CORES AND SHELLS OF GRAPHS

Allan Bickle, Kalamazoo

(Received June 13, 2011)

Abstract. The k-core of a graph G, $C_k(G)$, is the maximal induced subgraph $H \subseteq G$ such that $\delta(G) \geqslant k$, if it exists. For $k > 0$, the k-shell of a graph G is the subgraph of G induced by the edges contained in the k-core and not contained in the $(k+1)$-core. The core number of a vertex is the largest value for k such that $v \in C_k(G)$, and the maximum core number of a graph, $\widehat{C}(G)$, is the maximum of the core numbers of the vertices of G. A graph G is k-monocore if $\widehat{C}(G) = \delta(G) = k$.

This paper discusses some basic results on the structure of k-cores and k-shells. In particular, an operation characterization of 2-monocore graphs is proven. Some applications of cores and shells to graph coloring and domination are considered.

Keywords: k-core, k-shell, monocore, coloring, domination

MSC 2010: 05C75, 05C15

1. Introduction

One of the basic properties of graphs is the existence of subgraphs with specified degree conditions. (See [11] and [29] for basic terminology.)

Definition 1.1. The *k-core* of a graph G, $C_k(G)$, is the maximal induced subgraph $H \subseteq G$ such that $\delta(G) \geqslant k$, if it exists.

Cores were introduced by S. B. Seidman [26] and have been studied extensively in [6]. They have mostly been studied in the context of random graph theory (e.g. [21]).

Cores have applications outside of mathematics. Seidman briefly explores social networks in his paper. Cores have applications in computer science to network visualization [2], [17]. They also have applications in bioinformatics [1], [3], [31].

It is easy to show that the k-core is well-defined and that the cores of a graph are nested.

Definition 1.2. The *core number* of a vertex, $C(v)$, is the largest value for k such that $v \in C_k(G)$. (This has also been named the coreness of v.) The *maximum core number* of a graph, $\widehat{C}(G)$, is the maximum of the core numbers of the vertices of G. Given $k = \widehat{C}(G)$, the *maximum core* of G is $C_k(G)$.

It is immediate that $\delta(G) \leqslant \widehat{C}(G) \leqslant \triangle(G)$. We can characterize the extremal graphs for the upper bound. For simplicity, we restrict the statement to connected graphs. (See also [29], p. 199.)

Proposition 1.1. *Let G be a connected graph. Then $\widehat{C}(G) = \triangle(G) \iff G$ is regular.*

P r o o f. If G is regular, then its maximum and minimum degrees are equal, so the result is obvious.

For the converse, let $\widehat{C}(G) = \triangle(G) = k$. Then G has a subgraph H with $\delta(H) = \triangle(G) \geqslant \triangle(H)$, so H is k-regular. If H were not all of G, then since G is connected, some vertex of H would have a neighbor not in H, implying that $\triangle(G) > \triangle(H) = \delta(H) = \triangle(G)$. But this is not the case, so $G = H$, and G is regular. □

We also consider the extremal graphs for the lower bound $\delta(G) \leqslant \widehat{C}(G)$.

Definition 1.3. A graph G is *k-monocore* if $\widehat{C}(G) = \delta(G)$. A graph is *monocore* if it is k-monocore for some k.

There is a simple algorithm for determining the k-core of a graph, which we shall call the k-core algorithm.

A l g o r i t h m 1.1 (k-Core Algorithm). Iteratively delete vertices of degree less than k until none remain.

It is straightforward to show that this will produce the k-core if it exists. The k-core algorithm can be implemented in polynomial time. If an adjacency matrix is employed, it can be implemented in $O(n^2)$ time, while [4] showed that using an edge list, it can be implemented in $O(m)$ time, which is better for sparce graphs.

The core number algorithm successively deletes vertices of relatively small degree in a graph until none remain. We can define a sequence that orders the vertices of a graph based on this process. We may also wish to construct a graph by successively adding vertices of relatively small degree.

Definition 1.4. A *deletion sequence* of a graph G is a sequence of its vertices formed by iterating the operation of deleting a vertex of smallest degree and adding it to the sequence until no vertices remain. A *construction sequence* of a graph is the reversal of a corresponding deletion sequence.

We can also consider the degrees of vertices when deleted in a deletion sequence.

Definition 1.5. A graph is k-*degenerate* if its vertices can be successively deleted so that when deleted, each has degree at most k. The *degeneracy* of a graph is the smallest k such that it is k-degenerate.

Graphs that are k-degenerate were defined in [20] and have been explored recently in [5]. As a corollary of the k-core algorithm, we have the following min-max relationship.

Corollary 1.1. *For any graph, its maximum core number is equal to its degeneracy.*

Definition 1.6. A graph is k-*core-free* if it does not contain a k-core.

A graph is maximal with respect to some property if no edge can be added without violating this property. The k-core algorithm also implies that a graph G is k-degenerate if and only if G is $(k+1)$-core-free, and maximal k-degenerate graphs are equivalent to maximal $(k+1)$-core-free graphs.

