Published in "Physics Reports 832(): 1–32, 2019"
which should be cited to refer to this work.

k-core: Theories and applications☆

Yi-Xiu Kong [a,b], Gui-Yuan Shi [a,b,*], Rui-Jie Wu [b], Yi-Cheng Zhang [b]

[a] *Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian 233003, PR China*
[b] *Department of Physics, University of Fribourg, Fribourg 1700, Switzerland*

Keywords:
k-core
k-shell
Coreness
Critical phenomenon
Phase transition

A B S T R A C T

With the rapid development of science and technology, the world is becoming increasingly connected. The following dire need for understanding both the relationships amongst individuals and the global structural characteristics brings forward the study of network sciences and many interdisciplinary subjects in recent years. As a result, it is crucial to have methods and algorithms that help us to unveil the structural properties of a network. Over the past few decades, many essential algorithms have been developed by scientists from many different fields. This review will focus on one of the most widely used methods called the *k*-core decomposition. The *k*-core decomposition is to find the largest subgraph of a network, in which each node has at least *k* neighbors in the subgraph. The most commonly used algorithm to perform *k*-core decomposition is a pruning process that to recursively remove the nodes that have degrees less than *k*. The algorithm was firstly proposed by Seidman in 1983 and soon became one of the most popular algorithms to detect the network structure due to its simplicity and broad applicability. This algorithm is widely adopted to find the densest part of a network across a broad range of scientific subjects including biology, network science, computer science, ecology, economics, social sciences, etc., so to achieve the vital knowledge under different contexts. Besides, a few physicists find that an exciting phase transition emerges with various critical behaviors during the pruning process. This review aims at filling the gap by making a comprehensive review of the theoretical advances on *k*-core decomposition problem, along with a review of a few applications of the *k*-core decomposition from many interdisciplinary perspectives.

Contents

☆ Authors are listed in alphabetical order.

* Corresponding author at: Department of Physics, University of Fribourg, Fribourg 1700, Switzerland.
E-mail address: guiyuan.shi@unifr.ch (G.-Y. Shi).

1. Introduction

In the recent years, complex networks are widely used to model the real-world systems that are composed of interacting individuals [1–8]. By exploring the structural characteristics of the network, we are able to have an in-depth understanding of the properties of the real-world systems. k-core decomposition, is one of the most widely accepted algorithms due to its linear time complexity [9] and intuitive characteristics. In general, the k-core of a network is the maximal subgraph in which each node has at least k connections to other nodes in the subgraph, despite how many links we have outside the subgraph. The idea of k-core can be traced back to Erdős and Hajnal [10] in 1966, they proposed that the coloring number of a graph G to be the least k for which there exists an ordering of the nodes of G in which each node has fewer than k neighbors that are earlier in the ordering. An equivalent expression called the degeneracy was later defined by Lick and White in 1970 [11], they defined that the degeneracy of a graph G is the least k such that for each induced subgraph of G, at least one node in the subgraph has k or fewer neighbors.

The commonly accepted concept of k-core was first proposed by Seidman [12] and Seidman also derived an algorithm called the k-core pruning process to obtain the k-core of a given network, which is to remove the nodes that have degree less than k recursively. Here we show a simple illustration of the concepts in Fig. 1.

With this method, the densely connected area can be identified and in order to be included in the k-core, a node must have at least k links to other nodes in the k-core, regardless of how many other nodes they are connected to outside the k-core. Another closely related concept is called the k-shell [13], which is defined as the group of nodes that belong to k-core but not belong to $(k+1)$-core. Also, a very similar concept that is widely used, the coreness of a node [14]. The statement that the coreness of a node equals k, is equivalent to the statement that the node is in the k-shell of the network. As the k-core pruning process is both simple (linear time complexity) and intuitive, later on, the concept of k-core has become surprisingly popular, and widely been applied in many scientific fields.

In the researches of the k-core, people come to notice the emergence of criticality of the k-core pruning process. The only two outcomes after the pruning process are either the network disappears and no node survives the pruning, or no nodes can be further removed so that the network finally has a fix-sized k-core. Researchers [15–20] find that whether or not the network has a k-core remaining after the pruning process very much depends on the density of the network, and there exists a specific criterion of the initial density of the network that controls the existence of the k-core. Many mathematicians and physicists have made essential contributions in solving the problem theoretically in the past decades [15,18,19,21–23]. Recently, it is reported that the exact analytical result has been obtained [20,24,25]. The k-core pruning process is among the few examples that the precise critical behavior is presented in detail and solved analytically. The analytical solution of k-core pruning process provides new insights to study the critical phenomena that attract so many scientists. In this review we give a summary of the related researches in this direction, along with a survey of some important applications of k-core on various research fields.

The review article will be organized as follows. Section 2 will review the theoretical researches of the critical behavior in k-core decomposition, with special emphasis on the theoretical framework to solve the problem and the exact analytical solution that describes the entire pruning process. In Section 3, we will review the applications of k-core decomposition in many different scientific fields, to show how this method is used to uncover the underlying information in various kinds of real systems, as well as multiple variations of the standard k-core decomposition that have been extended to many different types of networks, including the weighted networks, directed networks, multi-layer networks, dynamic networks and also. Many of the research papers that will be covered in this section already made a great impact in their own disciplines by introducing k-core decomposition as a tool to unveil the underlying information. To conclude, Section 4 will summarize the most important messages of this review, and outline the future challenges related to this research topic.

2. Theoretical studies of k-core

2.1. Background concepts

Before we start to survey the researches, we will briefly introduce some fundamental concepts that we will frequently use in the numerous theoretical studies of the k-core, to facilitate the reading throughout the theoretical section in this review.

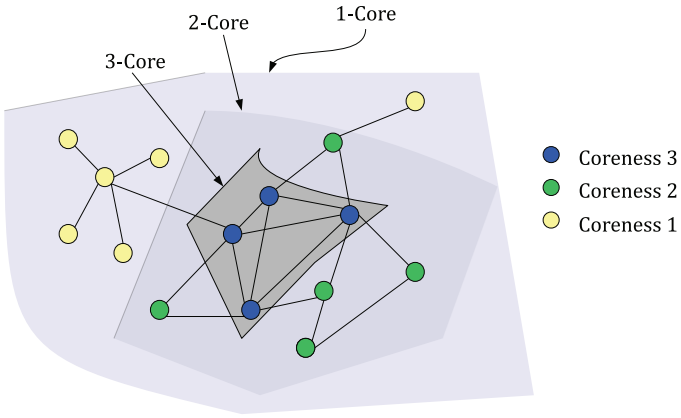


Fig. 1. An illustration of k-core and coreness. k-shell consists of the nodes that have coreness equal to k. *Source:* Reproduced from Ref. [26].

2.1.1. Degree distribution

The degree of a node in a network is the number of nodes connected to the node. We define the degree distribution p_k to be the fraction of nodes in a network that have degree k. That is to say if there are n nodes in total in a network and n_k of them have degree k, we have $p_k = n_k/n$. Obviously, the sum of all the p_k must be equal to 1, $\sum_{k=0}^{\infty} p_k = 1$.

On the other hand, we can also regard the value p_k as the probability that a randomly chosen node in the network has degree k. This point of view will be useful in our theoretical studies throughout the whole review.

2.1.2. Excess degree distribution

The degree distribution describes the statistical property of a node, however, sometimes we need more than just node to understand the network. As an example among many, the excess degree distribution is introduced below because it is one of the most important concepts in the study of k-core, it is related to the majority of the contents in the following sections. We define the excess degree distribution q_j to be that, upon following a randomly chosen Link, the distribution of the number of node's other links except the link we arrived by Ref. [27]. Through this definition, one can easily obtain the following equation:

$$q_j = \frac{(j+1)p_{j+1}}{c}, \tag{1}$$

here c is the average degree: $c = \sum_{k=0}^{\infty} k p_k$.

2.1.3. Generating functions

Suppose we have a non-negative integer variable, such as the degree or the excess degree that we have introduced above, and naturally we can have the corresponding probability distribution of these quantities. As a usual treatment, we can define such a generating function that is a polynomial series whose coefficients are the probabilities p_k or q_k. Therefore, we can obtain the following generating functions [27] for degree distribution:

$$G_0(z) = p_0 + p_1 z + p_2 z^2 + \cdots + p_n z^n + \cdots = \sum_{k=0}^{\infty} p_k z^k, \tag{2}$$

and for excess degree distribution:

$$G_1(z) = q_0 + q_1 z + q_2 z^2 + \cdots + q_n z^n + \cdots = \sum_{k=0}^{\infty} q_k z^k. \tag{3}$$

Obviously, $G_0(1) = 1$ and $G_1(1) = 1$, $c = G_0'(1)$. Since the coefficients of $G_1(z)$ are determined by the coefficients of $G_0(z)$, they are not actually independent. Meanwhile the relationship between these two functions can obtain by the following:

$$G_1(z) = \sum_{k=0}^{\infty} q_k z^k = \sum_{k=0}^{\infty} \frac{1}{c}(k+1)p_{k+1} z^k = \frac{1}{c}\sum_{k=1}^{\infty} k p_k z^{k-1} = \frac{1}{c}\frac{dG_0(z)}{dz}. \tag{4}$$

Since $c = G_0'(1)$, we have the relationship:

$$G_1(z) = \frac{G_0'(z)}{G_o'(1)}. \tag{5}$$

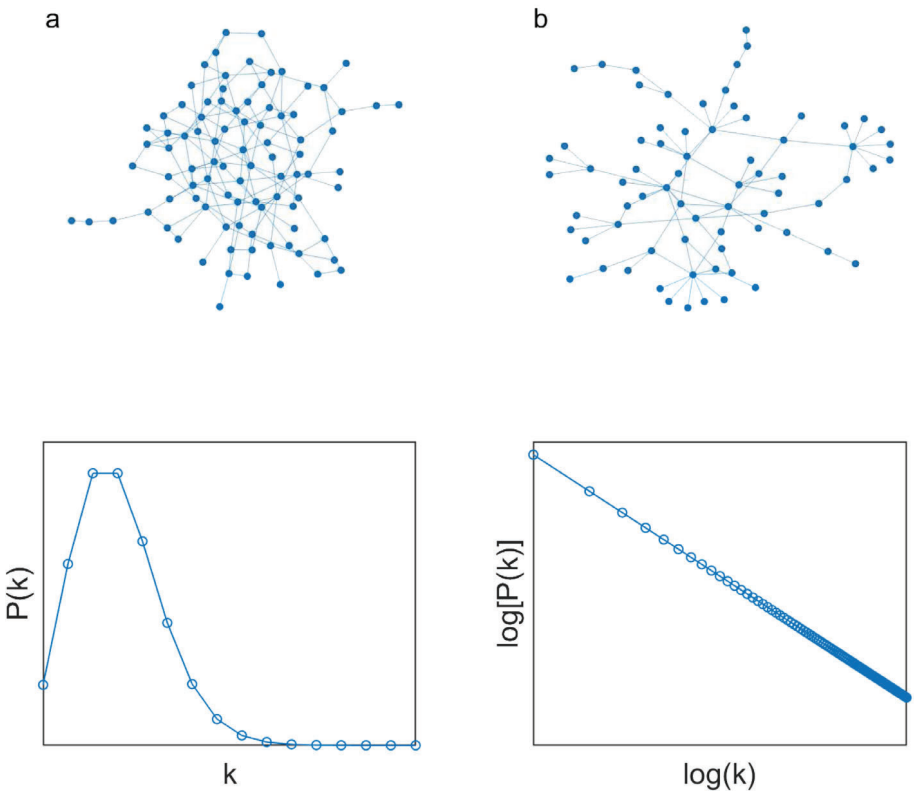


Fig. 2. (a) An Erdős–Rényi network and its degree distribution. (b) A scale-free network and its degree distribution.

