Applied Network Science

RESEARCH Open Access

# A bridge between influence models and control methods

Abida Sadaf[1*], Luke Mathieson[1], Piotr Bródka[2] and Katarzyna Musial[1]

*Correspondence: abida.sadaf@uts.edu.au

[1] Complex Adaptive Systems Lab, University of Technology Sydney, Sydney, Australia
[2] Department of Artificial Intelligence, Wroclaw University of Science and Technology, Wroclaw, Poland

**Abstract**

Understanding how influence is seeded and spreads through social networks is an increasingly important study area. While there are many methods to identify seed nodes that are used to initialize a spread of influence, the idea of using methods for selecting driver nodes from the control field in the context of seed selection has not been explored yet. In this work, we present the first study of using control approaches as seed selection methods. We employ a Minimum Dominating Set to develop a candidate set of driver nodes. We propose methods based upon driver nodes (i.e. Driver-Random, Driver-Degree, Driver-Closeness, Driver-Betweenness, Driver-Degree-Closeness-Betweenness, Driver-Kempe, Driver-Ranked) for selecting seeds from this set. These methods make use of centrality measures to rank the driver nodes in terms of their potential as seed nodes. We compare proposed methods to existing approaches using the Linear Threshold model on both real and synthetic networks. Our experiment results show that the proposed methods consistently outperform the benchmarks. We conclude that using driver nodes as seeds in the influence spread results in faster and thus more effective spread than when applying traditional methods.

**Keywords:** Complex networks, Influence spread, Control, Driver nodes, Seed selection methods

## Introduction

Since the beginning of social media, our online activities transformed how we interact with others, which has changed our social networks. Social media allow us to communicate and interact with others through sending direct messages, sharing opinions and information, as well as commenting on others' content. Interactions over social media platforms may play an effective role in the quick and worldwide proliferation of news and can shape the opinions of users. Although the social media proved to be an effective way to influence the public opinion, we know that not all users play the same role in this process. An example of that is 'influencers' who are seen as key players in the propagation of the information quickly and effectively (Zareie and Sakellariou 2021). Spread of influence, in particular, has gained a lot of attention in recent years as various research groups and commercial companies try to understand how people's opinions and decisions can be influenced and potentially changed and to what extent we are receptive to others' opinions. How does the influence spread in networks? is a question for which many researchers from variety of fields try to find answer. This includes physics (Buldyrev et al. 2010), ecology (Fath et al. 2007), biology (You et al. 2003) and network science (Zhang et al. 2016).

Many studies focus on how to quantify the influence of nodes in a complex network (Guo et al. 2016; Lu and Dong 2019) with the hope that if the most influential nodes are chosen to propagate a given phenomenon, then the spread of this phenomenon will be optimal. One of the avenues to explore, in the search for more effective ways to assess the influence potential of a given node, is to look into the direction of control over complex networks. We know that any network can be fully controlled if we control every single node, but this is a very costly approach which in most cases is not feasible (Nacher and Akutsu 2012). Thus, one of the goals of research in the space of network control is to find a minimum number of driver nodes that enable us to control a given network.

There are a few notable works in this regard such as very recently, a recommender system to identify more influential nodes to increase the efficiency of spreading process is proposed by (Vitoropoulou et al. 2021) is one such example. Previously, in another related work, Koutsopoulos et.al, proposed a low-complexity heuristic algorithm to build a recommender system to achieve efficient coverage of nodes (Koutsopoulos and Halkidi 2018). We used the control approaches in influence models in the hope to achieve the efficient coverage of nodes with a smaller set of influential seed nodes.

There are conceptual similarities between driver nodes in the network control space and seed nodes in the spreading processes, and the goal of this study is to explore the possibility of using driver nodes as seed nodes and proposing and developing new seed selection strategies for spread of influence inspired by driver node concept. Control can be seen as a "stronger" version of influence (Watts and Dodds 2007). Driver nodes Driver nodes, are nodes which can be directly controlled by external inputs, and these nodes play a crucial role in controlling all the other network nodes (Qin et al. 2023). While seed nodes are the nodes that can be identified by employing different seed selection strategies traditionally relying on the node degree based seed selection. Which means the nodes that have a higher node degree can be the seed nodes and these nodes work as the basis of the first set of seed nodes which are then used to propagate the influence spread process which has been explained at length in the Section 1.1.2. The hypothesis is that the influence can spread effectively (affect a bigger number of nodes) through driver nodes than when using other, traditional seed selection strategies. Thus, we use driver nodes and their rankings with the aim of getting bigger influence coverage. It means that we focus on maximising the number of nodes influenced in smaller number of iterations by utilising minimum seed size.

*The aim of this paper mainly revolves around utilising the concepts from the field of network control and apply those to improve the spread of influence in the network by using seed selection methods based upon driver nodes.*

To address this goal, the following research questions are explored and answered in this study:

1 Research Question 1 (RQ1): How can the concepts from network control be used in the spread of influence field?

2 Research Question 2 (RQ2): How effective can be the implementation of concepts originating from the control field in the influence field?

