The Choosability of the Order-2 Sudoku Graph

Eric Miller

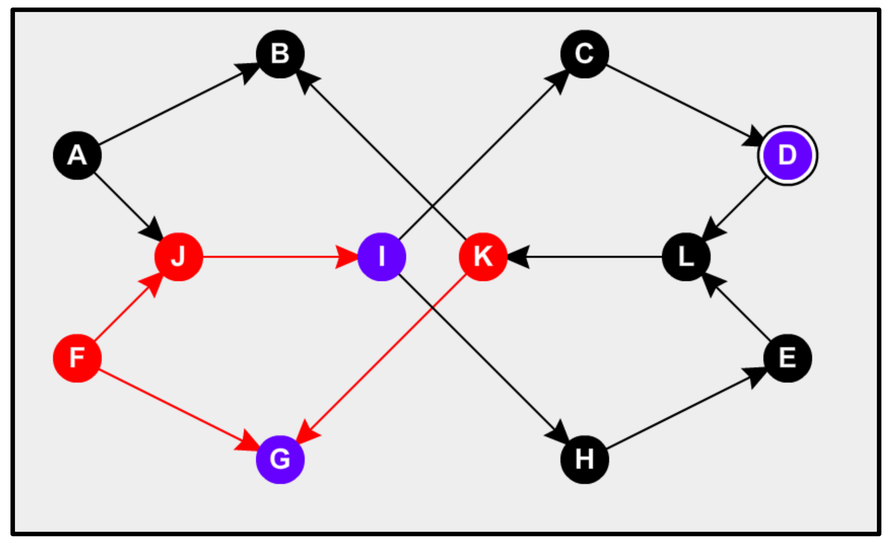
Advisors: Greg Clark and Dr. Joshua Cooper, USC Department of Mathematics

Introduction

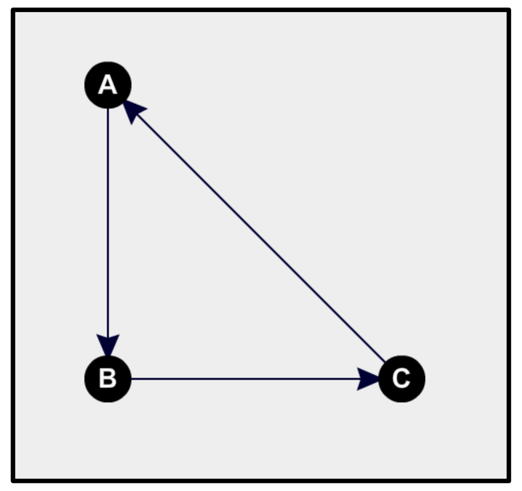
Let $S_n = (V,E)$ be the graph whose vertex set are the n^2 cells of an *(nxn)*-array where two vertices are adjacent if and only if they are in the same row or column. In addition, a graph coloring is defined as the condition where all vertices $\in V$ are colored such that no adjacent vertices have the same color, with the chromatic number notating the number of colors required to achieve this property. In 1979 Dinitz conjectured that any *nxn* Latin square had a list coloring number of *n*, where a list coloring (denoted as $\chi l(G)$) is defined as the smallest number *k* such that for any list of color sets $C(v)$ with $|C(v)| = k$ for all $v \in V$ there always exists a list coloring. Galvin proved this notion in 1994 for not only the *nxn* Latin square, but a much larger family of graphs. However, he failed to prove this result for $O_n = (V, E)$, where O_n denotes the graph corresponding to a $n^2 \times n^2$ Latin square with *nxn* blocks, otherwise known as a order *n*-Sudoku board. We aim to prove the list chromatic number for our S_n graph to be at most $n+1$.

Graphs

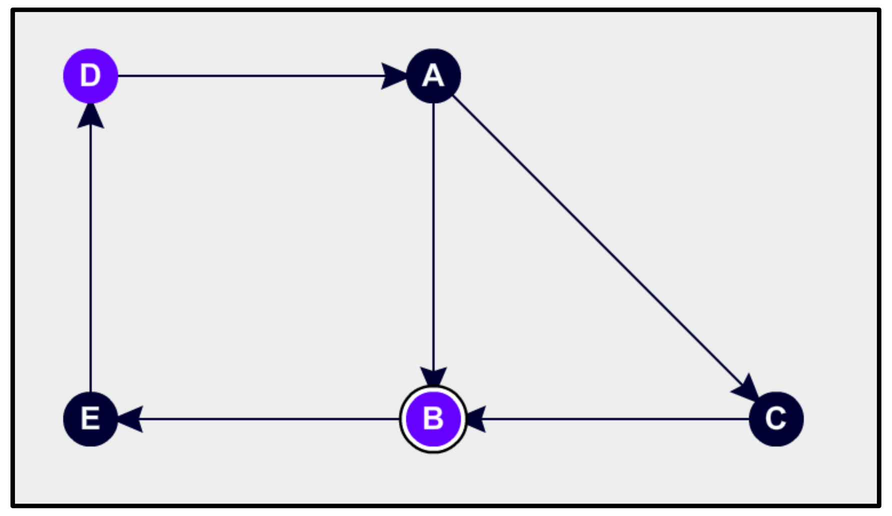
Example 1: An arbitrary digraph. The highlighted vertices (in any color) and their corresponding edges identify an induced subgraph. The vertices highlighted in blue indicate the kernel.