The generating function of Erdős–Rényi networks. In an Erdős–Rényi network (ER network) [28], a node is connected to $n-1$ other nodes with a given probability p. Its degree distribution follows a binomial distribution:

$$p_k = \binom{n-1}{k} p^k (1-p)^{n-1-k}, \tag{6}$$

and its average degree is

$$c = \sum_{k=1}^{n-1} k p_k = (n-1)p. \tag{7}$$

For large n, it is easy to prove that the degree distribution can be written as $p_k = \frac{e^{-c}c^k}{k!}$, which is a Poisson distribution. Thus, the generating functions of ER-network can be written as:

$$G_0(z) = \sum_{k=0}^{\infty} \frac{e^{-c}c^k}{k!} z^k = e^{c(z-1)}, \tag{8}$$

and

$$G_1(z) = \frac{G_0'(z)}{G_0'(1)} = \frac{c \cdot e^{c(z-1)}}{c} = e^{c(z-1)}, \tag{9}$$

which happens to be the same in this case. In Fig. 2(a) we show an example of the degree distribution of ER network.

The generating function of scale-free networks. A scale-free network (SF network) [29], also called the Barabási–Albert (BA) model, is a network whose degree distribution follows a power law distribution:

$$p_k = \frac{k^{-\gamma}}{\zeta(\gamma)} \qquad (k \geq 1) \tag{10}$$

here, $\gamma > 2$, $\zeta(\gamma)$ is the Riemann zeta function: $\zeta(\gamma) = \sum_{k=1}^{\infty} k^{-\gamma}$, and for $k = 0$, $p_0 = 0$. We can give the generating function for the degree distribution:

$$G_0(z) = \sum_{k=1}^{\infty} \frac{k^{-\gamma}}{\zeta(\gamma)} z^k = \frac{\mathrm{Li}_\gamma(z)}{\zeta(\gamma)}, \tag{11}$$

here, $\mathrm{Li}_\gamma(z)$ is the polylogarithm function: $\mathrm{Li}_\gamma(z) = \sum_{k=1}^{\infty} z^k/k^\gamma$. Obviously, $\mathrm{Li}_\gamma(1) = \zeta(\gamma)$. In addition, we can obtain the average degree of the network by taking the derivation of the above generating function at $z = 1$:

$$c = G_0'(1) = \frac{\mathrm{Li}_{\gamma-1}(z)}{z \cdot \zeta(\gamma)}\Big|_{z=1} = \frac{\zeta(\gamma-1)}{\zeta(\gamma)} = \frac{\mathrm{Li}_{\gamma-1}(1)}{\mathrm{Li}_\gamma(1)}. \tag{12}$$

Finally, the generating function for the excess degree distribution can be written as follows:

$$G_1(z) = \frac{G_0'(z)}{G_o'(1)} = \frac{\mathrm{Li}_{\gamma-1}(z)}{\zeta(\gamma-1)}. \tag{13}$$

In Fig. 2(b) we show an example of the degree distribution of scale-free network.

The correlated networks. All the previous definitions or models are used to describe the random networks. However, in reality, most of the networks, are not completely random. Many networks show "assortative mixing", that means, a preference that a high-degree node tends to attach to other high-degree nodes, while a few other networks show "disassortative mixing", which means high-degree nodes tend to connect low-degree nodes. We often referring these characteristics as 'correlated', in contrast to 'uncorrelated'.

In the year of 2002, M. Newman [30] defined the quantity e_{ij} to be the joint probability distribution of the excess degrees of the two nodes at either end of a randomly chosen link, that is to say, given a randomly chosen link, the probability that a node at one end of the link has a degree of $i + 1$ and the other node on the other end of the link has a degree of $j + 1$. Obviously, e_{ij} satisfies the following rules:

$$\sum_{i=0}^{\infty}\sum_{j=0}^{\infty} e_{ij} = 1, \tag{14}$$

and

$$\sum_{i=0}^{\infty} e_{ij} = q_j. \tag{15}$$

Besides, for undirected networks that we focus in the review, e_{ij} should be symmetric in its indices: $e_{ij} = e_{ji}$

In particular, in a network with no assortative mixing (also called the degree–degree correlation), $e_{ij} = q_i \cdot q_j$. Such a network is called an uncorrelated network. And if there exists assortative mixing, e_{ij} will differ from this value and the level of assortative mixing can be quantified by the degree–degree correlation function: $\langle ij\rangle - \langle i\rangle\langle j\rangle = \sum_{ij} ij \cdot (e_{ij} - q_i q_j)$.

Consider the extreme case that the correlation function takes its maximum value, which it achieves on a perfectly assortative network, i.e. $e_{ij} = q_j\delta_{ij}$. This value is equal to $\sigma_p^2 = \sum_i i^2 q_i - (\sum_i i q_i)^2$. It is convenient to normalize the correlation function by the maximal value:

$$r = \frac{1}{\sigma_p^2}\sum_{ij} ij \cdot (e_{ij} - q_i q_j), \tag{16}$$

which is the Pearson correlation coefficient of the degrees at either ends of a link and it satisfies $-1 \le r \le 1$.

2.2. The theoretical studies of the k-core

The theoretical study of k-core mainly wants to answer such questions: Given an initial network, will the network have a remaining k-core after the pruning process? Where is the corresponding critical point? What is this critical behavior? If k-core exists, what is its network structure?

Although in reality most networks are not random, to answer the above questions, we first consider a simple situation, the large random networks. We will introduce the researches on correlated (not random) networks later. The statistical properties of a random network are determined by the degree distribution of the network, i.e., a random network with the same degree distribution and the same density of degree can be regarded as indistinguishable. So given an initial network, it usually means that our starting point is a large uncorrelated network with a known degree of distribution and density. In the following we will present the most important researches that advances the theoretical studies of k-core on large uncorrelated (random) networks.

2.2.1. k-core on large uncorrelated networks

In 1987, Luczak [15] studied the size of the k-core of the random networks $G(n, p)$ ($G(n, p)$ is a network with n nodes in which each pair of nodes is connected independently with probability $p = p(n)$). Denote the number of the nodes in k-core by $v(k, n, p)$, he obtained the following three theorems in this paper.

Theorem 1. *If $k \geq 3$, with probability tending to 1 either $v(k, n, p) = 0$ or $v(k, n, p) \geq 0.0002n$ as $n \to \infty$.*

Theorem 2. *For every $\epsilon > 0$, there exists a constant d that for $c > d$ and $k < c - c^{0.5+\epsilon}$, $v(k, n, p) \geq n - n \cdot exp(-c^{\epsilon})$ with probability tending to 1 as $n \to \infty$.*

Theorem 3. *For every $\epsilon > 0$, there exists a constant d that for every k and $c > d$, with probability tending to 1, either $v(k, n, p) = 0$ or $v(k, n, p) > n - nc^{-0.5+\epsilon}$ as $n \to \infty$.*

The first theorem states that for $k \geq 3$, the number of the nodes in the k-core is either 0 or larger than $0.0002n$. That suggests the existence of a discontinuous transition. From the rest two theorems we know that in some conditions while the average degree c is large enough, the size of k-core is either 0 or very large (almost the whole network), and also imply that the transition is discontinuous. To the best of our knowledge, this paper is the first attempt to deal with the k-core problem theoretically.

In 1996, Pittel et al. [16] found the exact solution of the transition point of the k-core of $G(n, p)$. Given $\lambda > 0$, let $Z(\lambda)$ be a Poisson distributed random variable with mean λ. Then they introduce $p_k(\lambda) = \text{Probability}(Z(\lambda) \geq k)$ and $\pi_k(\lambda) = \text{Probability}(Z(\lambda) \geq k - 1)$. Here $k \geq 3$ is fixed integer. Define $\gamma = \inf\{\lambda/\pi_k(\lambda) : \lambda > 0\}$. It is easy to see that the function $\lambda/\pi_k(\lambda)$ approaches to ∞ as $\lambda \to 0$, thus γ_k is attained at $\lambda_k > 0$. Clearly, when $c < \gamma_k$, the equation $c = \lambda/\pi_k(\lambda)$ has no root when $c > \gamma_k$. There are two roots. Denote the larger root by $\gamma_k(c)$. Pittel et al. obtained the following theorems:

Theorem 4. *Given $\delta \in (0, 0.5)$, suppose $c \leq \gamma_k - n^{-\delta}$. Let $\epsilon \in (0, 1)$ be chosen arbitrarily small. The probability that $G(n, p)$ has a k-core with at least ϵn nodes is $O(exp(-n^{\rho}))$, $\forall \rho < (0.5 - \delta)^{1/6}$. The probability that there is a k-core of any size($\geq k+1$) is $O(n^{-(k-2)(k+1)/2})$.*

Theorem 5. *Given $\delta \in (0, 0.5)$, suppose $c \geq \gamma_k + n^{-\delta}$. Fix $\sigma \in (3/4, 1 - \delta/2)$ and define $\bar{\zeta} = min\{2\sigma - 3/2, 1/6\}$. $\forall \zeta < \bar{\zeta}$, with probability $\geq 1 - O(exp(-n^{\zeta}))$, $G(n, p)$ contains a giant k-core of size $np_k(\lambda_k(c)) + O(n^{\sigma})$.*

From these two theorems, they implied that while $k \geq 3$, there exists a sudden emergence of a giant k-core which the size is np_k, when the number of the links increases from smaller than c_k to larger.

In 2003, Fernholz et al. [17] obtained the results of k-core on random graph with degree sequences that are smooth and sparse, that means we can use degree distribution to describe the degree sequences and the average degree is finite. They stated that the existence of a giant k-core is related to the probability that the genealogy tree of a certain branching process contains a perfect infinite $(k - 1)$-ary subtree. They obtained the final results by using generating function. Their results show that for any such random graph, the 2-core exists almost surely if and only if it has a giant component. They also apply their results to power law distribution networks, and find the thresholds of the existence of the k-core.