We now determine the cores of some special classes of graphs. It is immediate that the k-core of a graph is the union of the k-cores of its components. It turns out that many important classes of graphs are monocore. Table 1 summarizes some of the most common. The verification of their core structure is straightforward.

Class of Graphs	Maximum Core Number
r-regular	r
nontrivial trees	1
forests (no trivial components)	1
complete bipartite $K_{a,b}$, $a \leqslant b$	a
$K_{a_1,\ldots,a_n}$, $a_1 \leqslant a_2 \leqslant \ldots \leqslant a_n$	$a_1 + \ldots + a_{n-1}$
wheels	3
maximal outerplanar, $n \geqslant 3$	2

Table 1. Classes of monocore graphs.

As a consequence of a much more difficult theorem of Mader ([8], pp. 21–24), it follows that if a graph is minimally k-connected, then it is k-monocore. For more general classes of graphs, we may only be able to bound the maximum core number. For example, it is well-known that any planar graph G is 5-degenerate, that is, $\widehat{C}(G) \leqslant 5$.

We have seen that the cores of a graph are nested. This in turn can be used to define a decomposition of a graph into subgraphs defined based on those parts of the graph contained in one core and not in the next higher number core.

Definition 2.1. For $k > 0$, the *k-shell* of a graph G, $S_k(G)$, is the subgraph of G induced by the edges contained in the k-core and not contained in the $(k+1)$-core. For $k = 0$, the *0-shell* of G consists of the vertices of the 0-core not contained in the 1-core.

Thus the 0-shell is simply the set of isolated vertices of G. If $M = \widehat{C}(G)$, the M-shell of G is just the maximum core of G. As intended, the k-shells of G form a decomposition of G. Unless each k-shell is a separate component or components of G, the shells of G will have some vertices in common.

Definition 2.2. The *k-boundary* of G, $B_k(G)$, is the set of vertices contained in both the k-shell and the $(k+1)$-core.

Thus a vertex is contained in the k-boundary exactly when it is contained in the $(k+1)$-core and adjacent to a vertex in the k-core. Sometimes it is convenient to exclude the boundary when considering the shell.

Definition 2.3. The *proper k-shell* of G, $S'_k(G)$, is the subgraph of G induced by the non-boundary vertices of the k-shell. The *order of the k-shell* of G is defined to be the order of the proper k-shell.

Thus the vertices of the proper k-shells partition the vertex set of G. A vertex has core number k if and only if it is contained in the proper k-shell of G. Thus the proper k-shell is induced by the vertices with core number k.

Note that the proper k-shell was called the k-remainder of G by Seidman [26] in the 1983 paper that introduced k-cores. That term does not appear to have been used since.

We would like to know which graphs can be k-shells.

Theorem 2.1. *A graph F with vertex subset B is a k-shell of a graph with boundary set B if and only if no component of F has vertices entirely in B, $\delta_F(V(F) \setminus B) = k$, and F contains no subgraph H with $\delta_H(V(H) \setminus B) \geqslant k+1$.*

P r o o f. ($\Rightarrow$) Let F be a k-shell of graph G with boundary set B. If any component of F had all vertices in B, it would be contained in the $(k+1)$-core of G. If a vertex $v \in V(F) \setminus B$ had $d(v) < k$, it would not be in the k-core of G. If F had such a subgraph H, it would be contained in the $(k+1)$-core of G.

($\Leftarrow$) Let F be a graph satisfying these conditions. Overlap each vertex in B with a distinct vertex of a $(k+1)$-core G with sufficiently large order. Then F is the k-shell of the resulting graph. $\square$

The 1-shell of a graph can be characterized in terms of a familiar class of graphs.

Corollary 2.1. *The 1-shell of G, if it exists, is a forest with no trivial components and at most one boundary vertex per component.*

We can also characterize graphs that can be proper k-shells. Certainly such a graph cannot contain a $(k+1)$-core. This obvious necessary condition is also sufficient.

Proposition 2.1. *A graph F is a proper k-shell if and only if F does not contain a $(k+1)$-core.*

Proof. The forward direction is obvious. Let F be a graph that does not contain a $(k+1)$-core. Let M be a $(k+1)$-core. For each vertex $v \in V(F)$, let $a(v) = \max\{k - d(v), 0\}$. For each vertex $v \in V(F)$, take $a(v)$ copies of M and link each to v by an edge between v and a vertex in M. The resulting graph G has F as its proper k-shell. $\square$

Corollary 2.2. *A graph F can be a proper 1-shell if and only if F is a forest.*

We can determine sharp bounds for the size of a k-shell.

