCanon to canonically label all graphs, such as those based on Hadamard matrices. By randomly coloring a logarithmic number of vertices prior to executing the algorithm, sufficient noise is introduced to successfully identify all tested graphs. We then explore a deterministic method for coloring a logarithmic set of vertices using the matrix inverse prior to execution. This current version identifies more difficult graphs, such as Paley graphs and many posed by Mathon. The end result is the problem of determining isomorphism can be framed as "how hard is it to make a graph easy to identify?".

IsoCanon also has three key features useful for sorting multi-dimensional data: it terminates in polynomial time, it canonically orders many graphs, and this ordering is akin to PageRank with respect to connectivity and vertex significance. IsoCanon uses parallel numerical libraries to obtain the inverse and has been extensively tested on dense graphs having up to 4,000 vertices.