Based upon the research questions, main objective of the paper is defined as: "To develop and validate new seed selection methods that are utilised concepts from network control field."

To achieve this research objective, we begin by proposing new methods for seed selection that utilize the concept of driver nodes. The methods were developed by identifying global-level driver nodes. We define the methodology in Sect.  in detail which explains the experimental setup to test the proposed seed selection methods with globally identified driver nodes, hence helps in achieving the research objective. In short there are three main steps in the methodology which are defined as:-

- Seed selection from traditional methods, for example, highest node degree.
- Seed selection from driver-based methods, where driver nodes are the basis for seed nodes set.
- Influence spread process by using Linear Threshold Model.
- A comparison of efficiency and effectiveness of traditional and driver-based seed selection methods in spreading influence in synthetic and social networks.

The paper's main contribution is the development and validation of newly manufactured seed selection methods that helped bring together control and influence fields. The efficient and effective seed selection method(s) have been identified from a set of various methods.

In this paper, Sect.  describes the related work including influence and control in complex networks. Section  describes the detailed experiment setup of the experiments being conducted to answer the research questions. Section  describes the results and their comprehensive analysis. Lastly, the conclusion and future work are discussed in Sect. .

## Background

This section provides a brief overview of influence in complex networks, influence models, seed selection strategies, and the effectiveness of influence spreaders. Then, we explore and present the main concepts behind network control, such as methods to identify and rank driver nodes.

### Influence in complex networks

Spreading models are widely used to simulate the propagation of information, influence, opinion, content, virus, etc., over a complex network to see how many nodes can be affected and how fast they can be affected when different approaches are used (Kempe et al. 2003). Our research focuses on developing new strategies for choosing a set of seed nodes as source spreaders (a.k.a. seeds).

There are two most commonly used influence spreading models, namely the Linear Threshold Model (LTM) and Independent Cascade Model (ICM) (Kempe et al. 2003). For this research, we use LTM for influence spread in synthetic as well as real social networks. The same model is used across all the experiments to enable comparison of results across the whole study.

Regardless of the spreading model, at the beginning we need to select at least one node as a seed node which starts the spreading process. We can do it at Random, like in the case of epidemic models, or we can use some heuristics. Some of the most commonly used methods, where top ranked influential nodes are selected, are Degree Centrality, Betweenness Centrality, Closeness Centrality, PageRank, LeaderRank, ClusterRank, K-Shell, Hill-climbing, HITS, ARL and Social Position (Banerjee et al. 2020; Musiał et al. 2009; Erlandsson et al. 2016) (see Sect. ).

### *Spreading models*

There are dozens of spreading models designing to model a specific spreading process, for example, epidemiological models like SI, SIS or SIR (Pastor-Satorras and Vespignani 2001), awareness spreading models like UAU (Zang 2018), UAF (Scatà et al. 2016), opinion formation models like Voter model (Holley and Liggett 1975) or innovation spreading models like Bass model (Bass 1969). For full description of those we refer the reader to some of the recent survey papers in this area e.g. (Banerjee et al. 2020; Bródka et al. 2020). However, as mentioned above, in this paper we focus on influence spread where the two most commonly used influence spreading models, are *the Linear Threshold Model (LTM)* and *the Independent Cascade Model (ICM)* (Kempe et al. 2003). In both models, we can distinguish between active nodes, which spread the influence, and inactive ones that do not. As the number of iterations or cascading spread cycles increases, we observe how the spread progresses and how many nodes change their status and become active.

Independent Cascade Model (ICM) The main idea behind ICM is a common phenomenon defined in the field of behavioural economics and network theory, which occurs when a number of people make the same decision in a sequential order (Kempe et al. 2003). An information cascade model works in two steps: (i) first step is that an individual must encounter a scenario with a decision (yes or no) only then a cascade can begin; (ii) second step includes outside factors, that can influence this decision (DuanW 2009). In ICM, an active node $u$ attempts to influence all of its inactive neighbours but the success of the node $u$ in activating its inactive neighbour $v$ depends on the activation probability (a.k.a. propagation probability) of the edge from $u$ to $v$ (each edge can have its own value and the value $u \rightarrow v$ can be different from $v \rightarrow u$). Regardless of its success, the same node will never get another chance to activate the same inactive neighbour. The process ends when no further node gets activated.

Linear Threshold Model (LTM) In LTM the idea is that a node becomes active if a sufficient part of its neighbourhood is active. Each node $u$ has a threshold $t \in [0, 1]$. The threshold represents the fraction of neighbours of $u$ that must be active in order for $u$ to become active (e.g., how many of our friends have to switch to iPhone to push us to switching as well). At the beginning of the process a small percentage of nodes (seeds) is set as active in order to start the process. In the next steps a node becomes active if the fraction of its active neighbours is greater than its threshold (D'Angelo et al. 2016) and the whole process stops when no node is activated in the current step. The linear threshold model (LTM) postulates that the thresholds are constrained by a linear relation to each other and therefore are completely defined by the first threshold $t_0$ and the linear increase $\delta$ as the sequence progresses (Kempe et al. 2003):

$$t_i + 1 = t_i + \delta.i. \tag{1}$$

### *Seed selection strategies*

As mentioned at the beginning of this subsection, to initiate the process we need to select at least one seed node as a seed node which will start the spreading process. We can do it at random, like in case of epidemic models, or we can use some heuristic to select the most optimal seed set which meets our needs, e.g. the total number of activated people will be the highest possible (advertisement campaign) or the total number of activated people will reach some threshold within some period (presidential campaign). Many different seed selection strategies have been developed to address different challenges, constraints and requirements. A brief description of most often used methods is included below.