Proposition 2.2. *The size m of a k-shell with order n satisfies $\lceil \frac{1}{2}k \cdot n \rceil \leqslant m \leqslant k \cdot n$.*

Proof. The non-boundary vertices of the k-shell of G can be successively deleted so that when deleted, they have degree at most k. Thus $m \leqslant k \cdot n$. The non-boundary vertices have degree at least k, so there are at least $\frac{1}{2}k \cdot n$ edges. $\square$

The lower bound is sharp for all k. For k or n even, the extremal graphs have every component k-regular, and no vertices adjacent to the $(k+1)$-core, if it exists. For k and n both odd, the extremal graphs have a single component with one vertex of degree $k+1$ and all others of degree k, and no vertices adjacent to the $(k+1)$-core, if it exists.

The upper bound is sharp for all $k < \widehat{C}(G)$. The extremal graphs have vertices having degree exactly k when deleted, so they can be constructed by reversing this process. Thus they must have at least k boundary vertices. For $k = \widehat{C}(G)$, the maximum core is k-degenerate, so it can have size at most $k \cdot n - \binom{k+1}{2}$. Thus we have the following corollary.

Corollary 2.3. *Let s_k be the order of the k-shell of G, $0 \leqslant k \leqslant r = \widehat{C}(G)$. Then the size m of G satisfies*

$$\sum_{k=1}^{r} \left\lceil \frac{k \cdot s_k}{2} \right\rceil \leqslant m \leqslant \sum_{k=1}^{r} k \cdot s_k - \binom{k+1}{2}.$$

Both upper and lower bounds are sharp. The extremal graphs have each k-shell extremal, as above.

The bound on the size of a k-shell can be improved by considering the number of boundary vertices.

Proposition 2.3. *The size m of a k-shell with order n and b boundary vertices satisfies*

$$\left\lceil \frac{k \cdot n + b}{2} \right\rceil \leqslant m \leqslant k \cdot n - \binom{k-b+1}{2}.$$

P r o o f. When deleted, the i^{th} to last vertex can have degree at most $b + i - 1$. Thus the upper bound must be reduced by $\sum_{i=1}^{k-b} i = \frac{(k-b)(k-b+1)}{2} = \binom{k-b+1}{2}$. The boundary vertices each contribute degree at least one to the lower bound. The result follows. □

The lower bound is sharp for all k. If $k \cdot n + b$ is even, then every component of the extremal graphs connected to the $(k+1)$-shell can be formed in the following way. Start with a connected k-regular graph with an edge cut of b_i edges. Take the half of the graph on one side of the edge cut and add edges joining the vertices adjacent to the cut to b_i boundary vertices. Add enough components so that the number of boundary vertices and non-boundary vertices sum to the appropriate values. If $k \cdot n + b$ is odd, the construction is similar, but there must be one vertex with degree $k + 1$.

The upper bound is sharp for all k, and the extremal graphs are graphs whose vertices have the maximum possible degree for deletion at each step.

Corollary 2.4. *Let s_k be the order of the k-shell of G and let b_k be the order of the k-boundary of G, $0 \leqslant k \leqslant r = \widehat{C}(G)$. Then the size m of G satisfies*

$$\sum_{k=1}^{r} \left\lceil \frac{k \cdot s_k + b_k}{2} \right\rceil \leqslant m \leqslant \sum_{k=1}^{r} \left(k \cdot s_k - \binom{k - b_k + 1}{2} \right).$$

Note that the maximum core must have no boundary vertices.

3. Applications of k-shells

Cores and shells can be applied to other problems in graph theory.

3.1. Vertex coloring. Cores are essential to a basic upper bound on chromatic number. Establish a deletion sequence for a graph by successively deleting vertices of smallest degree. This orders the vertices in terms of core number. Reverse this sequence to obtain a construction sequence for the graph. Color the graph using this sequence. We obtain a bound first proved by Szekeres and Wilf in 1968 [28], restated in terms of cores.

Theorem 3.1 (The core number bound). *For any graph G, we have $\chi(G) \leqslant 1 + \widehat{C}(G)$.*

Proof. Establish a construction sequence for G. Each vertex has degree at most equal to its core number when colored. Coloring it uses at most one more color. Thus $\chi(G) \leqslant 1 + \widehat{C}(G)$. □

The core number bound implies the corollary that if G has a k-shell, then $\chi(S_k(G)) \leqslant k+1$. This immediately implies that if G has a 2-core, then $\chi(G) = \chi(C_2(G))$. This is because any nonempty graph requires at least two colors, while the 1-shell of a graph is a forest which requires at most two colors. Similarly, we find that if G has a 3-core which is not bipartite, then $\chi(G) = \chi(C_3(G))$. Thus the problem of optimally coloring a graph can be readily reduced to coloring its 3-core.

Similar arguments apply to list coloring.

Definition 3.1. A *list coloring* of a graph begins with lists of length k assigned to each vertex and chooses a color from each list to obtain a proper vertex coloring. A graph G is k-choosable if any assignment of lists to the vertices permits a proper coloring. The *list chromatic number* $\chi_l(G)$ is the smallest k such that G is k-choosable.

The same argument as before implies that $\chi_l(G) \leqslant 1 + \widehat{C}(G)$. Hence we see that every tree is 2-choosable. Thus it is immediate that if G has a 2-core, then $\chi_l(G) = \chi_l(C_2(G))$.