Example 2: A graph containing a 3-cycle. Notice that because no one vertex directs into both of the others, a kernel does not exist.



Example 3: A graph containing a 5-cycle. Despite this, it still has a kernel (highlighted in purple).



Results

Definition 1: Let $G = (V, E)$ be a directed graph. A kernel $K \subseteq V$ is a subset of the vertices such that:

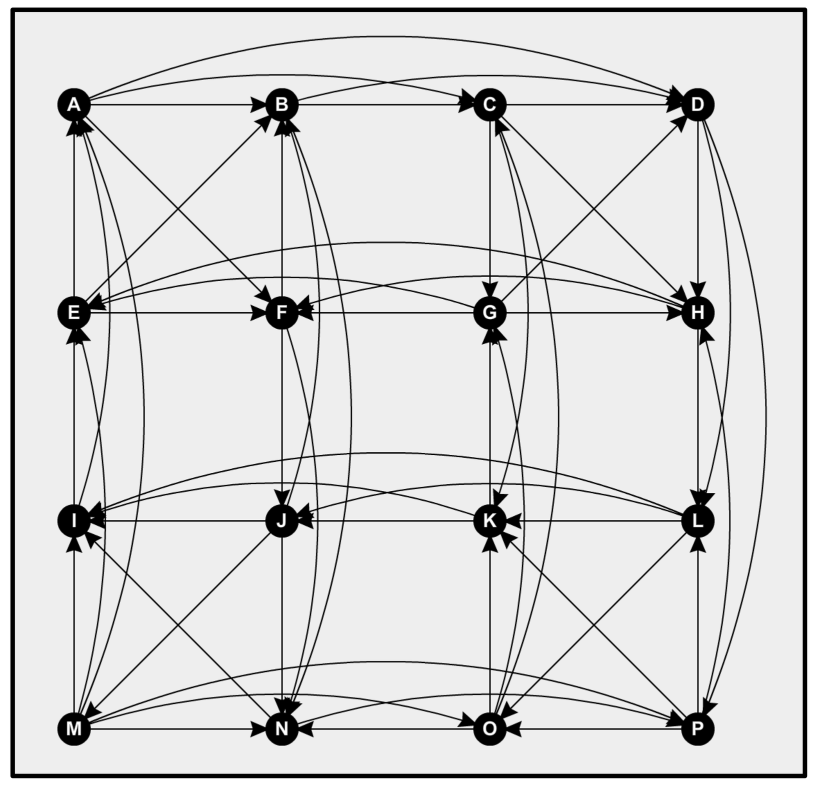
1. K is independent in G, and
2. for every $u \notin K$ there exists a vertex $v \in K$ with an edge $u \rightarrow v$.

Definition 2: Let $G = (V, E)$ be a directed graph. An induced subgraph S_G is the graph inherited from any subset of vertices $v \in V$ and their corresponding edges.

Definition 3: A simple digraph is kernel-perfect if every induced subgraph has a kernel.

We show that the order-2 Sudoku board is kernel-perfect using the following orientation of the edges of the order-2 Sudoku graph:

1. orient all rows from smaller to larger values
2. orient all columns from larger to smaller values
3. orient diagonals towards vertex with indegree 2
4. orient remaining diagonals to prevent directed 3 cycles (from 1 to 4).



Let *D* be the digraph inherited from *L* with the orientation *Z*. By construction *D* has no directed 3-cycles; however, D does contain several 5-cycles, 7-cycles, etc. To determine if *D* is kernel-perfect, it remains to be shown that every induced subgraph of *D* containing a directed odd cycle contains a kernel.

Theorem 4: (Richardson, 1946)**:** Every digraph without an odd directed cycle is kernel-perfect.

Lemma 5: Let $G = (V, E)$ be a directed graph, and suppose that for each vertex $v \in V$ we have a color set $C(v)$ that is larger than the outdegree, $|C(v)| \geq d+(v) + 1$. If every induced subgraph of *G* possess a kernel, then there exists a list color of *G* with a color from $C(v)$ for each v.

Main Result: $4 \leq \chi l(O_2) \leq 5$.

To show that $\chi l(O_2) = 5$ would require finding an assignment of (at least) four colors to each cell of the order-2 Sudoku board such that any choice of colors would not yield a valid board. We accomplish this by testing combinations of vertices using SageMath. We do not need to test every possible set of vertices. This is due to the symmetry exhibited by our orientation. By testing both possible 3-cycles reflected over the block-1 negative-sloped diagonal, it is possible to show via symmetry that there are no odd-cycles remaining in the graph that do not have a kernel.