- Random Seed Selection (R): In random seeds are selected Randomly from the node set of each network. Random seed selection is the baseline method to be used in comparing other seed selection methods.
- Degree Seed Selection (D): It starts by ranking the nodes according to degree centrality, and selecting a number of nodes with the highest values of degree measure (Lü et al. 2016).
- Closeness Centrality Seed Selection (C): It dictates that a top percentage of number of nodes should be selected as seeds based upon their higher closeness centrality values (Lü et al. 2016).
- Betweenness Centrality Seed Selection (B): In this method a top percentage of number of nodes is selected as seeds based upon their higher betweenness centrality values (Lü et al. 2016).
- Kempe Seed Selection (K): It is a generalisation of hill-climbing algorithm where the seed set is constructed in the following way. For each node in the network, we run the spreading process (or predefined number of steps of the process) and evaluate each node potential to activate as many other nodes as possible. We add the best node to the seed set. For the following nodes we do the same, but we evaluate each node potential in combination with all the nodes already in the seed set i.e. for the second node we check every combination of the first node in the seed set with all remaining nodes to find the best couple; for the third node we check every combination of the first two nodes in the seed set with all remaining nodes to find the best trio, etc. We continue adding nodes until we consume our seeding budget, i.e. reach the predefined size of the seed set. This approach on average produce the solution which is $(1 - \frac{1}{e})$ of maximum solution and outperforms centrality based methods like D, C or B. The disadvantage is that since we need to run the spreading process for $Nk$ times (where $k$ is the size of the seed set) it is very time-consuming, costly in terms of resources and hardly applicable for any real world solution (Kempe et al. 2003).

- PageRank: It selects a top percentage of number of nodes as seeds based upon their higher PageRank.
- LeaderRank: In this method, a top percentage of number of nodes is selected as seeds based upon their higher LeaderRank values (Lü et al. 2011).
- ClusterRank: In this method a top percentage of number of nodes is selected as seeds based upon their higher ClusterRank values (Chen et al. 2013).
- K-Shell decomposition: This method starts by selecting a top percentage of number of nodes as seeds based upon their higher K-Shell values (Wei et al. 2015; Liu et al. 2015).
- TwitterRank: In TwitterRank, a top percentage of number of nodes is selected as seeds based upon their higher TwitterRank values (Weng et al. 2010). TwitterRank is an extension of PageRank algorithm, designed to measure the importance of Twitter users taking into account similarity between users and the links between them.
- ShaPley value-based Influential Nodes (SPIN) algorithm: In ShaPley a top percentage of number of nodes is selected as seeds based upon their higher ShaPley value (Narayanam and Narahari 2010).
- Optimal Influencers: In this method, optimal seeds are identified using optimal percolation, i.e. by evaluating the size of the giant connected component after the removal of the seed nodes (Morone and Makse 2015).
- ARL: In this approach, authors use Association Rule Learning (ARL). Thanks to the use of association rules and the simple assumption that people who often start a discussion, in which many other people then take part, are important for a given community, authors developed a new ARL method. It can find key people on “raw” data without the need to project users interaction towards objects (posts and comments) to the social network of interactions between users, which we need to use “traditional” methods to finding key users such as node rank or PageRank. The evaluation showed that there is no statistically significant difference between the results achieved by ARL and PageRank, and by omitting the expensive network projection process, ARL is on average 36 times faster than the node degree and 70 times faster than PageRank (research was conducted on 108 different datasets coming from public Facebook pages) (Erlandsson et al. 2016, 2017).

Out of these methods, the traditional seed selection methods such as D, R, B, C and K are most frequently used methods. Their usefulness is already established (Kempe et al. 2003). So, we mean to utilise these as our basis to bring together the control methods and influence models.

### Control in complex networks

Control in complex networks is primarily based upon the structural controllability theory. Structural Controllability states that there is a set of driver nodes to which if external inputs or control signals are injected they can control a complex directed network. This framework allows identifying the driver nodes set in any given network. Over the years, many researchers have generated some notable research in this area has recently attracted a lot of attention (Liu and Barabási 2016; Zañudo et al. 2017; Guo et al. 2018; Zhang et al. 2019). Any network can be fully controlled if we control every single node but, as we mentioned in the sec., this is a very costly and often not feasible approach. Thus, the control method of a complex network is defined by determining the minimum number of driver nodes that are required to control the whole system. Previously Maximum Matching Algorithm (Zhou and Ou-Yang 2003), Minimum Dominating Set (MDS) (Nacher and Akutsu 2012), Control Profiles (Ruths and Ruths 2014), and Preferential Matching Algorithm (Zhang et al. 2019) were proposed to identify driver nodes. Once the driver nodes are identified, they can be ranked to determine which nodes are more critical from the perspective of network control. This can be done using such approaches as: control centrality (Liu et al. 2012), control range (Wang et al. 2012), control capacity (Jia and Barabási 2013) and control contribution (Zhang et al. 2019). We use centrality based ranking methods to prioritise more influential driver nodes in our experiments. It has been shown in relevant studies that nodes with higher value of centrality measures are more influential and can be used to influence the network (Chen et al. 2012).