Erdős, Rubin, and Taylor [16] characterized 2-choosable graphs. The θ-graph $\theta_{i,j,k}$ is the graph formed by identifying the endpoints of three paths of lengths i, j, and k. They showed that a connected graph G is 2-choosable if and only if it is a tree or its 2-core is an even cycle or $\theta_{2,2,2k}$ for $k \geqslant 1$. Thus every 2-monocore graph G that is not an even cycle or $\theta_{2,2,2k}$, $k \geqslant 1$, has $\chi_l(G) = 3$. Note the theorem implies that every 2-choosable graph has no 3-core. This implies that if G has a 3-core, then $\chi_l(G) = \chi_l(C_3(G))$.

A more general problem than determining the chromatic number of a graph is to determine the number of distinct colorings of a graph using k colors, for any k. Specifically, the problem is to determine a function that gives the number of distinct k-colorings of G in terms of k. This function must be a polynomial, so it is called the chromatic polynomial of G. (See [12], pp. 211–216 for background.)

Algorithms exist to determine this polynomial, but they are not efficient, so determining the chromatic polynomial is difficult for large graphs. However, this problem can readily be reduced to the 2-core. For each vertex of degree one, one choice is excluded, so there are $k-1$ colors available for it. This leads to the next theorem.

Theorem 3.2. *Let G be a connected* 1*-core containing a* 2*-core, n_1 the order of its* 1*-shell. Then*

$$\chi(G, k) = (k-1)^{n_1} \chi(C_2(G), k).$$

P r o o f. If $n_1 = 0$, then

$$\chi(G, k) = \chi(C_2(G), k) = (k-1)^0 \chi(C_2(G), k).$$

Assume the result holds for order $n_1 = r$ and let G have a 1-shell of order $r+1$. Then G has a leaf vertex v. Let $H = G - v$ and $e = uv$, the edge incident with v. By the chromatic recurrence,

$$\begin{aligned}\chi(G, k) &= \chi(G - e, k) - \chi(G \cdot e, k) = k \cdot \chi(H, k) - \chi(H, k) = (k-1) \cdot \chi(H, k) \\ &= (k-1)(k-1)^r \cdot \chi(C_2(G), k) = (k-1)^{r+1} \cdot \chi(C_2(G), k).\end{aligned}$$

□

3.2. Edge coloring. A proper edge coloring of a graph assigns a color to each edge so that adjacent edges are colored differently. The edge chromatic number of a graph, $\chi_1(G)$, is the smallest number of edges that can be used in a proper edge coloring. Clearly the edge chromatic number is at least as large as the maximum degree. Vizing showed that it is never more than $\triangle(G)+1$. A graph is called class one if $\chi(G) = \triangle(G)$, and class two if $\chi(G) = \triangle(G)+1$. Determining which of the two is the case is a difficult problem in general.

Theorem 3.3. *Let G be a graph with D the maximum degree in G of the vertices in the* 1*-shell of G. Then*

$$\chi_1(G) = \max\{D, \chi_1(C_2(G))\}.$$

P r o o f. Certainly $\chi_1(G) \geqslant D$ and since the 2-core of G is contained in G, $\chi_1(G) \geqslant \chi_1(C_2(G))$.

To show equality, color the 2-core of G with $\chi_1(C_2(G))$ colors. Now color the 1-shell using a construction sequence. Adding an edge adjacent to a boundary vertex will require an additional color if and only if edges of every color used up to that point are incident with v. This holds for adding any edge of the 1-shell. Thus we have equality. □

It is not hard to see that every tree is class one. Since the 1-shell is a forest, we see that G is class two if and only if the 2-core of G is class two and $\triangle(G) = \triangle(C_2(G))$.

Similar techniques are useful for studying other coloring problems. These include arboricity and point partition numbers (see [6]) and 2-tone coloring (see [7]).

3.3. Domination. A set of vertices of a graph is a dominating set if each vertex of G is either in the set or adjacent to a vertex in the set. The domination number of a graph $\gamma(G)$ is the minimum size of a dominating set. See [18] for background.

Many authors have provided upper bounds for the domination number of graphs with some minimum degree. These results are summarized in the table, where there are seven small exceptional graphs for $k = 2$. Note also that after the bounds for $1 \leqslant k \leqslant 3$ had been proved, it was conjectured in [18] that for k-cores, $k \geqslant 1$, $\gamma(G) \leqslant \frac{k}{3k-1}\,n$. The bound for large k, which can be proved by probabilistic means, is superior to this conjecture for $k \geqslant 7$. The conjecture was verified for the cases $4 \leqslant k \leqslant 6$ in three subsequent papers.

k	Bound	Citations
0	n	
1	$\frac{1}{2}n$	[23]
2	$\frac{2}{5}n$ $(G \neq C_4,\, n \neq 7)$	[22]
3	$\frac{3}{8}n$	[24]
4	$\frac{4}{11}n$	[27]
5	$\frac{5}{14}n$	[32]
6	$\frac{6}{17}n$	[19]
large	$[1 - k(\frac{1}{k+1})^{1+\frac{1}{k}}]n$	[9, 10]

Table 2. Bounds on the domination number of k-cores

Combining these results allows us to obtain the following upper bound.