In 2005, Schwarz [18] solved the k-core percolation on Bethe latices. At first the nodes in the Bethe lattice are independently occupied with probability p, after that, they eliminate the occupied nodes with fewer than k neighboring occupied nodes step by step until all surviving nodes (if any) have at least k surviving neighbors. They solved the problem by considering the half Bethe lattice, and found the recursion relations for two quantities at level $n + 1$ in terms of quantities at level n. By analyzing these two quantities, finally they obtained the size of k-core.

In the same year, S. N. Dorogovtsev [22] solved the k-core percolation on random uncorrelated networks. Their model is also at first to randomly remove some nodes of such a network with the probability $(1 - p)$, and then find the size of k-core in the remaining subgraph. They also took into account the treelike structure of the infinite sparse network and concluded that the existence of k-core coincides with the infinite $(k - 1)$-ary subtree. Also in Ref. [19], they demonstrated that the so called 'corona' of the k-core that is a subset of nodes in the k-core that have exactly k-neighbors in the k-core plays a crucial role. The threshold of the k-core percolation is at the same time the percolation threshold of finite corona clusters. Riordan et al. [23] also considered the k-core of random graph $G(n, \gamma/n)$. Here γ is the average degree of the network. They also considered the branching process, and derived the exact results.

In 2014, N. Azimi-Tafreshi [21], derived the exact self-consistency equations to obtain the birth points of the k-core and their relative sizes for uncorrelated multi-layer networks, and presented the pruning algorithm for the k-core decomposition of multiplex networks.

In 2015, G. J. Baxter [20] presented the theory of the k-core pruning process, derived the exact equations describing this process and solved them numerically. They got four recurrence equations by analyzing the pruning process from nth step to the $(n + 1)$th step. And by the numerical result they find that for the $k(k \geq 3)$-core pruning process of the *ER* network, when a little below the threshold (critical point), a long-lasting transient process occurs, and this transient process ends with a collapse in which the entire network disappears completely. Fig. 3 shows their results.

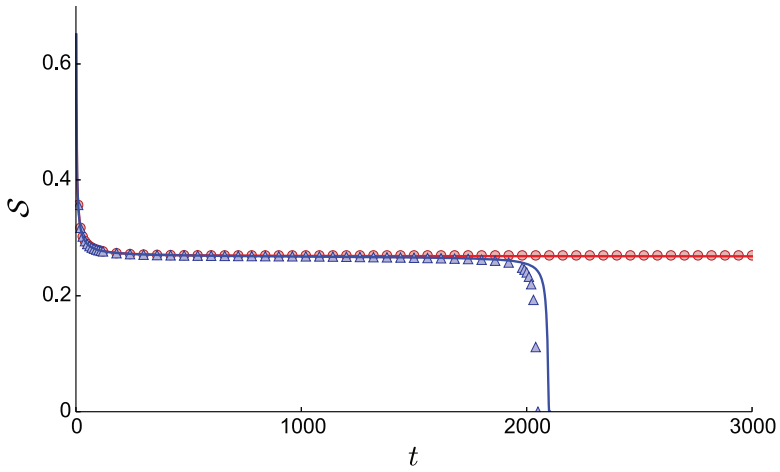


Fig. 3. Size S of the Erdős–Rényi network versus time t during the pruning process for $k = 3$ in two cases. (i) Below the threshold $\langle q\rangle_c$, the system experiences a long "plateau" stage before the collapse in the end. Shown are numerical calculations for mean degree $\langle q\rangle = 3.3509$ (blue solid line) and simulations (triangles) for a network of 10^8 nodes showing similar total time. (ii) Above $\langle q\rangle_c$, the system relaxes to a finite size, numerical solution for $\langle q\rangle = 3.35092$ (red solid line) and simulations (circles).
Source: Reprinted from Ref. [20].

Previously we have shown that there exist many attempts and efforts on solving the k-core pruning process on large uncorrelated networks, among them the work presented by Baxter et al. [20] is especially important and profound. They proposed a set of equations to describe the process and the four equations forge the building block of the analytical solution. While theoretical framework is successful, due to the fundamental difficulty of solving the math, Baxter et al. had not obtained the closed form of the solution of the intermediate states of the pruning process.

Until recently, Shi et al. [24] published a preprint and stated that the closed form of the solution has been obtained, which is an important step forward. Their results show that the four equations proposed by Baxter et al. [20] can be deduced to a univariable iteration and the k-core pruning process can be perfectly mapped into this iteration. They obtained the general form of the iteration in their paper and once given the initial state, each state of the univariable iteration can be easily computed and hence provides the result of the corresponding k-core pruning process. Specifically, if the network is randomly generated and uncorrelated, their results can then give the exact analytical expressions of the size and the structure of the remaining k-core for any intermediate step during the pruning process. The final state can be obtained by extrapolation easily. Their theoretical analysis is further validated by numerical simulations. With the exact solution, they have found the precise behavior of the critical phenomenon observed in the k-core pruning process. The critical exponents can then be naturally obtained, and they also obtain the exact coefficients of the critical power-law relations for ER networks. In the following, we show in detail how the exact solution of k-core pruning process is obtained.

We begin with a brief introduction of the theoretical framework given by Baxter et al. in the previous paper [20]. Now consider the nth pruning process on the network $\mathcal{N}_{n-1}$. Let v_{n-1} be the probability that if we randomly follow a link to one node in $\mathcal{N}_{n-1}$, the node has an excess degree more than $k-2$:

$$v_{n-1} = 1 - \sum_{j=0}^{k-2} q_{n-1,j}. \tag{17}$$

The nodes that will be removed after nth k-core pruning consist of two terms: (1) the nodes that have degree less than k. (2) the nodes that have degree no less than k but their neighbors all have degree less than k. Note that in the original theoretical model proposed in the previous paper [20] the second term was missing by an oversight. Shi et al. pointed out this oversight and proposed the correct equations in their paper:

$$p_{n,0} = \sum_{j=0}^{k-1} p_{n-1,j} + \sum_{j=k}^{\infty} p_{n-1,j}(1 - v_{n-1})^j. \tag{18}$$

The nodes whose degree will be i after nth pruning are the nodes that have degree of j which is no less than $max\{i, k\}$ and $j - i$ neighbors will be removed after nth pruning:

$$p_{n,i} = \sum_{j=max\{i,k\}}^{\infty} p_{n-1,j} \binom{j}{i} v_{n-1}^i (1 - v_{n-1})^{j-i}. \tag{19}$$

And the excess degree distribution after nth pruning can be easily obtained:

$$q_{n,i} = \frac{(i+1)p_{n,i+1}}{\sum_{i=0}^{\infty} ip_{n,i}} \tag{20}$$

In order to solve the equations that was not analytically resolved, Shi et al. proposed a mathematical treatment to simplify the complex infinite-dimensional simultaneous recurrence equations to an equivalent univariable iteration process, by introducing an auxiliary series y_n. Then the desired quantities like the size of the remaining subgraph in nth step S_n as well as its degree distribution, can be obtained and expressed in a simple function of y_n. Here y_n is defined as:

$$y_n = 1 - \sum_{j=0}^{k-2} \frac{y_{n-1}^j}{j!} G_1^{(j)}(1 - y_{n-1}), \tag{21}$$

here $G^{(j)}(z)$ denotes the jth derivative of $G(z)$, i.e., $G^{(j)}(z) = d^j G(z)/dz^j$. $y_0 = 1$ and then each y_n can be obtained by computing a univariable iteration process.

The recurrence relation of $G_{n,0}(z)$ (the degree distribution generating function $G_0(z)$ in the nth step) can be obtained from (18) and (19):

$$G_{n,0}(z) = G_{n-1,0}(1 - v_{n-1} + zv_{n-1}) + \sum_{j=0}^{k-1} p_{n-1,j}(1 - (1 - v_{n-1} + zv_{n-1})^j). \tag{22}$$

Then by induction:

$$G_{n,0}(z) = G_0(1 - y_n + y_n z) + \sum_{j=0}^{k-1} \frac{G_0^{(j)}(1 - y_{n-1})}{j!}(y_{n-1}^j - (y_{n-1} - y_n + y_n z)^j), \tag{23}$$

which can be obtained given y_{n-1} and y_n. Here the nodes in the remaining subgraph after the nth pruning have degrees no less than k in the $\mathcal{N}_{n-1}$ network. The degree distribution can be directly obtained from the generating function $G_{n,0}(z)$, and the size of the remaining subgraph S_n after pruning n is:

$$S_n = \sum_{j=k}^{\infty} p_{n-1,j} = \sum_{j=k}^{\infty} \frac{G_0^{(j)}(1 - y_{n-1})}{j!} y_{n-1}^j = 1 - \sum_{j=0}^{k-1} \frac{G_0^{(j)}(1 - y_{n-1})}{j!} y_{n-1}^j, \tag{24}$$

which is a unary function of y_{n-1}.

Define:

$$f(y) = 1 - \sum_{j=0}^{k-2} \frac{y^j}{j!} G_1^{(j)}(1 - y),$$

and

$$g(y) = 1 - \sum_{j=0}^{k-1} \frac{G_0^{(j)}(1 - y)}{j!} y^j.$$

Eqs. (21) and (24) can be rewritten as:

$$y_n = f(y_{n-1}), \tag{25}$$

and the size of the subgraph S_n is

$$S_n = g(y_{n-1}). \tag{26}$$

In addition, they validate their results in SF networks and ER networks. The numerical simulation results confirm that their analytical results are solid.

2.2.2. k-core on large correlated networks and multi-layer networks

Despite the recent progresses mentioned above, later on, Wu et al. [25] also stated they obtained the exact results of the k-core pruning process, but using a completely different approach from the generating function. The method they used is called the NonBacktracking Expansion Branch (NBEB), which was proposed as a representation of a network in Computer Science [31,32]. They found that the series y_n introduced in Ref. [24] that is used as a pure mathematical trick, now can be endowed with a probabilistic meaning. They also proved that the solution given by the new method is equivalent to the generating function approach, and more importantly, the NBEB approach is found to be capable of dealing with degree–degree correlated networks, which is an unprecedented progress. They also validated their results by numerical simulations on several networks. Following this work, in another paper by Wu et al. [33], they

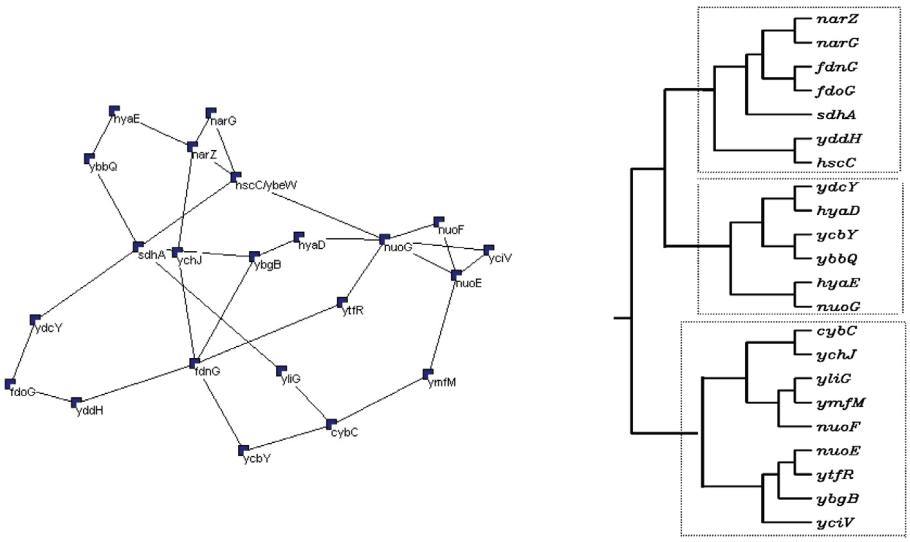


Fig. 4. The Protein–Protein Interaction (PPI) network introduced by Altaf et al. [38] and the classification structure by using *k*-core. *Source:* Reproduced from Ref. [38].

further extended the NBEB method to solve the *k*-core pruning process on multi-layer networks and developed the so-called Multicolor Nonbacktracking Expand Branch (MNEB). This method allows to analyze the results under different circumstances from the simplest uncorrelated multi-layer networks to the most complex situation that both in-layer and inter-layer correlations exist.