### *Driver nodes selection methods*

Below we name and briefly describe methods that has been used to identify driver nodes in relevant research. **Maximum Matching Algorithm:** This algorithm treats a network as a bipartite graph (Harary 1972). For a directed network $G$, where $V(G)$ is the node set and $E(G)$ is the edge set, with $N = |V|$ and $L = |E|$. A subset of edges in $G$ is called a matching $M$ if no two edges in $M$ have a node in common. A node $v_i$ is matched by $M$ if there is an edge of $M$ pointing to $v_i$, otherwise $v_i$ is unmatched. A path $P$ is said to be $M - alternating$ if the edges of $P$ are alternately in and not in $M$. An $M$-alternating path $P$ that starts and ends at the unmatched nodes is called an $M$ augmenting path. A matching with the maximum number of nodes is called a maximum matching $M^*$. A matching $M$ is called a perfect matching if all of the nodes of $G$ are matched by $M$ and number of unmatched nodes are then called the driver nodes (Hopcroft and Karp 1973; Zhou and Ou-Yang 2003).

*Minimum dominating set (MDS)* Another development is the optimisation procedure for undirected networks which determines the minimum dominating set of nodes which are required to control the network (Nacher and Akutsu 2012). MDS is the smallest subset of nodes such that every node of a network either belongs to this subset or is adjacent to at least one node in this set. The central idea behind Minimum Dominating Set (MDS) is that each node can control all of its neighbours simultaneously, but this signal cannot propagate any further. In this method the driver nodes are identified by the minimal set such that every node is separated from another one by at most one interaction (Nacher and Akutsu 2012, 2013). MDS tells us that each driver node can control its associated nodes independently. MDS further states that each non-driver node is controllable if it is at least adjacent to one driver node. It has been used to identify control variables in protein interaction networks (Wuchty 2014) and characterise how disease genes perturb the human regulatory network (Wang et al. 2015). A dominating set of a graph $G$ is a subset $D$ of the vertices of $G$ such that every vertex $v$ of $G$ is either in the set $D$ or $v$ has at least one neighbour that is in $D$. A minimum dominating set (MDS) is the smallest possible dominating set.

*Control profiles* (Ruths and Ruths 2014) proposed an idea of building control profiles of complex networks. Those profiles can be calculated by the minimum number of independent controls ($N_c$) required for full control of a complex network which is the sum of the number of source nodes $N_s$, external dilation points $N_e$, and internal dilation points $N_i$ (Ruths and Ruths 2014). The set of nodes can also be identified by maximum matching algorithm, as explained earlier in this section. Maximum matching algorithm provides us with accurate results but with expensive running time on large networks (Ruths and Ruths 2014). By counting source and sink nodes in linear time, we obtain a relatively good lower bound on the number of controls. In terms of time complexity, this approach is an improvement over the maximum matching algorithm (Ruths and Ruths 2014).

*Preferential Matching algorithm* (Zhang et al. 2014) proposed an algorithm to find driver nodes by using preferential matching. They had designed an iterative preferential matching method in which nodes are sorted in the ascending order of their degrees and they are denoted as $M$ as the number of preferential matching nodes. The method starts from the sub graph $H_0$ with the lowest-degree node ranked first; at each iterative step $i$, the sub graph $H_i$ is extended by adding the node with the $i$–th rank, and the maximum matching of $H_i$ is calculated based on the previously obtained maximum matching of $H_(i-1)$. This process is repeated until the sub graph $H_i$ is equal to the whole network or until $M$ preferential nodes have been added. Then, a maximum matching of a graph $G$ is obtained. Preferential matching makes sure to find out the maximum number of matched nodes of $H_i$ from the first $i$ ranking nodes. It also ensures that a high-degree node is not matched in early steps because the node is not included in the early subgraphs. Hence, the matching order of the nodes becomes quite similar to the predefined order of node degrees (Zhang et al. 2014).

#### *Ranking driver nodes*

The significance of a node in controlling the overall network can be determined by ranking driver nodes. Some of the methods are described below.

*Control centrality* It calculates the ability of a single node to control a directed weighted network. In a directed network without loops the control centrality of a node is uniquely determined by its layer index i.e. $C_c(i) = l_i$ where $C_c(i)$ represents the control centrality of node $i$ and $l_i$ represents the layer index of that node (Liu et al. 2012). In a directed star each node can be labelled with a unique layer index, it means that the leaf nodes are in the first layer *i.e.,bottom layer* and the central hub is in the second layer *i.e.,top layer*. In this case the control centrality of the central hub equals its layer index (Liu et al. 2012).