Theorem 3.4. *Let n_k be the number of vertices in the (non-proper) k-shell of G and suppose the 2-shell of G has no component being one of the seven exceptional graphs for $k = 2$. Then*

$$\gamma(G) \leqslant n_0 + \sum_{k=1}^{6} \frac{k}{3k-1} n_k + \sum_{k=7}^{\infty} \left[1 - k\Big(\frac{1}{k+1}\Big)^{1+1/k}\right] n_k.$$

How good a bound is will depend on the graph. It depends on a construction that dominates every boundary vertex at least twice, so it will tend to be worse when the boundaries are larger.

We may improve this bound with more information on the structure of the shells. Since the 2-core and 1-shell decompose a nontrivial connected graph, we consider domination of trees. There is a straightforward algorithm to determine the domination number of a tree detailed in [18]. It starts from the leaves of the tree and works inward, determining a minimal dominating set.

Corollary 3.1. *Let G be a graph with a 1-shell composed of rooted trees T_i with domination numbers $\gamma(T_i)$. Let r' be the number of roots or vertices adjacent to roots contained in the dominating sets D_i produced by the algorithm. Then*

$$\gamma(C_2(G)) + \sum \gamma(T_i) - r' \leqslant \gamma(G) \leqslant \gamma(C_2(G)) + \sum \gamma(T_i)$$

In particular, if no component of the 2-core is one of the seven exceptional graphs, then $\gamma(G) \leqslant \frac{2}{5}|C_2(G)| + \sum \gamma(T_i)$.

P r o o f. The algorithm optimally dominates the 1-shell, possibly overlapping the 2-core, producing an overestimate of the domination number. Removing the vertices in the 2-core or dominating part of the 2-core produces an underestimate. The final bound follows from the bound for $k = 2$. □

We note interesting contrast between vertex coloring and domination. In both cases, we have employed the decomposition of a graph into its 1-shell and 2-core. But when coloring, the trees of the 1-shell are simply annoying appendages to be lopped off toward determining the chromatic number. In contrast, the trees provide a cornerstone upon which to build the foundation of an optimal dominating set, greatly reducing the number of possible dominating sets that need to be checked.

4. The structure of 2-cores

We would like to understand the structure of k-cores. We have already seen several structural results. Certainly G is its own 0-core, and the 1-core of G is formed by deleting all isolated vertices of G. The structure of the 2-core of a graph is less trivial. The following result was observed by Seidman [26].

Proposition 4.1. *If G is connected and has a 2-core, then its 2-core is connected.*

The 3-core of a connected graph need not be connected. For example, joining two vertices of two $(k+1)$-cliques by a path of length at least two yields a connected graph with a disconnected k-core for $k \geqslant 3$.

One way to characterize 2-cores is with a local characterization. That is, describing the structure of G 'near' an arbitrary vertex v.

Theorem 4.1. *A vertex v of G is contained in the 2-core of G if and only if v is on a cycle or v is on a path between vertices of distinct cycles.*

Proof. ($\Leftarrow$) Let v be on a cycle or a path between vertices of distinct cycles. Both such graphs are themselves 2-cores, so v is in the 2-core of G.

($\Rightarrow$) Let v be in the 2-core of G. If v is on a cycle, we are done. If not, then consider a longest path P in the 2-core through v. All the edges incident with v must be bridges, so v is in the interior of P. An end-vertex u of P must have another neighbor, which cannot be a new vertex, so it must be on P. If its neighbor were on the opposite side of v, then v would be on a cycle. Thus its neighbor must be between u and v on P. Repeating this argument for the other end of P shows that v is on a path between vertices on cycles. $\square$

This characterization does not extend easily to higher values of k. The key to the local characterization for the 2-core is the fact that every 2-core contains a simple subgraph that is itself a 2-core. But as we shall see later on, there are arbitrarily large k-cores that do not contain any proper subgraph which is a k-core for $k \geqslant 3$.

It is also possible to offer a more global characterization of the structure of 2-cores.

Corollary 4.1. *A graph G is a 2-core $\Longleftrightarrow$ every end-block of G is 2-connected.*

This leads to another corollary.

Definition 4.1. A block-tree decomposition of a 2-core G is a decomposition of G into 2-connected blocks and trees so that if any tree T is nontrivial, each end-vertex of T is shared with a distinct 2-connected block, if T is trivial, it is a cut-vertex contained in at least two 2-connected blocks, and there are no two disjoint paths between two distinct blocks.