Results (Code)

```
def solver(fgraph, oddcycle):
    nextlist = list(set(fgraph.keys()) - set(oddcycle))
    nextgraph = dict((z, fgraph[z]) for z in (nextlist))
    subgraphlist = []

    for a in range(0, len(nextgraph)+1):
        for subset in itertools.combinations(nextgraph, a):
            subgraphlist.append(list(set(subset) | set(oddcycle)))
    for b in subgraphlist:
        if checker(b) == False:
            return False
    return True
def checker(combo):
    subgraph = dict((z, graph[z]) for z in (combo));
    sublist = [];
    for a in range(0, len(combo)+1):
        for subset in itertools.combinations(combo, a):
            sublist.append(subset)
    for b in sublist:
        if kerneler(subgraph,b) == True:
            return True
def kerneler(subg,propk):
    kernel = dict((z, subg[z]) for z in (propk));
    nklist = list(set(subg.keys()) - set(list(propk)));
    notknl = dict((w, subg[w]) for w in (nklist));
    if kernel.keys() == []:
        return False
    if bool(set(kernel.keys()) & set(lister(kernel))) == True:
        return False
    for i in notknl.keys():
        subknl = dict((x, subg[x]) for x in (list(i)))
        if bool(set(lister(subknl)) & set(kernel.keys())) == False:
            return False
    return True
def lister(othergraph):
    valuelist = [];
    svaluelist = [];
    for a in range(len(othergraph)):
        for b in range(len(othergraph.values()[a])):
            valuelist.append(othergraph.values()[a][b])
    for all in valuelist:
        svaluelist = list(set(valuelist));
    return svaluelist
```

Conclusion

Using this algorithm, we show that the list coloring number of a graph inherited from a Latin square with blocks (a Sudoku square) of order 2 is either 4 or 5. Further application of this algorithm to graphs of order greater than 2 is possible, and we predict it will return the same result.

Conjecture 6: $n^2 \leq \chi l(O_n) \leq n^2 + 1$ for any n

Possible problems with this course of action are surmountable in the short run, but not as *n* gets large (beyond about 5 or 6). One issue will be that as the graph gets larger, the number of induced subgraphs needed to be tested would rise exponentially. Another problem is that our orientation was selected specifically for its symmetrical properties; each graph for larger and larger *n* will need its own specific orientation. To this end, we will need to construct an algorithm for selecting the best orientation.

Further Theoretical Work

Definition 7: A digraph is transitive if for all $\{x,y,z\} \subseteq V$, $(x,y),(y,z) \in E$ implies $(x,z) \in E$.

While it is possible to extend this argument to O_n where $n > 2$, it would become much more computationally intensive as *n* gets arbitrarily large (or even simply above 4 or 5). However, we observe that each block of O_n induces a graph K_n. In essence, for any valid orientation of O_n does not contain three cyclic vertices, as we cannot have any 3-cycles. Thus, each block is transitive.

We seek to extend our results. As each block is transitive there is a vertex *v* whose edge is directed outwards to all vertices in the same block. In particular, *v* has $n^2 - 1$ edges directed outwards towards other vertices in the same row and column. Moreover, *v* has $(n-1)^2$ edges directed towards the vertices in the same block which are not in the same row or column. Summing over these possibilities and applying Lemma 4 yields the following.

Conjecture 8: $n^2 \leq \chi l(O_n) \leq (n^2-1) + (n-1)^2 + 1 = 2n^2 - 2n + 1$.

Application

Beyond the inherent satisfaction garnered from solving a thus-far unsolved problem, our result has potentially important implications for statistics, specifically block design. Randomized block design on ANOVA tests with more than one block category are traditionally modeled using a Latin square. This is in order to save money on test materials/conducting without significantly sacrificing test precision. Our result may allow for the application of more complex Latin square block designs, allowing firms to potentially further cut experiment costs.

References

1. H. Galeana-Sanchez and V. Neumann-Lara, *On Kernel-Perfect Critical Graphs* (North-Holland, Amsterdam, 1986)
2. P. Erdos, A. L. Rubin, H. Taylor, *Choosability in Graphs*
3. D. Gale and L.S. Shapley, *College Admissions and the Stability of Marriage* (*The American Mathematical Society,,* Jan. 1962*)*
4. H. Galeana-Sanchez and M. Guevara, *Kernel perfect and critical kernel imperfect digraphs structure* (2015)
5. P. Haxell and J. Verstraete, *List coloring hypergraphs* (2010)
6. S. Gravier, F. Maffray, and L. Pastor, *On the choosability of claw-free perfect graphs*, (Nov. 2015)
7. J. C. M. Janssen, *The Dinitz Problem Solved for Rectangles* (*The American Mathematical Society Bulletin*, Oct. 1992)
8. M. Aigner and G.M. Ziegler. "Chapter 26." *Proofs from the Book*. (Berlin: *Springer*, 2004)
9. F. Galvin, *The List Chromatic Index of a Bipartite Multigraph* (*Journal of Combinatorial Theory*, 1995)
10. D. Zeilberger, *The Method of Undetermined Generalization and Specialization Illustrated with Fred Galvin's Amazing Proof of the Dinitz Conjecture* (Jun. 1995)
11. R.H. Hammack,. *Book of Proof*. (Richmond, VA: Virginia Commonwealth U, 2013)