Now we have introduced the most important contributions to the body of knowledge that are related to the theoretical analyses of *k*-core problem, mainly focusing on the criticality of the *k*-core. Although difficult, multiple theoretical progresses [34–36] have been made to the center of the problem. Until recently it is reported that the exact analytical results are obtained by using generating functions and also the NonBacktracking Expansion Branch. Especially, with the generating functions, one can obtain the detailed critical behaviors of the *k*-core pruning process, and this result suggests a new approach to analytically study the giant class of critical phenomena [37].

3. Applications of *k*-core decomposition

Above we have reviewed the theoretical advances in analyzing the *k*-core problem. Recent studies in the critical behavior of the *k*-core pruning process have shown significant developments. Meanwhile, due to the simplicity and the effectiveness of the *k*-core decomposition, extensive applications of *k*-core, *k*-shell, or coreness have been widely seen in a variety of scientific fields, including biology, ecology, computer sciences, information spreading, geology and so on. In the following, we will survey some of the progresses in recent years, according to chronological order. We hope the survey will shed lights on the potential directions where *k*-core and its multiple variations can be used as powerful tools.

3.1. The applications in Biology

We start with introducing the *k*-core applications in biology. In 2003, Altaf et al. [38] proposed a procedure to predict the feature of functional-unknown proteins based upon *k*-core and phylogenetic analysis of Protein–Protein Interaction (PPI) networks. They have carried out pull-down assay to establish interactions among the proteins of Escherichia coli (E. coli). 10 238 distinct binary interactions have been actually identified from experiment data. They prepared a list of 1972 function-unknown proteins and then divided the function-known proteins into several groups according to their functionality. As an instance, in this paper they used their method for the group called Electron Transfer (ET) proteins. These proteins are involved in energy metabolism of E. coli. They extracted 193 binary interactions out of 10 238 interactions from the original data. For each of these 193 binary interactions either each proteins are of ET group or one is of ET group and the other one is function unknown. They built a network from the 193 interactions and obtain the largest *k*-core of this system, which is a 2-core and is composed of 22 proteins. Furthermore, they classified these proteins into several trees as shown in Fig. 4. They found that with phylogenetic analysis both *yciV* and *nuoF* interact with *nuoE*, and accordingly they predict that the function of *yciV* is similar to those of *nuoF* and *nuoE*. The *k*-core decomposition in this paper is used as a method to extract the central information, i.e., to remove the extra unimportant links of the PPI network.

Later in 2005, Wuchty et al. [39] found that the probability that a node in protein network is both necessary and evolutionarily conserved, increases continuously to the innermost cores. Although connections alone are usually not

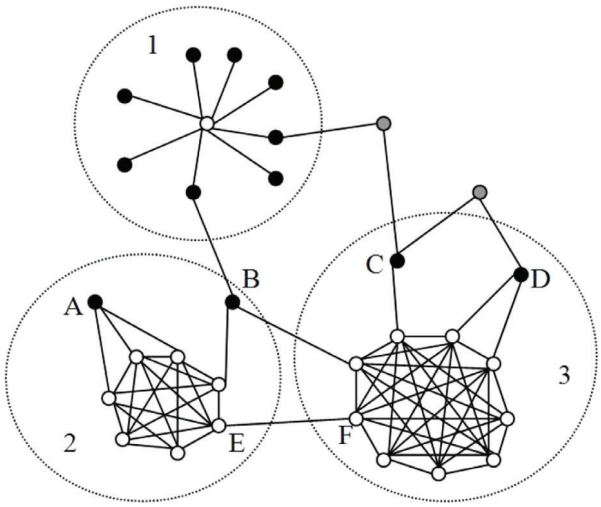


Fig. 5. A sample network consisting of three core/peripheral structures, represented by gray circles (1–3). An empty loop node is a core member. The black and gray nodes represent 1-peripheries and 2-peripheries, respectively. Labeled nodes (AF) are different types of 1-peripheral: (A) closed single-core periphery; (B) multi-core periphery; (C) fully open single-core periphery; (D) limited open single-core periphery; (E) and (F) core Member periphery.
Source: Reproduced from Ref. [40].

sufficient criteria for assessing protein function, evolution, and topological correlation, they classify nodes as global and local centers, depending on appearance in the kernel or outer cores. The observation that global central proteins are involved in a large number of protein complexes indicates that global central proteins can work as a pillar in the evolution of proteomes. Even with the shortcomings of protein interaction data, they have found that the results are very robust for inaccurately determined protein interactions.

Luo et al. [40] systematically explored the core/peripheral structure of the protein interaction network (PIN). They proposed several calculation methods to identify two types of cores from the PIN, namely the *k*-plex core and the star cores. An example is shown in Fig. 5. After applying these methods to the yeast protein interaction network, they successfully identified 110 *k*-plex cores and 109 star cores. The *k*-plex core is one of the largest subgraphs of the network, where each node of the subgraph is connected to at least $n-k$ other nodes, here n is the number of nodes in the induced subgraph. The basic algorithm is done through depth-first search. They found that the *k*-plex core consists of a "party" protein, a "date" protein, or both. They also revealed that there exist two types of 1-peripheral proteins: the "party" periphery, tends to be part of a protein complex, and the "connecter" periphery, tends to be linked to different proteins or protein complexes. Their results also show that in addition to connectivity, other changes in structural properties are associated with changes in biological properties. In addition, they showed a negative correlation between evolution rate and connectivity, and core/peripheral structures help to reveal the existence of multiple levels of protein expression kinetics. Their results indicate that structure and connectivity can be used to characterize the topological properties of protein interaction networks and to elucidate the functional organization of cellular systems.

Schwab et al. [41] studied a small neuronal network that functions in the control of the mammalian breathing rhythm through periodic firing bursts. They used *k*-core to understand these discontinuities in the SO ("stable oscillation") or HA ("High Activity") phase boundaries. They pointed out that as the number of nodes in the Erdős–Rényi random network increases, the *k*-core clusters show a large *k* value at a well-defined threshold. As an example, they showed that the *k* kernel of a symmetric random adjacency matrix in Fig. 6(a), almost all nodes form a five-core cluster. Randomly deleting a node does not change this feature, as shown in Fig. 6(b), but randomly deleting two nodes produces an obvious phase transition process in which the network is dominated by a quad-core cluster of Fig. 6(c). The extra continuation of the deletion of the additional node does not change the main structure of the quad core, as shown in Fig. 6(d). They emphasized that the *k*-core concept does not apply to SO-Q (Q phase is where all neurons are permanently in a state of low activity) transitions. Along the transition line, neurons with the highest connectivity are able to trigger excitation waves that propagate throughout the system. They studied a simple version of the Feldman and Del Negro (FDN) description of a rhythmic neural network in which a combination of excitable integration and evoked neurons modified by activity-mediated slower desensitization is used. They showed that there is asymmetry in the phase diagram between the transition from the stable blasting phase to the stationary phase and the transition from the stable blasting phase to the HA phase. The first transition is well described by the mean field theory, while the staircase structure of the phase boundary of the second transition reflects the network connectivity properties. This asymmetry stems from the difference between voltage-mediated excitation of diffusion wave dynamics and collective desensitization. The asymmetry

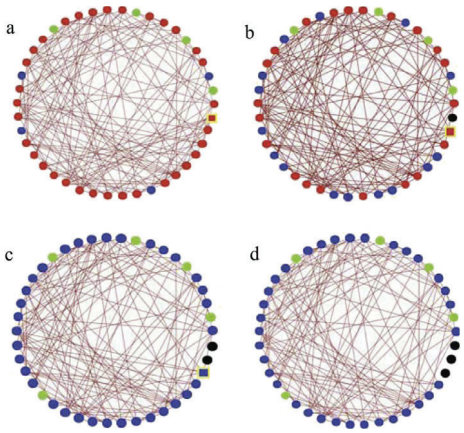


Fig. 6. *k*-cores of a symmetric $N \times N$ random adjacency matrix. Nodes making up the five core are marked in red (medium gray), four-core nodes in blue (dark gray), three-core nodes in green (light gray), while removed nodes are marked in black. The yellow outlined square indicates the node to be removed on the subsequent panel. The four figures (a), (b), (c) and (d) show a progressively decreasing network size: $N_A = 43$, $N_B = 42$, $N_C = 41$, and $N_D = 40$.
Source: Reproduced from Ref. [41].

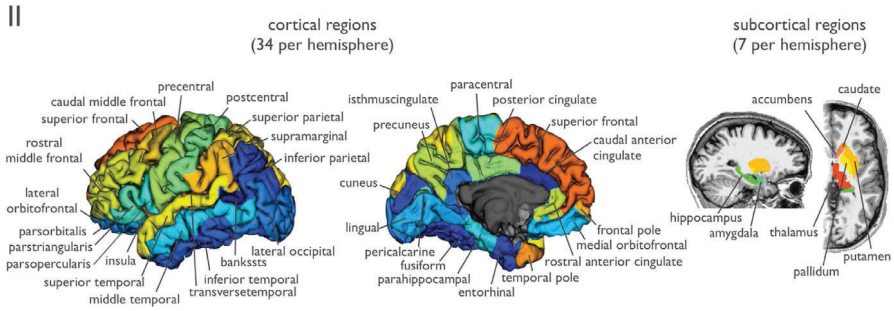


Fig. 7. Regional segmentation scheme. The brain is divided into 82 brain regions, consisting of 68 cortical regions (34 per hemisphere) and 14 subcortical regions (7 per hemisphere).
Source: Reproduced from Ref. [42].

and failure of the mean field theory in the FDN model is due to the fact that it identifies desensitization by the number of action potentials that neurons receive rather than generate.