Corollary 4.2. *Every 2-core has a unique block-tree decomposition.*

P r o o f. Let F be the subgraph of a 2-core G induced by the bridges and cut-vertices of G. Then F is acyclic, so it is a forest. Break each component of F into branches at any vertex contained in a component of $G - F$. Also break $G - F$ into blocks, which must be 2-connected. By the previous corollary, each end-vertex of each of the trees must overlap a 2-connected block. If any block contained two end-vertices of the same tree, then there would be a cycle containing edges from the tree. If there were two disjoint paths between two blocks, they would not be distinct. This decomposition is unique because the block decomposition of a graph is unique and any blocks that are K_2 and on a path between 2-connected blocks that does not go through any other 2-connected blocks must be in the same tree. $\square$

These corollaries provide no help when the 2-core in question is itself 2-connected. But there is a well-established description of the structure of 2-connected graphs. See [29], p. 163 and [8], p. 15 for some such results.

We can state an operation characterization of 2-cores. An operation characterization is a rule or rules that can be used to construct all graphs in some class of graphs. An ear of a graph is a maximal path whose internal vertices have degree two.

Theorem 4.2. *A graph G is a connected 2-core $\Longleftrightarrow$ it is contained in the set S whose members can be constructed by the following rules.*

1. *All cycles are in S.*
2. *Given one or two graphs in S, the result of joining the ends of a (possibly trivial) path to it or them is in S.*

P r o o f. ($\Leftarrow$) A cycle has minimum degree 2, and applying step 2 does not create any vertices of lower degree, so a graph in S is a 2-core.

($\Rightarrow$) This is clearly true if G has order 3. Assume the result holds for orders up to r, and let G have order $r + 1$. Let P be an ear or cut-vertex of G. Making $P = K_2$ is only necessary when G has minimum degree at least 3 and is 2-connected. In this case, edges can be deleted until one of these conditions fails to hold. Then if P has internal vertices, deleting them results in a component or components with order at most r. The same is true if P is a cut-vertex, and G is split into blocks. Then the result follows by induction. $\square$

We can also describe 2-monocore graphs by an operation characterization.

Theorem 4.3. *The set of connected 2-monocore graphs is equivalent to the set S of graphs that can be constructed using the following rules.*

1. *All cycles are in S.*

2. *Given one or two graphs in S, the graph H formed by identifying the ends of a path of length at least two with vertices of the graph or graphs is in S.*
3. *Given a graph G in S, form H by taking a cycle and either identifying a vertex of the cycle with a vertex of G or adding an edge between one vertex in each.*

P r o o f. ($\Leftarrow$) We first show that if G is in S, then G is 2-monocore. Certainly cycles are 2-monocore. Let H be formed from G in S by applying rule 2. Then H has minimum degree 2 and since G is 3-core-free and internal vertices of the path have degree 2, H is also 3-core-free. Thus H is 2-monocore. The same argument works for adding a path between two graphs. Let H be formed from G in S by applying rule 3. Then H has minimum degree 2 and since G is 3-core-free and all but one vertex of the cycle have degree 2, H is also 3-core-free. Thus H is 2-monocore.

($\Rightarrow$) We now show that if G is 2-monocore, it is in S. This clearly holds for all cycles, including C_3, so assume it holds for all 2-monocore graphs of order up to r. Let G be 2-monocore of order $r+1$ and not a cycle. Then G has minimum degree 2, so it has a vertex v of degree 2. Then v is contained in P, an ear of length at least 2, or C, a cycle which has all but one vertex of degree 2.

Case 1. G has an ear P. If $G-P$ is disconnected, then the components of G are 2-monocore, and hence in S. Then G can be formed from them using rule 2, so G is in S. If $G-P$ is connected, then it is 2-monocore, and hence in S. Then G can be formed from $G-P$ using rule 2, so G is in S.

Case 2. We may assume that G has no such ear P. Then G has a cycle C with all but one vertex of degree 2, and one vertex u of degree more than 2. If u has degree at least 4 in G, then let H be formed by deleting all the vertices of C except u. Then H is 2-monocore, and G can be formed from it using rule 3. If $d(u)=3$, then its neighbor not in the cycle has degree at least three, so $G-C$ is 2-monocore, and G can be formed from it by using rule 3. □

We can similarly describe the structure of 2-shells.

Corollary 4.3. *The set of* 2*-shells is equivalent to the set S' of graphs constructed using the following rules.*

1. *All graphs in the set S from Theorem* 4.3 *and all* 3*-cores are in S'.*
2. *Given one or two graphs in S', the graph H formed by identifying the ends of a path of length at least two with vertices of the graph or graphs is in S'.*
3. *Given a graph G in S', form H by taking a cycle and either identifying a vertex of the cycle with a vertex of G or adding an edge between one vertex in each.*

Finally, delete the 3*-cores (keeping boundary vertices).*

The proof is essentially the same as that of the previous theorem.

For $k \geqslant 3$, only scattered results on the structure of k-cores exist. Dirac [14] showed that every 3-core contains a subdivision of K_4. For 4-cores, we have the fact ([8], p. 113) that if G is a 4-core other than K_5, then G contains two independent cycles.