*Control capacity* It quantifies the likelihood that a node is driver node. A $\phi(i)$ is defined as the fraction of MDS's in which node $i$ is included. The method utilises a random sampling method to measure control capacity from different MDS's (Jia and Barabási 2013) The probability of node $i$ to be part of a minimum set of driver nodes as $P(D_i)$, is called "control capacity" $\mathcal{K}$ (Jia and Barabási 2013).

*Control range* When a node quantifies the size of the sub-network that the node can effectively control, i.e., the number of nodes, controlled by one driver node (Wang et al. 2012). Therefore a sub–network which can be fully influenced by the node is called its control range (Wang et al. 2012). For the control range $R_i$ of node $i$, it is defined by first calculating how many nodes $i$ controls, when it is a node in some minimum dominating sets. The set of nodes that $i$ controls according to this definition is denoted by that is denoted by $N_i$. $R_i$ is chosen as the maximum value of $N_i$ over all possible minimum dominating sets. For nodes with $\mathcal{K} = 0$, it means that $i$ never appears in any minimum dominating sets.

*Control range similarity* It is measured as structural similarity between two nodes (Wang et al. 2012). There can be many different maximum matchings of the same network. The method identifies the two different matchings of the same network, to gather two minimum input sets a.k.a *driver nodes.* Therefore, the common sub–network controlled by two input sets is defined as control range similarity.

*Control contribution* It describes that for one minimum set of driver nodes, each driver $i$ controls a non-overlapping sub-network of size $N_i$, which can be identified based on the corresponding cactus structure. A cactus is defined as a connected graph in which any two cycles in the graph have at most one vertex in common. Depending on which cactus structure is obtained, $N_i$ can vary, and its distribution is $f(N_i)$. This allows us to define the average $\langle N_i \rangle$ overall minimum sets of drivers in which node $i$ is a driver. We eventually define the probability that a given node is part of the sub-network of size $N_i$ as $P(N_i)$. In the following, the two measurements about $P(D_i)$ and $P(N_i)$ are combined to define the measure control contribution, $C_i$. Let $MDS_i$ denote the set of all driver node sets, which include node $i$ that combines the two parameters Control Range and Control Capacity and is calculated as $(ControlRange * ControlCapacity)$ (Zhang et al. 2019).

*Node ranking by gravity method* In this research work (Yi-Run et al. 2022), the importance of a node is calculated based upon the idea of a number of cores as the mass of the object. The method considers both local topology information and global position information, based on the gravity formula in Newtonian mechanics, and integrates multiple attribute information of nodes, including nodes, The centrality of the kernel, and the structural hole characteristics of nodes.

*Isolating centrlaity (ISC)* In this work (Ugurlu 2022), authors compare and analyze the centrality measures for detecting important nodes. The proposed centrality measure is based upon identifying the isolating nodes, which is calculated by looking into the degree of the node and its neighbor nodes with the minimum degree. Isolating Centrality (ISC) of a node is the product of its degree and its isolated coefficient (Ugurlu 2022).

### Controllability of complex networks

Controllability is the ability to control a given system. Before going into control, we analyze if it is at all possible to control the system. It means that we need to quantify the ability to steer a dynamical system to a desired final state in a finite time (Liu and Barabási 2015). For example, the act of balancing a stick on our hand. We know from our experience that this is possible, suggesting that the system must be controllable (David 1979). The scientific challenge is to decide for an arbitrary dynamical system if it is controllable or not, given a set of inputs (Liu and Barabási 2015). Considering the controllability of complex systems there are two independent factors that contribute towards it. Both factors have a level of complication, which limits the advances in this field. One of the factor is the system's architecture, represented by the network encapsulating how the components interact with each other; and second one are the dynamical properties that depict the time-dependent interactions between the components. Hence, the controllability can be achieved only in the systems where both these perspectives are taken into account, for example, as it has been done in the case of control in biological networks (Wuchty 2014). Recent advances towards quantifying the topological characteristics of complex networks (Strogatz 2001; Whalen et al. 2015; Newman et al. 2006) have shed light on the role of system's architecture in its controllability. Studies reveal that structure-only methods fail to properly characterise control, because there can be many different variations of possible dynamics that may occur in the networks (Liu and Barabási 2016; Zhang et al. 2019; Albert and Barabási 2002; Strogatz 2001; Zañudo et al. 2017; Sun and Ma 2017; Guo et al. 2018; Wuchty 2014; Delpini et al. 2013; Jia and Barabási 2013; Liu et al. 2011; Pasqualetti et al. 2014; Menichetti et al. 2014; Wang et al. 2012; Wang and Chen 2003; Lombardi and Hörnquist 2007; Chen et al. 2014; Gates and Rocha 2016). So, we not only need to consider/study the node behaviour, but also need to incorporate other factors, like role of links (Jia and Barabási 2013) and control profiles (Ruths and Ruths 2014) in the controllability of the complex networks. There is a substantial amount of work that has been done regarding the structural controllability of complex networks. In Chen (2022), we found out that, most of the controllability robustness techniques revolves around the local structure of the network for example, chain motifs and cycles. The global properties that effect the efficiency and effectiveness of controllability yet to be seen.