Heuvel et al. [42] showed that the brain center formed a so-called "rich club", which means the high degree nodes tend to connect with other high degree nodes rather than the fewer linked nodes, revealing important topological information of the brain network. The whole brain structure network of 21 subjects was reconstructed using diffusion tensor imaging data. By examining the connectivity characteristics of these networks, they found a set of 12 strongly interconnected bihemispheric central regions, including the forelimb, the upper frontal and superior parietal cortex, and the subcortical hippocampus, putamen and thalamus, as shown in Fig. 7. Importantly, it has been found that the density of these hub areas is more intensively connected than that would be expected based solely on their degree to form a rich club. They discussed the potential function of the rich club organization on the human connectome, especially considering its role in information integration and the robustness of its structural core. They computed the characteristic metrics of the network organization, including (node-specific) degree *k*, clustering coefficient, characteristic path length, betweenness centrality, normalized clustering coefficient, and normalized path length (both relative to a group of 100 comparable random graphs), global efficiency, assortativity and modularity, with the Brain Connectivity Toolbox as previously described (Rubinov and Sporns [43]). Their results are shown in Figs. 7, 8.

This "rich club" observation is closely related to the assortativity we have introduced in the theoretical part of the review. As we previously introduced, these types of networks are called the correlated networks, the NBEB method by Wu et al. [25] may have a potential to obtain interesting results.

Using graph theory, Shanahan et al. [44] showed that the pigeon telencephalon is organized in a similar way to mammals. Both are modular small-world networks with connected cores of hub nodes that includes prefrontal-like and hippocampal structures. Fig. 9 shows the transverse sections through the pigeon telencephalon. Topologically, these hub nodes are the most central area of the pigeon's brain and the most densely connected, meaning that information flow plays an essential role. With the help of the *k*-core decomposition, they found that for the pigeon connectome, complete

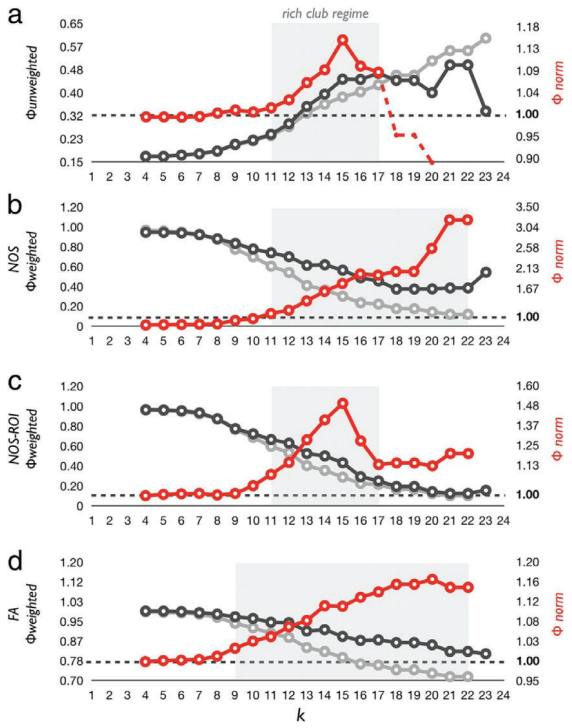


Fig. 8. Rich club's unweighted and weighted group network capabilities. *a* shows the rich club's average brain network's rich club $\Phi_{norm}(k)$ curve (i.e., reflecting all direct connections between brain regions). The figure shows the structure of the rich club behavior brain network, showing an increased standardized rich club coefficient $\Phi_{norm}(k)$ from k range 11 to 17. *b–d* show the weighted group average structure of the rich club value brain network $Phi^{w-nos}(k)$ [weighted by the number of connection streamline (*b*)], $\Phi^{w-nosROI}(k)$ [weighted with the number of connectivity streamlines corrected for ROI volume (c)], and $\Phi^{w-fa}(k)$ [weighted with the FA value of the white matter connections (*d*)]. These results show rich club coefficient values for the range k, forΦ^{w} (dark gray), Φ^{w}_{random} (lightgray)and $\Phi^{w}_{n}orm$ (red). Similar to the unweighted network, Φ^{w} is found to be larger than Φ^{w}_{random}, suggesting rich-club organization for all four variations of the structural brain network.
Source: Reproduced from Ref. [42].

erosion happened at $i = 11$ and the innermost k-core ($i = 10$) contained more than half of the nodes in the network. However, when the nodes are ranked according to the subshell members (as shown in Fig. 10), four of the five connector hubs (AI, APH, CDL, and NCL) are considered to be in the innermost sub-shell, and all five connector hubs are all located in the innermost 2 sub-shells (Fig. 11).

Based on all the network metrics used for evaluation (node degrees, betweenness centrality, k-core and sub-shell membership), the connector hubs are uniquely prominent, and may be designated as the connecting core of the pigeon forebrain. Overall, their analysis show that although there was no cortical layer and nearly 300 million years of independent evolution, the connectivity of the bird brain was identical to that of the mammalian brain.

Emerson et al. [45] established a human proteins Domains Co-occurrence Network (DCN) based on Pfam (Protein family) data. DCN can be used to study protein functions and interactions by representing protein domains and their coexistence in genes, and by mapping cancer mutations into individual protein domains. They then used the k-core decomposition technique to identify cores of DCN, which is a highly connected domain in the network. Their results show that these central regions were found to be more evolutionarily conserved than peripheral domains. Combining the somatic mutation information for ovarian, breast and prostate cancer diseases from the TCGA (The Cancer Genome Atlas) database, they projected somatic mutations to a single protein domain and used local false discovery rate to identify domains of significant mutations in each cancer type listed above. The Significantly mutation domain was found to be rich in cancer disease pathways. However, they found that the core of the DCN does not contain any significant mutation domains. The authors observed that the large coreness protein domains are highly conserved and that these domains coexist in a large number of genes with other protein domains. Mutations and DCNs provide a framework for understanding hierarchical structure in protein function from a network perspective. The results indicate that most of the protein domains in the DCN cores have lower mutation frequencies, and that the protein domains in the peripheral regions contribute more to the disease. These findings are suggested to possibly contribute to drug development in the future.

Bola et al. [46] studied the emergence of cognition from the network perspective. Previously the mainstream hypothesis states that the interaction between distributed brain regions through phase synchronization provides the basis for

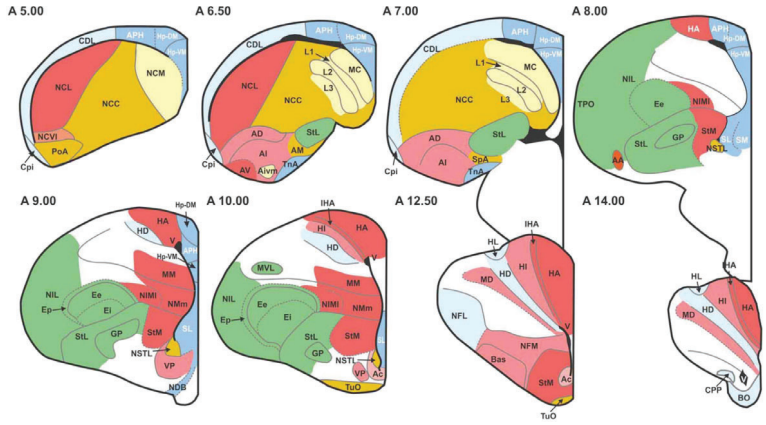


Fig. 9. Transverse sections through the pigeon telencephalon showing the locations of each of the 52 regions used in the study. Regions are colored according to their module and sub-module membership (see also Fig. 4 in Ref. [44]). Color codes: red, associative; blue, cortico-hippocampal; green, visual; brown, viscero-limbic; yellow, auditory. Regions colored white are excluded from the study. While the connections of these white regions have been explored, they have not been systematically clarified nor unequivocally confirmed. Black areas, such as the one labeled "V" at A14.00, are ventricles.
Source: Reproduced from Ref. [44].

Region	Degree	Region	Sub-shell
AI	42	**AI**	19
NCL	37	**APH**	19
AD	31	**CDL**	19
CDL	28	HD	19
APH	27	HL	19
Hp-DM	24	**NCL**	19
PoA	24	**AD**	18
SL	22	HA	18
MD	21	Hp-DM	18
NSTL	21	MD	18
HD	20	NFL	18
AA	19	MM	17
HA	19	NMm	17
NFL	18	PoA	17
StL	17	SL	17
SM	15	AA	16
MC	14	Ac	16
MM	14	Hp-VM	16
NMm	14	NSTL	16
Ac	13	SM	16
Hp-VM	13	StL	16
StM	13	StM	16
CPi	12	CPi	15
HL	12	TnA	15
TnA	12	TPO	15
TPO	12	TuO	15
		VP	15

Fig. 10. Node degree and sub-shell number (following *k*-core decomposition) for the top 50% of the nodes in rank order.
Source: Reproduced from Ref. [44].

cognitive processing. This phase-synchronized networks are transient and dynamic, built on a typical time scale of milliseconds to practice specific cognitive operations. The authors studied whether the cognitive processing changes the strength of the functional connection or, on the contrary requires a qualitatively new topological structure of the functional network. To solve this problem, they recorded a high-density electroencephalographic (EEG) while the subjects performed a visual discrimination task. They conducted Event-Related Network Analysis (ERNA), where the source-space weighted

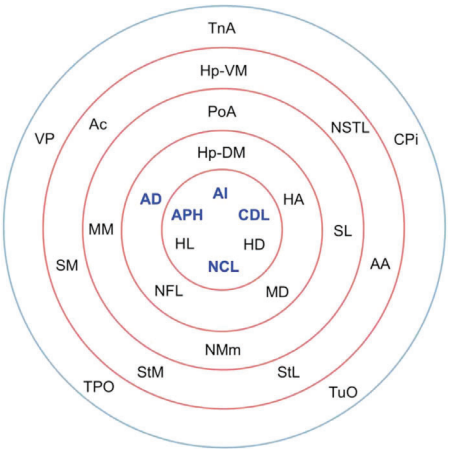


Fig. 11. Sub-shells of the innermost k-core following k-core decomposition. The innermost k-core ($i = 10$) contains almost half the nodes in the network, but its sub-shell structure reveals a finer level of organization. All five hub nodes (shown in bold) appear in the innermost two sub-shells. *Source:* Reproduced from Ref. [44].

functional network was characterized by graph measurements. ERNA reveals rapid, transient and frequency-specific recombination of network topology in cognitive processes. In particular, cognitive networks are characterized by strong clusters, low modularity, and strong interactions between hub nodes. Their findings indicate that dense and clustered connections between central nodes of different modules are cognitive "network fingerprints". This reorganization model may facilitate the global integration of information and provide the foundation for the "global workspace" that is necessary for cognition and awareness.