5. Applications of monocore graphs

Some results naturally break into cases depending on whether a graph is monocore.

The core number bound is useful in proving an upper bound for the chromatic number in terms of order and size only. This theorem is due to Coffman, Hakimi, and Schmeichel [13]. The proof below is a simplification of the original proof.

Theorem 5.1. *Let G be connected with a 2-core and order n, size m. If the 2-core of G is not a clique or an odd cycle, then*

$$\chi(G) \leqslant \left\lfloor \frac{3+\sqrt{1+8(m-n)}}{2} \right\rfloor.$$

P r o o f. Let $d = \widehat{C}(G)$. Deleting the 1-shell of G leaves $m-n$ unchanged, so we may assume G is a 2-core. If G is an even cycle then the bound is 2, which is exact. If G is any other 2-monocore graph, then $m \geqslant n+1$ and the bound is at least 3, so the inequality holds. Now we may assume $d \geqslant 3$, and we wish to show that $m \geqslant n + \binom{d}{2}$. Since G is not complete, $d \leqslant n-2$.

Let H be the maximum core of G with order $r \geqslant d+1$ and size at least $\frac{1}{2}r \cdot d$. If G is d-monocore, then

$$m \geqslant \left\lceil \frac{n \cdot d}{2} \right\rceil = n + \left\lceil \frac{n(d-2)}{2} \right\rceil \geqslant n + \left\lceil \frac{(d+2)(d-2)}{2} \right\rceil \geqslant n + \binom{d}{2}.$$

If G is not monocore, the size of $G-H$ is at least $n-r+1$ by Proposition 2.3. Then

$$m \geqslant n-r+1+\frac{r \cdot d}{2} \geqslant n+1+\frac{r(d-2)}{2} \geqslant n+1+\frac{(d+1)(d-2)}{2} = n + \binom{d}{2}.$$

Then $d^2 - d - 2(m-n) \leqslant 0$, so by the core number bound,

$$\chi(G) \leqslant 1+d \leqslant \frac{3+\sqrt{1+8(m-n)}}{2}.$$

□

There is another upper bound for the chromatic number worth considering. It involves the eigenvalues of a graph. Any graph can be represented by its (square) adjacency matrix, and its eigenvalues can be computed. The spectrum of a graph is defined to be the sequence of eigenvalues $\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_n$, so that $\lambda_1 = \lambda_1(G)$ is the largest eigenvalue of the graph. The following theorem is due to Wilf [30].

Theorem 5.2 (The eigenvalue bound). *Let G be a connected graph. Then $\chi(G) \leqslant 1 + \lambda_1$, with equality exactly for complete graphs and odd cycles.*

The following results on eigenvalues of graphs are useful. See [25] for background.

Theorem 5.3 (Properties of eigenvalues of graphs).

(a) *Let $\bar{d}$ be the average degree of G and Δ its maximum degree. Then $\bar{d} \leqslant \lambda_1 \leqslant \Delta$ with equality in both cases exactly when G is regular.*

(b) *If G is connected, $\lambda_2 < \lambda_1$. If H is an induced subgraph of G, then $\lambda_1(H) < \lambda_1(G)$.*

We can show that the core number bound is superior to the eigenvalue bound and determine the extremal graphs. The proof uses monocore graphs.

Theorem 5.4. *A connected graph G has $1 + \widehat{C}(G) \leqslant 1 + \lambda_1(G)$, with equality exactly for regular graphs.*

Proof. Let $k = \widehat{C}(G)$, $H = C_k(G)$. Then $\widehat{C}(G) = \widehat{C}(H) = \delta(H) \leqslant \lambda_1(H) \leqslant \lambda_1(G)$.

If G is regular, then $\widehat{C}(G) = \delta(G) = \lambda_1(G) = \Delta(G)$. Assume G is nonregular. Suppose first that G is monocore. Then $\widehat{C}(G) = \delta(G) < \bar{d}(G) < \lambda_1(G)$. Next suppose that G is not monocore. Then $\widehat{C}(G) = \widehat{C}(H) = \delta(H) \leqslant \lambda_1(H) < \lambda_1(G)$, so the result holds in either case. □

Acknowledgement. Thanks to Allen Schwenk for suggesting the topic of k-cores of graphs for me to research and for being my doctoral advisor.

References

[1] *M. Altaf-Ul-Amin, K. Nishikata, T. Koma, T. Miyasato, Y. Shinbo, M. Arifuzzaman, C. Wada, M. Maeda, et al.*: Prediction of protein functions based on k-cores of protein-protein interaction networks and amino acid sequences. Genome Informatics *14* (2003), 498–499.

[2] *J. Alvarez-Hamelin, L. Dall'Asta, A. Barrat, A. Vespignani*: k-core decomposition: a tool for the visualization of large scale networks. Advances in Neural Information Processing Systems 18, 2006, pp. 41.