### *Comparative analysis of controllability frameworks and future challenges*

The following points emphasize the challenges that are part of this research area.

1. Structural controllability methods find the driver nodes to control the network. To find an optimal and energy efficient driver nodes still remains to be an area worth exploring further (Yan et al. 2012). There is a need to work out an optimal solution to find a set of driver nodes that can be used to further propagate control/influence in the network.
2. Then, the complexity of choosing a smaller set of driver nodes arises. It means, given this number, the largest possible subset of the network can be controlled. If we have to restrict to this smaller set, we should have a ranking of driver nodes that allows us to pick those that have the largest impact on controlling the network. Existing measures for such a ranking, for example control capacity, and control range, are not best suited because they only focus on one aspect of driver nodes, either their probability to become a driver or the size of the sub-network they control. Control contribution combines both of these two aspects (Zhang et al. 2019).
3. In the literature, a categorisation of techniques according to the type/kind of network they can control is still missing. To further elaborate this point, there is a need to look into the network structural measures and their relationship with different control measures. For example, a question that, "Which network structural measures are in correlation with the control measures such as driver nodes?" is still needed to be explored.
4. From the literature survey, we find out that, an intersection of control methods and influence models needs to be explored further. We know that there are various seed selection methods i.e., traditional seed selection are already in use, when spreading

the influence in overall network. But, a large amount of work is needed to find out an optimal seed set. We believe that by employing new ways, specifically driver nodes identification methods to identify driver nodes, and then rank those driver nodes by using seed selection methods and other criteria can be beneficial in maximizing the influence spread process in the overall network.

5. Many studies focus on how to quantify the influence of nodes in a complex network (Guo et al. 2016; Lu and Dong 2019) with the hope that if the most influential nodes are chosen to propagate a given phenomenon, then the spread of this phenomenon will be optimal.

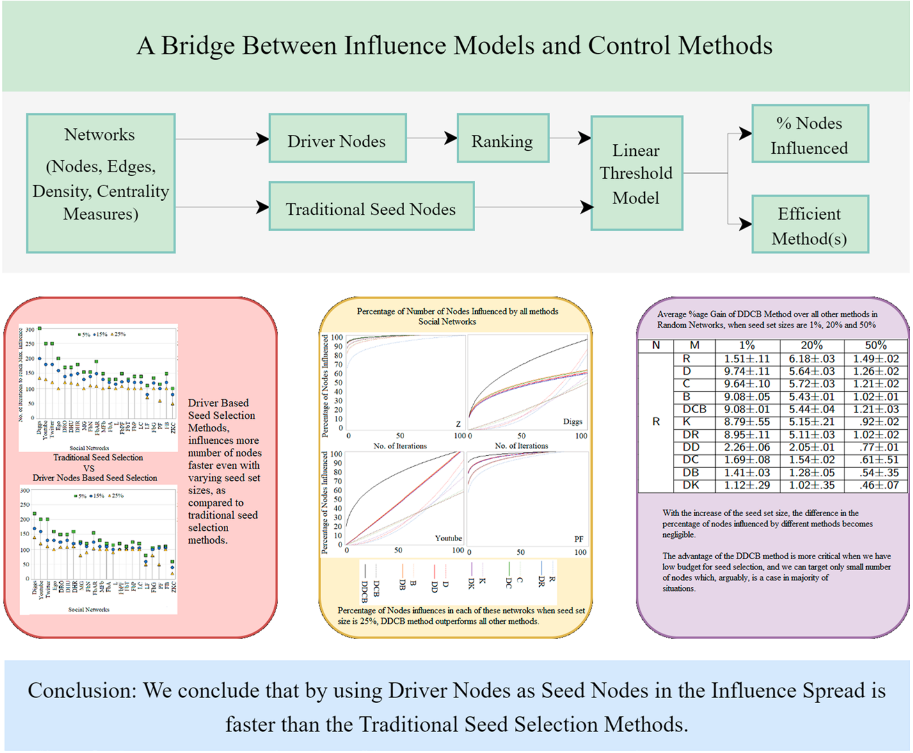
## Research questions and methodology

We see little work done in the space where techniques for finding driver nodes are used to support seed selection strategies; thus, we explore and address this research gap. Below are the research questions that are defined for this study.