In 2016, Lahav et al. [47] applied an analysis using k-shell decomposition to establish a human cortex topology model, revealing the hierarchical structure of the cortical network. As we previously introduced, the k-shell decomposition is closely related to the concept of k-core. It uses the exactly same decomposition algorithm but the output k-shell is the "shell" instead of the core, i.e., a k-shell is the collection of nodes that is left when removing $(k-1)$-core from k-core. These shells, representing known cortical networks, allow a detailed picture of cortical hierarchical structure. In the characterization of cortical regions as well as hierarchies, this method is shown to have an increased precision than common approaches. The analysis was applied on a human cortical network that is constructed from Magnetic resonance imaging (MRI) as well as Delayed Sequence Intubation (DSI) information of six individuals. Such analysis enables us to portray an in-depth photo of cortical connection concentrating on various areas of inter-connected layers across the cortex. Their findings show that the human cortex is highly connected and also efficient, and unlike the Internet network that consists of no separated nodes. The cortical network has an inner-core together with outer shells of increasing connections that altogether become a giant component. All these components were more categorized right into three hierarchies in accordance with connection account, with each hierarchy showing various useful roles. Such a model could describe an efficient flow of information from one of lowest hierarchy to the highest one, with each movement allowing increased information integration. On the top, the highest hierarchy (the core) functions as an global interconnected center and more importantly, shows high correlation with consciousness associated areas, suggesting that the core might function as a system for consciousness to emerge.

Above we only show a small fraction of the many researches that apply the k-core decomposition to study biological topics. More interesting topics can be found in Ref. [45,48–53].

3.2. The applications in Ecology

Recently there has been a number of novel researches [54–56] that study ecology from the perspective of network science. We briefly survey some of them in the following.

Garcia et al. [55] recently suggested an additional k-core decomposition application in ecology. They specify 3 k-degree based upon k-core decomposition: k-degree, k-risk as well as k-radius. The first one, k-radius, evaluates the range from the node to the innermost layer of the partner guild, while k-degree is a centrality measure based upon k-shell decomposition. k-risk is a step of the susceptibility of a network to the loss of a specific species. They assessed 89 cooperative networks involving plant pollinators or seed communicators. They utilized two extinction processes. The first process is to progressively eliminate a species of a species according to the chosen index, despite which guild it belongs to, and also in the second extinction procedure, just the animal species are actively eliminated The following extinction happens when a node is found to be isolated [57]. They found that when species from the both two guilds go extinct, if people want to find out the vital species that keep the giant component, then k-risk is the most effective ranking

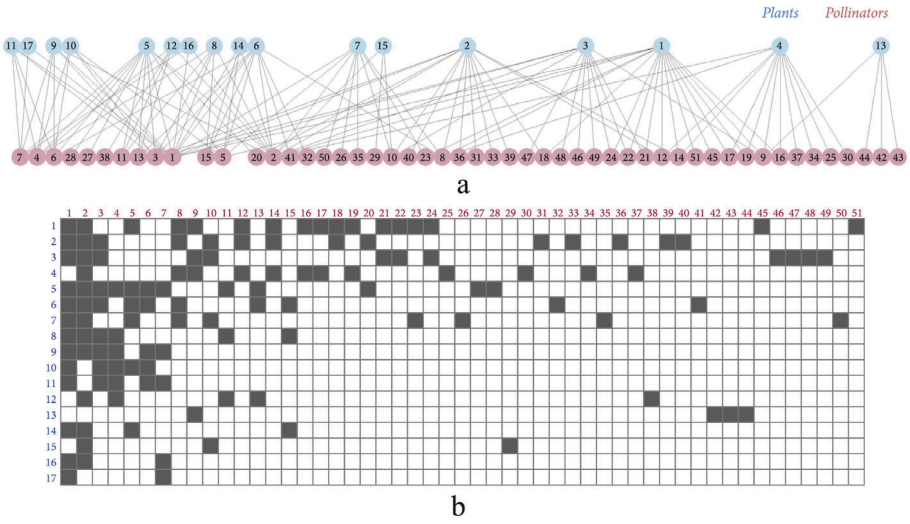


Fig. 12. Mutualistic community in Tenerife, Canary Islands (Spain), with 68 species and 129 links [29]. In mutualism, species fall into two disjoint guilds, such as plants and pollinators or plants and seed dispersers. Ties amongst species of the same guild are forbidden. (a) Bipartite plot of this community. (b) Interaction matrix.
Source: reproduced from Ref. [56].

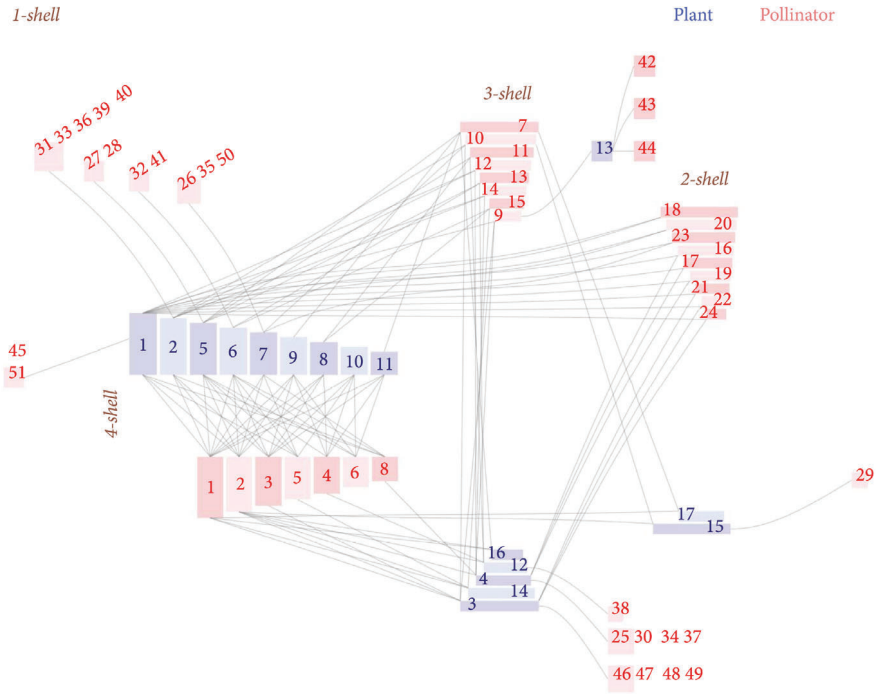


Fig. 13. Gene-pathway association network.
Source: reproduced from Ref. [56].

index. When only the animal guild species go extinct and the consequent cascading extinctions of species happen in the secondary guild, the most reliable ranking index for the essential species that preserve the giant component is *k*-degree. Nevertheless, the MusRank index is more efficient when the goal is to recognize essential species to maintain the highest degree of species richness in the second class.

Similarly, Garcia et al. [56] proposed a structural approach to visualize bipartite biological networks, as shown in Fig. 12, where they used the *k*-core decomposition method as a tool to obtain the nodes that share connectivity properties. An example is shown in Fig. 13.

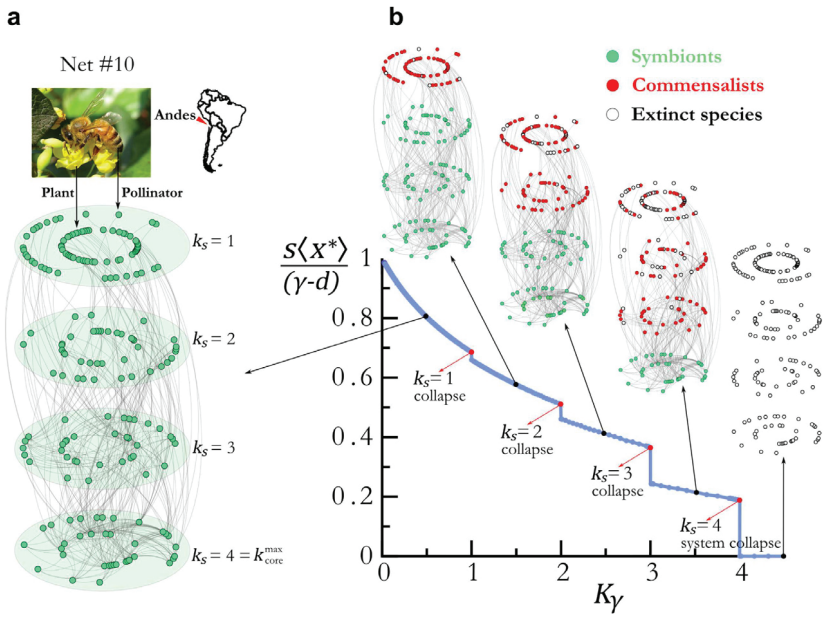


Fig. 14. Collapse of a plant–pollinator mutualistic network and the tipping line of mutualistic ecosystem. (a), a bipartite mutualistic network of a plant–pollinator ecosystem located in the Chilean Andes. The network is formed by 4 pairs of concentric rings. Each pair of rings contains species with the same k-shell ks, ranging from 1 to 4. The innermost core is at $k_{core}^{max} = 4$. Species in the inner rings of each k-shell represent the plants, and species in the outer rings represent the pollinators. (b) Fixed point average density (properly rescaled) $\langle x^* \rangle = N^{-1} \sum_i x_i^*$ as a function of the threshold K_γ for the mutualistic network in a, obtained by numerical integration.
Source: Reproduced from Ref. [59].

In 2018, Filho et al. [58] constructed a metabolic network of 17 plants covering unicellular organisms, as well as dicotyledonous plants. The network was constructed based on the substrate-product model and then they perform a k-core percolation on the network. The metabolites distribution across the percolation layer suggests a correlation exists between the metabolic integration layer and the topology of the metabolic network. Their results show that the metabolites in the maximum k-core which means they concentrated in the internal network only contain the molecules of the primary basal metabolism. In addition, they found a high proportion of a common set of metabolites in 17 plants, centered on the inner-core layer, while the metabolites considered to be participants in plant secondary metabolism are concentrated in the outermost layer of the network. This data indicates that metabolites in the central layer form a basic molecular module in which the entire plant is metabolically anchored. Elements from the core of the center are involved in almost all plant metabolic reactions, indicating that the plant metabolic network has a very centralized topology.

Dynamic system collapses into an unrecoverable state in ecosystems, human societies, financial systems, and network infrastructure. Although these events are widespread and widely influential, we are still largely in the search for causes of collapse and instability, and theoretical studies have so far failed to quantitatively determine the structural characteristics of networks formed by interacting species. Recently in 2018, Morone et al. [59] derive the theoretical conditions for the stability of the symbiotic ecosystem as constraints on the dynamic interaction strength between species and network topology invariants: k-core. They showed their important results that the k-core as a topological invariant of the network, is the condition for the stability of a mutualistic ecosystem as a constraint on the strength of the dynamical interactions between species. Their solution predicts that when the largest k-core species in the network go extinct due to weakening of interaction strength, the system will reach its critical point of collapse. Fig. 14 shows the process of the collapse of a plant–pollinator mutualistic network and the tipping line of mutualistic ecosystem. As a key variable involved in the collapse phenomenon, the core of the monitoring network may be a powerful method for predicting catastrophic events in the vast environment from ecological and biological networks to finance.