[3] *G. Bader, C. Hogue*: An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics *4* (2003).

[4] *V. Batagelj, M. Zaversnik*: An $O(m)$ algorithm for cores decomposition of networks. Unpublished manuscript: http://vlado.fmf.uni-lj.si/pub/networks/doc/cores/cores.pdf, 2003.

[5] *A. Bickle*: Structural results on maximal k-degenerate graphs. Discuss. Math., Graph Theory *32* (2012), 659–676.

[6] *A. Bickle*: The k-Cores of a Graph. Ph.D. Dissertation, Western Michigan University, 2010.

[7] *A. Bickle, B. Phillips*: t-Tone colorings of graphs. Submitted.

[8] *B. Bollobas*: Extremal Graph Theory. Academic Press, 1978. zbl

[9] *Y. Caro, Y. Roditty*: On the vertex-independence number and star decomposition of graphs. Ars Combin. *20* (1985), 167–180. zbl

[10] *Y. Caro, Y. Roditty*: A note on the k-domination number of a graph. Internat. J. Math. Math. Sci. *13* (1990), 205–206. zbl

[11] *G. Chartrand, L. Lesniak*: Graphs and Digraphs (4th ed.). Chapman & Hall, Bocca Raton, FL, 2005. zbl

[12] *G. Chartrand, P. Zhang*: Chromatic Graph Theory. Chapman & Hall, Bocca Raton, FL, 2009. zbl

[13] *W. C. Coffman, S. L. Hakimi, E. Schmeichel*: Bounds for the chromatic number of graphs with partial information. Discrete Math. *263* (2003), 47–59. zbl

[14] *G. A. Dirac*: A property of 4-chromatic graphs and some remarks on critical graphs. J. Lond. Math. Soc. *27* (1952), 85–92. zbl

[15] *G. A. Dirac*: Homomorphism theorems for graphs. Math Ann. *153* (1964), 69–80. zbl

[16] *P. Erdős, A. Rubin, H. Taylor*: Choosability in graphs. Combinatorics, Graph Theory and Computing, Proc. West Coast Conf., Arcata/Calif., 1979. Utilitas Mathematica Publishing, Winnipeg, 1980, pp. 125–157. zbl

[17] *M. Gaertler, M. Patrignani*: Dynamic analysis of the autonomous system graph. Proc. 2nd International Workshop on Inter-Domain Performance and Simulation, 2004, pp. 13–24.

[18] *T. W. Haynes, S. T. Hedetniemi, P. J. Slater*: Fundamentals of Domination in Graphs. Marcel Dekker, New York, 1998. zbl

[19] *C. Jianxiang, S. Minyong, M. Young Sohn, Y. Xudong*: Domination in graphs of minimum degree six. J. Appl. Math. & Informatics *26* (2008), 1085–1100.

[20] *D. R. Lick, A. T. White*: k-degenerate graphs. Canad. J. Math. *22* (1970), 1082–1096. zbl

[21] *T. Łuczak*: Size and connectivity of the k-core of a random graph. Discrete Math. *91* (1991), 61–68. zbl

[22] *W. McCuaig, B. Shepherd*: Domination in graphs with minimum degree two. J. Graph Theory *13* (1989), 749–762. zbl

[23] *O. Ore*: Theory of Graphs. Amer. Math. Soc. Colloq. Publ., Vol. 38, Amer. Math. Soc., Providence, RI, 1962. zbl

[24] *B. Reed*: Paths, stars, and the number three. Combin. Probab. Comput. *5* (1996), 277–295. zbl

[25] *A. J. Schwenk, R. J. Wilson*: Eigenvalues of graphs. Selected Topics in Graph Theory (L. W. Beineke, R. J. Wilson, eds.). Academic Press, London, 1978, pp. 307–336. zbl

[26] *S. B. Seidman*: Network structure and minimum degree. Social Networks *5* (1983), 269–287.

[27] *M. Y. Sohn, Y. Xudong*: Domination in graphs of minimum degree four. J. Korean Math. Soc. *46* (2009), 759–773. zbl

[28] *G. Szekeras, H. S. Wilf*: An inequality for the chromatic number of a graph. J. Comb. Th. *4* (1968), 1–3. zbl

[29] *D. West*: Introduction to Graph Theory, (2nd ed.). Prentice Hall of India, New Delhi, 2001. zbl

[30] *H. S. Wilf*: The eigenvalues of a graph and its chromatic number. J. Lond. Math. Soc. *42* (1967), 330–332. zbl

[31] *S. Wuchty, E. Almaas*: Peeling the yeast protein network. Proteomics *5* (2005), 444–449.

[32] *H. Xing, L. Sun, X. Chen*: Domination in graphs of minimum degree five. Graph. Combinator. *22* (2006), 127–143. zbl

Author's address: *Allan Bickle*, Western Michigan University, Department of Mathematics, 1903 W. Michigan, Kalamazoo, MI, 49008, USA, e-mail: `allan.e.bickle@wmich.edu`.