**RQ1–How can the concepts from network control be used in the spread of influence field?** The main focus of *RQ1* is to find out if it is feasible to use concepts from the field of network control in the context of influence spread and if so, how it can be done. To answer *RQ1*, we propose new methods that utilise driver nodes as seeds in influence spreading. First, we decide on the method that will be used to select driver nodes. This can be any of the approaches described before: *maximum matching* (Hopcroft and Karp 1973), *minimum dominating set* (Nacher and Akutsu 2012), *control profiles* (Ruths and Ruths 2014) *and preferential matching* (Zhang et al. 2014). To keep the consistency across all the experiments, and because, it is considered a benchmark approach to identify driver nodes across various kinds of networks, we use Minimum Dominating Set approach. The next decision point is to select a technique to rank the identified driver nodes so the obtained ordered list can be used in the seed selection process. This can be done using various approaches, including methods presented in the sec. , for both ranking the driver nodes or ranking seed nodes for influence models. The obtained ranking is used to extract seeds from all identified driver nodes. We have used centrality measures to rank driver nodes, please see sec.  for details. We also propose a new method based upon the centrality measures to rank the driver nodes. The details of this method is given in sec. . The top nodes from the created ranking are used as seeds to investigate the effectiveness of using driver nodes and their ranking in the seed selection process. The first research question *RQ1* and work that is done to answer it can be seen as an initial step to develop a framework where we use network control concepts (driver nodes selection and ranking) in the context of influence spread in networks. **RQ2–How effective can be the implementation of concepts originating from control field in the influence field?** To answer *RQ2* we need to measure the effectiveness of the driver nodes selection and ranking approaches when we use them as seed selection strategies in influence models. When we run experiments, we need to be able to assess to what extent the methods identified in *RQ1* are able to improve the influence spread over the traditional seed selection approaches. The evaluation of seeding strategies on different networks is done on the basis of how much influence the seed nodes are going to spread *a.k.a coverage of influence*, with respect to the spreading time (Jankowski et al. 2018). So, the seed selection method which results in shorter spreading process time and/or has bigger coverage *influence over the network* can be regarded as more effective than the others. **"Control meets influence"** is an idea where we first identify driver nodes in a given network and apply them as seeds in influence model to see the result of influence spread in the network. Figure 1 presents the main stages of the proposed research setup. It defines the inputs, process, and outputs of the conducted research. From the same figure we can see that we utilise traditional seed selection strategies such as random (R), degree centrality (D), betweenness centrality (B) and Kempe seed selection (K). The major outputs will include a comparison by network and a comparison by seed selection method. In network comparison, we observe the percentage of nodes influenced in each network. In method comparison, we compare the performance of seed selection strategies. The performance is measured on the basis of total number of iterations it takes for each method to obtain the highest influence in each network based upon a certain seed set size. Detailed experiment set-up is presented in the next section.

## Experiment setup

In order to bring concepts from control space into the influence agenda, we propose to use Minimum Dominating Set (MDS) to identify driver nodes and then rank those driver nodes using the same ranking methods as in case of seed selection strategies. Additionally, we propose new method—Driver Degree Closeness Betweenness (DDCB) ranking method that first identifies MDS set and then rank the driver nodes on the basis of their average degree, betweenness and closeness centralities. A percentage of ranked



Driver Based Seed Selection Methods, influences more number of nodes faster even with varying seed set sizes, as compared to traditional seed selection methods.

Percentage of Nodes influences in each of these netwroks when seed set size is 25%, DDCB method outperforms all other methods.

Average %age Gain of DDCB Method over all other methods in Random Networks, when seed set sizes are 1%, 20% and 50%

| N | M | 1% | 20% | 50% |
|---|---|---|---|---|
| R | R | 1.51±.11 | 6.18±.03 | 1.49±.02 |
|  | D | 9.74±.11 | 5.64±.03 | 1.26±.02 |
|  | C | 9.64±.10 | 5.72±.03 | 1.21±.02 |
|  | B | 9.08±.05 | 5.43±.01 | 1.02±.01 |
|  | DCB | 9.08±.01 | 5.44±.04 | 1.21±.03 |
|  | K | 8.79±.55 | 5.15±.21 | .92±.02 |
|  | DR | 8.95±.11 | 5.11±.03 | 1.02±.02 |
|  | DD | 2.26±.06 | 2.05±.01 | .77±.01 |
|  | DC | 1.69±.08 | 1.54±.02 | .61±.51 |
|  | DB | 1.41±.03 | 1.28±.05 | .54±.35 |
|  | DK | 1.12±.29 | 1.02±.35 | .46±.07 |

With the increase of the seed set size, the difference in the percentage of nodes influenced by different methods becomes negligible.

The advantage of the DDCB method is more critical when we have low budget for seed selection, and we can target only small number of nodes which, arguably, is a case in majority of situations.

Conclusion: We conclude that by using Driver Nodes as Seed Nodes in the Influence Spread is faster than the Traditional Seed Selection Methods.

**Fig. 1** Control meets influence. The general concept for evaluating the usefulness of driver nodes selection methods in seed selection for influence spread problem

driver nodes are then used as seeds. Linear Threshold model is used to simulate the spread of influence over both randomly generated networks (using random, small-world and scale-free models) and real social networks. A description of both randomly generated and real networks is included in the next subsection.

### Networks

This section includes the tables and figures describing the networks used during our evaluation.