3.3. The applications in Computer Sciences

As a widely used computational algorithm, k-core is naturally to be applied to the Computer Sciences. In the numerous researches [60–71], we show some examples that contribute to the existing knowledge.

In 2004, Gaertler et al. [72] found the k-core useful in examining the Autonomous System (AS) Graph. They apply the method to the field of Internet graph evaluation at the AS level. When dealing with large amounts of data, the usual technique is to filter out irrelevant information. While the most common are the degree of nodes, such as Refs. [73–75], they used core concepts in their study. The core can be used to filter peripheral ASes. For example, an appropriate k value

can be selected based on the number of remaining ASes, in their case they chose to use the maximum k that keeps at least 200 ASes. After the cleaning process, they devised a new algorithm that takes temporal characteristics of the network into account, and found that the numerous transients that affect the Internet seem to be to have a very different effect on their activities, ranging from no (earthworm attack, misconfiguration, etc.) to significant one (DDoS for DNS hosting servers), offering a method to distinguish them.

Alvarez et al. [76] also applied the k-core decomposition to study large scale Internet graphs at the AS level. This method enables the characterization of increasingly central cores of networks, easily revealing ordered and structural features. This structure suggests that the Internet is organized in a specified hierarchy of linked subgraphs of raising centrality with self-similar properties. The research study of the k-core subgraphs reveals the primary hierarchical layers of the network and also permits their analytical characterization.

In the same year, they also used k-core decomposition to study large-scale Internet diagrams at the autonomous system level [76]. This approach allows for a step-by-step characterization of the central core of the network, making it easy to reveal hierarchies and structural attributes. The Internet map shows the remarkable properties of all k-cores with a single connected component with constant statistical characteristics (degree distribution, correlation spectrum, etc.). This feature shows that the Internet is organized in a defined hierarchy of connection subgraphs, which have the centrality of self-similar properties. Their results show that k-core decomposition also provides an interesting tool to track the time evolution of Internet maps and test the stability and reliability of different mapping strategies. k-core decomposition allows the network to be pruned step-by-step and the subgraphs that increased centrality are identified. The study of the obtained subgraphs reveals the main layers of the network and allows them to be statistically characterized. Worthwhile to mention, they have also observed statistical self-similarity in the topological properties of the Internet in ASes, which have increasingly important cores.

Soon in 2006, Alvarez et al. [77] proposed a universal visualization tool for large-scale networks. Using k-core decomposition and the natural hierarchy generated from it, the algorithm-generated layout has a simple 2D representation and encodes a large amount of information. One can easily read the basic features of the graph (degrees, levels, positions of the highest nodes, etc.) and more in-depth features, such as the relationship between nodes and the hierarchical position of the neighbors. Their results show that the rationalization of the corresponding graphs provides a clear understanding of the structure of many real-world and computer-generated networks.

Carmi et al. [13] explored Internet maps (at the AS level) by introducing and using k-shell decomposition methods and methods of seepage theory and fractal geometry to find models of the Internet structure. They show that their decomposition method is robust and provides insight into the infrastructure of the Internet and its functional consequences. Their approach to decomposing the network is generic and useful when studying other complex networks.

Zhang et al. [78] applied the k-core decomposition approach on the real Internet networks and discovered that the size of the k-core with a larger k was basically unchanged, as well as the observation that maximum coreness stays relatively unchanged after 2003. They randomized the data by link-exchanging operations and additionally, they methodically compared the framework of the real Internet and its random versions and found that the genuine Internet was a lot more loosely connected, in accordance with the empirical outcomes reported in Ref. [79]. They also found that the full Internet and the cores are much more disassortative than their randomized versions. The Internet is found to be steady in terms of the maximum degree, contrasted to the predictions of the majority of previous models. One of the most exact Internet model (as established by several topological criteria) is the so-called Positive Feedback Preference (PFP) model. In this model, the node's capability to obtain new links increases with the feedback loop to the degree of the node, so the optimum k increases really rapidly as the size of the network increases (faster than the BA model). As shown in Fig. 15, the maximum degree of the Internet is additionally reasonably stable relative to time (as shown in Fig. 16), suggesting that there are some unknown evolving mechanisms, rather than PFP models. Actually, a lot of previous models embedded with the preferential attachment mechanism do not recreate maximum degree stability. Even though the aging effect is a possible candidate to reproduce such stability of maximum degree [80], there is no clear evidence of the existence of an aging mechanism in the real Internet. Individual degrees of website traffic capacity limitations can result in limits for individual links. Another prospect that might influence the statistical properties is the interaction between existing nodes: new links between existing nodes can be produced, while some existing links can rewire or disappear.

Li et al. [81] proposed a social-aware k-core selection algorithm that can be used in the Pocket Switched Network. The social relationship of the network indicates that the social location of the mobile node can help find key nodes in the Pocket Switched Network. The Sk-core selection algorithm is designed to take advantage of the social capabilities of nodes to improve the performance of data communications. In addition, with social behavioral information, these key nodes are better suited to represent and improve the performance of the entire network. Nodes are mobile devices, such as mobile phones, iPads, and laptops, all of which are associated with people with certain information tags. The links are the communications between the nodes within the transmission range. This network has two kinds of interpretation: it is both a physical network and a social network. This network belongs to Pocket Switched Network [82], also known as opportunistic mobile ad hoc network [83]. In fact, human social behavior determines and affects the data exchange of the Pocket Switched Network formed on mobile devices. They demonstrate the effectiveness of the Sk-core selection algorithm in selecting key nodes in the Pocket Switched Network. Their article summarizes three important contributions. (1) Sk-core introduces a new recognition system that uses physical topologies and locations in social networks to identify mobile nodes. (2) Sk-core introduces a new concept, social status, for assigning weighting factors to k-degree calculations.

Time	N	E	C	$\langle d \rangle$	r	k^*	$k_{\max}$	N_n
2001–12	12666	25672	0.296	3.62	−0.199	2609	15	18
2002–6	13631	27749	0.292	3.65	−0.190	2692	15	24
2002–12	14625	29057	0.257	3.70	−0.193	2591	14	40
2003–6	15740	32263	0.264	3.71	−0.199	2507	17	40
2003–12	16691	35604	0.257	3.72	−0.199	2436	21	46
2004–6	17861	39637	0.266	3.72	−0.194	2437	24	47
2004–12	19085	42175	0.273	3.74	−0.200	2424	24	54
2005–6	20349	44016	0.275	3.75	−0.202	2462	23	37
2005–12	21588	45917	0.260	3.79	−0.196	2456	23	41
2006–6	22960	48545	0.242	3.82	−0.196	2460	23	78
2006–12	24403	52826	0.242	3.82	−0.196	2467	25	76

Fig. 15. The basic topological properties of the Internet at AS level for about five years with sampling interval of six months. Here, N and E are the total number of nodes and links, C denotes the average clustering coefficient, $\langle d \rangle$ is the average distance, r is the assortative coefficient that quantifying the degree–degree correlation, and k^* denotes the maximal degree among all nodes. Note that another symbol, k_{max} is used to denote the maximal core index. N_n denotes the size of the cores, that is to say, the number of nodes in the k_{max}-core.
Source: Reproduced from Ref. [78].

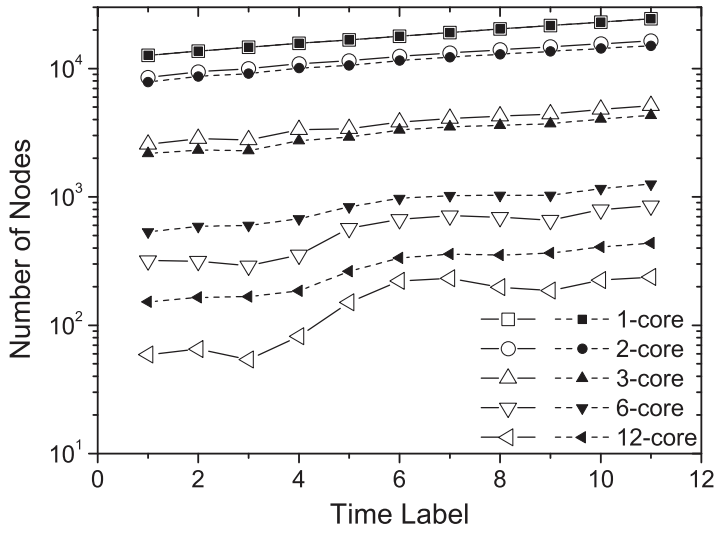


Fig. 16. The sizes of k-cores versus time. The time labels 1 to 11 correspond to December 2001 to December 2006 with six month intervals. The large open symbols denote the empirical results of the real Internet, and the filled small symbols denote the numerical results of randomized networks.
Source: Reproduced from Ref. [78].

(3) They performed numerical simulations and evaluated the performance of the *Sk*-core selection algorithm in static, slow motion and fast motion.

Shin et al. [84] explored the universal model associated with k-core and appeared in several different areas of the graphs. Their findings are as follows: (1) Mirror Pattern: The coreness of the nodes is closely related to their degree. (2) Core Triangle Pattern: The degeneracy of the graph (i.e., the maximum k such that k-core exists in the graph) follows a 3-to-1 power law with respect to the number of triangles. (3) Structured Core Pattern: Degeneracy-cores are not cliques, but have non-trivial structures such as core–periphery and communities.

