= y . The resulting directed pseudoforest is maximal , and may include self @-@ loops whenever some value x has ? (x) =
x . Alternatively , omitting the self @-@ loops produces a non @-@ maximal pseudoforest . In the other direction , any maximal directed pseudoforest determines a function ? such that ? (x) is the target of the edge that goes out from x , and any non @-@ maximal directed pseudoforest can be made maximal by adding self @-@ loops and then converted into a function in the same way . For this reason , maximal directed pseudoforests are sometimes called functional graphs . Viewing a function as a functional graph provides a convenient language for describing properties that are not as easily described from the function @-@ theoretic point of view ; this technique is especially applicable to problems involving iterated functions , which correspond to paths in functional graphs .

Cycle detection , the problem of following a path in a functional graph to find a cycle in it , has applications in cryptography and computational number theory , as part of Pollard 's rho algorithm for integer factorization and as a method for finding collisions in cryptographic hash functions . In these applications , ? is expected to behave randomly ; Flajolet and Odlyzko study the graph @-@ theoretic properties of the functional graphs arising from randomly chosen mappings . In particular , a form of the birthday paradox implies that , in a random functional graph with n vertices , the path starting from a randomly selected vertex will typically loop back on itself to form a cycle within O (? n) steps . Konyagin et. al. have made analytical and computational progress on graph statistics .

Martin , Odlyzko , and Wolfram investigate pseudoforests that model the dynamics of cellular automata . These functional graphs , which they call state transition diagrams , have one vertex for each possible configuration that the ensemble of cells of the automaton can be in , and an edge connecting each configuration to the configuration that follows it according to the automaton 's rule . One can infer properties of the automaton from the structure of these diagrams , such as the number of components , length of limiting cycles , depth of the trees connecting non @-@ limiting states to these cycles , or symmetries of the diagram . For instance , any vertex with no incoming edge corresponds to a Garden of Eden pattern and a vertex with a self @-@ loop corresponds to a still life pattern .

Another early application of functional graphs is in the trains used to study Steiner triple systems . The train of a triple system is a functional graph having a vertex for each possible triple of symbols ; each triple pqr is mapped by ? to stu , where pqs , prt , and qru are the triples that belong to the triple system and contain the pairs pq , pr , and qr respectively . Trains have been shown to be a powerful invariant of triple systems although somewhat cumbersome to compute .

= = Bicircular matroid = =

A matroid is a mathematical structure in which certain sets of elements are defined to be independent , in such a way that the independent sets satisfy properties modeled after the properties of linear independence in a vector space . One of the standard examples of a matroid is the graphic matroid in which the independent sets are the sets of edges in forests of a graph ; the matroid structure of forests is important in algorithms for computing the minimum spanning tree of the graph . Analogously , we may define matroids from pseudoforests .

For any graph G = (V , E) , we may define a matroid on the edges of G , in which a set of edges is independent if and only if it forms a pseudoforest ; this matroid is known as the bicircular matroid (or bicycle matroid) of G. The smallest dependent sets for this matroid are the minimal connected subgraphs of G that have more than one cycle , and these subgraphs are sometimes called bicycles . There are three possible types of bicycle : a theta graph has two vertices that are connected by three internally disjoint paths , a figure 8 graph consists of two cycles sharing a single vertex , and a handcuff graph is formed by two disjoint cycles connected by a path . A graph is a pseudoforest if and only if it does not contain a bicycle as a subgraph .

= = Forbidden minors = =

Forming a minor of a pseudoforest by contracting some of its edges and deleting others produces another pseudoforest . Therefore , the family of pseudoforests is closed under minors , and the Robertson ? Seymour theorem implies that pseudoforests can be characterized in terms of a finite set of forbidden minors , analogously to Wagner 's theorem characterizing the planar graphs as the graphs having neither the complete graph K5 nor the complete bipartite graph K3,3 as minors . As discussed above , any non @-@ pseudoforest graph contains as a subgraph a handcuff , figure 8 , or theta graph ; any handcuff or figure 8 graph may be contracted to form a butterfly graph (five @-@ vertex figure 8) , and any theta graph may be contracted to form a diamond graph (four @-@ vertex theta graph) , so any non @-@ pseudoforest contains either a butterfly or a diamond as a minor , and these are the only minor @-@ minimal non @-@ pseudoforest graphs . Thus , a graph is a pseudoforest if and only if it does not have the butterfly or the diamond as a minor . If one forbids only the diamond but not the butterfly , the resulting larger graph family consists of the cactus graphs and disjoint unions of multiple cactus graphs .

More simply , if multigraphs with self @-@ loops are considered , there is only one forbidden minor , a vertex with two loops .

= = Algorithms = =

An early algorithmic use of pseudoforests involves the network simplex algorithm and its application to generalized flow problems modeling the conversion between commodities of different types . In these problems , one is given as input a flow network in which the vertices model each commodity and the edges model allowable conversions between one commodity and another . Each edge is marked with a capacity (how much of a commodity can be converted per unit time) , a flow multiplier (the conversion rate between commodities) , and a cost (how much loss or , if negative , profit is incurred per unit of conversion) . The task is to determine how much of each commodity to convert via each edge of the flow network , in order to minimize cost or maximize profit , while obeying the capacity constraints and not allowing commodities of any type to accumulate unused . This type of problem can be formulated as a linear program , and solved using the simplex algorithm . The intermediate solutions arising from this algorithm , as well as the eventual optimal solution , have a special structure : each edge in the input network is either unused or used to its full capacity , except for a subset of the edges , forming a spanning pseudoforest of the input network , for which the flow amounts may lie between zero and the full capacity . In this application , unicyclic graphs are also sometimes called augmented trees and maximal pseudoforests are also sometimes called augmented forests .

The minimum spanning pseudoforest problem involves finding a spanning pseudoforest of minimum weight in a larger edge @-@ weighted graph G. Due to the matroid structure of pseudoforests , minimum @-@ weight maximal pseudoforests may be found by greedy algorithms similar to those for the minimum spanning tree problem . However , Gabow and Tarjan found a more efficient linear @-@ time approach in this case .

The pseudoarboricity of a graph G is defined by analogy to the arboricity as the minimum number of pseudoforests into which its edges can be partitioned ; equivalently , it is the minimum k such that G is (k , 0) -sparse , or the minimum k such that the edges of G can be oriented to form a directed graph with outdegree at most k . Due to the matroid structure of pseudoforests , the pseudoarboricity may be computed in polynomial time .

A random bipartite graph with n vertices on each side of its bipartition , and with cn edges chosen independently at random from each of the n2 possible pairs of vertices , is a pseudoforest with high probability whenever c is a constant strictly less than one . This fact plays a key role in the analysis of cuckoo hashing , a data structure for looking up key @-@ value pairs by looking in one of two hash tables at locations determined from the key : one can form a graph , the " cuckoo graph " , whose vertices correspond to hash table locations and whose edges link the two locations at which one of the keys might be found , and the cuckoo hashing algorithm succeeds in finding locations for all of its keys if and only if the cuckoo graph is a pseudoforest .

Pseudoforests also play a key role in parallel algorithms for graph coloring and related problems .