In social network applications, large amounts of data are structured in the form of graphs, and graphical data analysis requires a lot of computation time. In 2014, an in-memory computing framework, Spark, was introduced for big data analysis. By reloading data in memory to solve long runtime problems, Spark can complete tasks in less time than Hadoop. In November 2014, Spark broke the world record for sorting data in the benchmarking competition, and the previous record was done by Hadoop. In addition, GraphX is a Spark API (application programming interface) that provides a

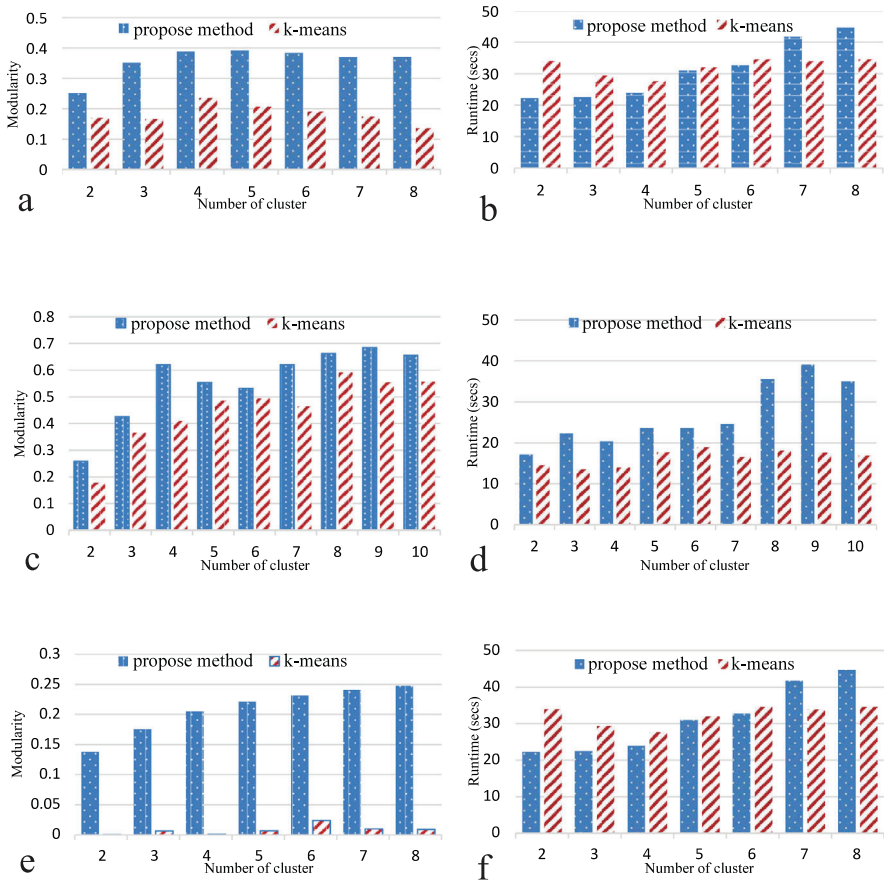


Fig. 17. The experimental results by Cheng et al. [85]. (a) and (b) compare the Modularity and runtime of their method and *k*-means, on dolphon social network. (c) and (d) compare the Modularity and runtime of their method and *k*-means on Facebook social network. (e) and (f) compare the Modularity and runtime of their method and *k*-means on Gnutella social network.
Source: Reproduced from Ref. [85].

graphical interface that makes graphical data analysis simple and effective. Cheng et al. [85] devised a new algorithm in GraphX on Spark. Their study proposes an improved *k*-means clustering method by integrating *k*-core decomposition. They combined *k*-core decomposition with graph-based *k*-Means, where *k*-core is used for searching the center and *k*-Means is used for finding clusters. This algorithm can further improve performance and results. The experimental results are shown in Fig. 17.

Due to the linear time complexity of *k*-core decomposition, the method can be extended to large real networks as long as the input graph is suitable for the main memory. For graph that are larger for the main memory, an external memory based approach or a distributed solution based on an iterative MapReduce platform has long been proposed. However, due to the high cost of disk Input/Output, both external storage solutions and iterative MapReduce-based solutions are slow. In addition, Mandal et al. [86] proposed Spark-*k*-core, a distributed *k*-core decomposition algorithm that runs on the Spark cluster computing platform. They use a think-like-a-node paradigm, and the proposed method uses a messaging paradigm to solve the *k*-nuclear decomposition, which greatly reduces the I/O cost. Experiments on 15 large real-life networks show that their approach is much faster than existing *k*-core decomposition solutions.

We assume that if the method proposed by Mandal is integrated into the algorithm proposed by Cheng et al. [85] that combines the *k*-core and *k*-means methods, it will have a great potential to produce a very economic and efficient algorithm, that can finish the clustering analysis and core-decomposition in a very short runtime even for millions of data. This method is particularly useful, especially in today's fast technological advancement, where commercial companies have to process massive amounts of data in real time online.

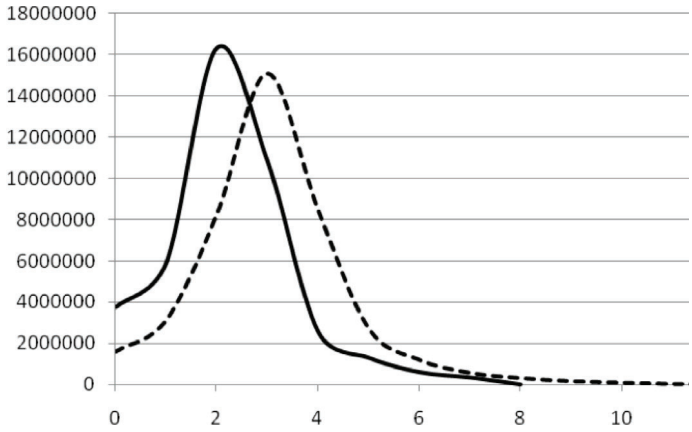


Fig. 18. Users counts per *k*-shell.
Source: reproduced from Ref. [93].

3.4. The applications in Social Networks

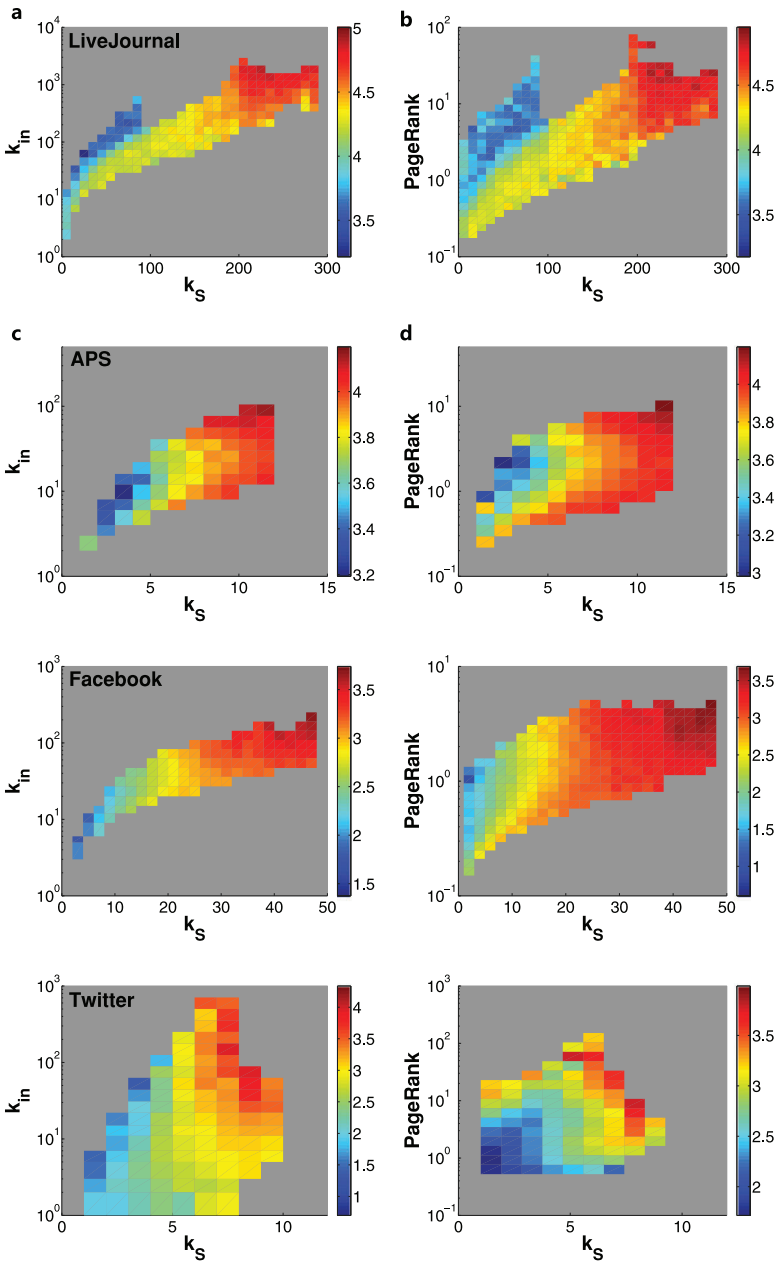
The *k*-core decomposition is widely accepted to reveal the network structure, as a result, many researchers have proposed a number of methods [87–91] that can identify the key nodes in the network or to measure the influence of users in online social networks.

Kitsak et al. in their paper published in 2010 [92] showed that, contrary to the popular belief, there is a paradoxical situation where the best spreaders do not correspond to the most connected or the most central nodes. Instead, one of their findings is that the most effective spreader is a spreader located within the core of the network, as determined by *k*-shell decomposition analysis. In addition, they found that an infection persists in the high *k*-shell of the network, even if the recovered individual does not produce immunity. Their analysis provides a reasonable approach to optimal design for efficient dissemination strategy.

Brown et al. [93] modified *k*-shell decomposition to assign a logarithmic *k*-shell value to the users, resulting in a user metric that is well distributed in a well-fitted bell curve, as shown in Fig. 18. In addition, they identified and removed peering relationships in the network to further differentiate users. They used two Twitter data sets, one of them was collected by KAIST in 2009 [94]. It includes 41.7 million user profiles (user data) and 1.47 billion social relationships (network data). Another data set was collected by Lehigh University, which included more than 80 million actual tweets since October 2009, representing more than 7 million users, accounting for around 17% of the total Twitter community. They modified the original algorithm by applying a logarithmic mapping, where each *k*-shell roughly corresponds to the logarithm of the node degree. While the original *k*-shell decomposition algorithm classifies nodes with *k* or fewer degree into *k*-shell, our modified algorithm will place the nodes with 2^{k-1} or fewer connections in the *k*-shell level *k*, effectively consolidating the higher *k*-shell level. This modified algorithm produces smaller but more meaningful *k*-shell values. They found that the user's position in the log *k*-shell level produces a more useful distribution. They also found that the modified algorithm has a time complexity of $O(log_2 n)$ times for a given network, and is faster than the original algorithm. In this paper, the modified *k*-shell decomposition algorithm was used to measure the impact of users on the Twitter network. These measures are verified against Twitter usage data. The application of this algorithm can define an effective ranking of influence for the user, so it can be used as a baseline measure of the influence of the Twitter network.

Pei et al. [95] looked for influential spreaders by tracking the actual spreading dynamics in a large number of networks. They found that the widely used Degree and PageRank could not rank the impact of users. They continued to discover that the best communicators have been on *k*-core on different social platforms, such as Twitter, Facebook, Livejournal and the American Physical Society. In addition, they found that when the complete global network structure is not available, they find that the sum of the nearest neighbors is a better local approximation of the user's influence. Their results are shown in Fig. 19.

In 2015, through a lot of numerical simulations, Liu et al. [96] observed that not all network nodes in high k-shells are very influential: the core nodes in some networks are the most influential, they call it the "true core", but in other cases although the nodes' coreness are high, even in the innermost cores, are not good spreaders, they call them "core-like" groups. By analyzing the *k*-core structure of the network, they found that the "true core" of the network has connections with the different *k*-shell layer of the network, and the "core-like" group nodes are connected locally within the group. For nodes in core-like groups, *k*-shell indexes do not reflect their positional importance in the network. They further introduced a metric based on link diversity of the shells to effectively distinguish between "true-core" and "core-like" groups and identify "core-like" groups throughout the network. Their findings help to better understand the structural characteristics of real networks and influential nodes.


a
LiveJournal
b
c
APS
d
Facebook
Twitter
k_{in}
PageRank
k_S