- Figure 2 shows the Random, Small-World and Scale-Free networks sizes (i.e., number of nodes) plotted against their densities. We generated ten network profiles of each Random, Small-World and Scale-Free networks of size ranging from number of nodes equal to 100, 200, 300, 400 and 500. We kept the connections such that to make sure that networks are always connected. In total, 750 networks were generated. This includes the networks starting from smaller densities such as (0.05) to the highest density (1). Density is increasing for every network type when the nodes are from 100–500, due to increase in number of edges. Sometimes it took ten iterations to generate networks with varying sizes and densities, with the goal of achieving the highest density, i.e., 1. More details of network characteristics is given in Supplementary Material. From Table S1, Table S2 and Table S3, We observe that the larger the number of edges, the greater the density of the network. Also, other

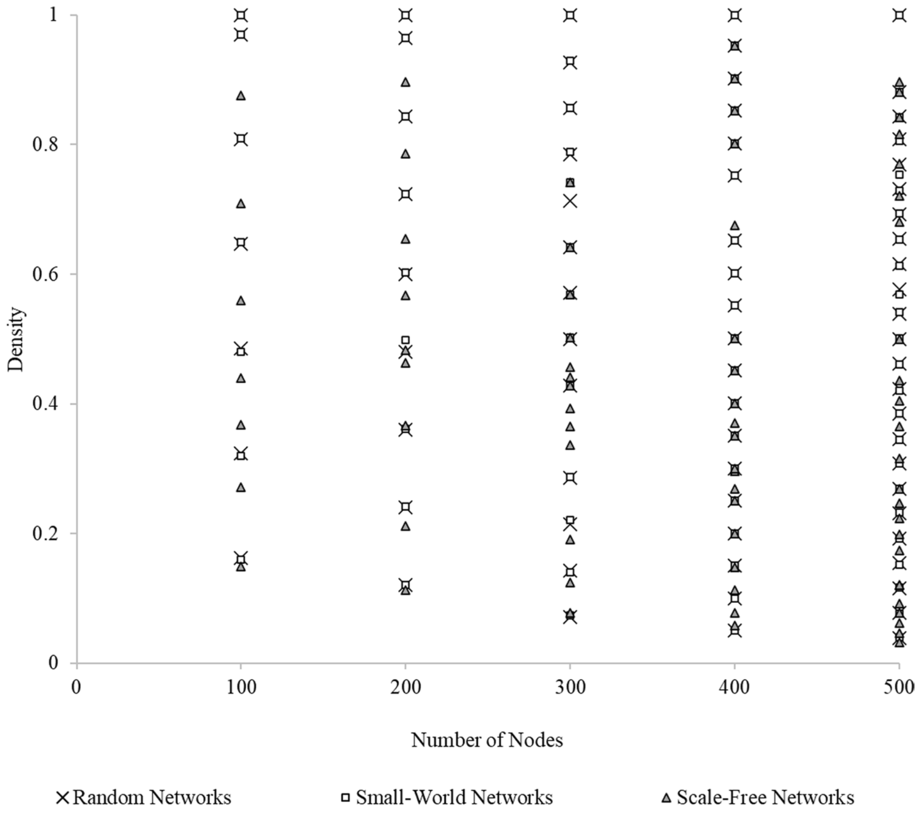


**Fig. 2** Size vs. density in Random, Small-World and Scale-Free networks

centrality parameters, such as, degree, closeness and eigenvector centralities tend to increase when the network size and density increases. Betweenness centrality tend to decrease as the network size and density increases. As we note that number of driver nodes decreases as the density increases in Sadaf et al. (2021). So, naturally, the assumption can be by taking into account these centrality measures, we will have an optimal driver nodes set that can be ranked based upon centrality measures to determine the influential seed set. The details of which comes later.

- Table 1 includes information about twenty-two real social networks used during our experiment and their network structure measures, i.e., the number of nodes, number of edges, and corresponding network density. The networks were downloaded from Stanford Large Network Dataset Collection repository (Leskovec and Krevl 2014).
- Figure 3 shows the densities of social networks with their number of nodes in a logarithmic scale chart.

## Experiments

To be able to answer the research questions, we designed the following experiments.

1 Building Network Profiles: To enable systematic analysis of both traditional and driver–based seed selection strategies, the experiments are conducted on synthetic networks, including Random (R), Small-World (SW) and Scale-Free (SF) network

**Table 1** Network structure measures (social networks)

| Network | Name | Ref. | Nodes | Edges | Density |
|---|---|---|---|---|---|
| Zachary's Karate Club | Z | (Zachary 1977) | 34 | 78 | 0.13903 |
| Facebook | FB | (McAuley and Leskovec 2012) | 4039 | 88234 | 0.01082 |
| Twitter | Twitter | (McAuley and Leskovec 2012) | 23371 | 32832 | 0.00120 |
| Diggs-Friends | Diggs | (Hogg and Lerman 2012) | 1924000 | 3298475 | $2\times10^{-6}$ |
| Youtube | Youtube | (Yang and Leskovec 2015) | 1134891 | 2987625 | $4\times10^{-6}$ |
| Ego-gplus | Ego | (McAuley and Leskovec 2012) | 23629 | 39195 | 0.00140 |
| Librec-ciaodvdnetwork | LC | (Kunegis 2013) | 4658 | 33116 | 0.03050 |
| Librec-filmtrust-trust | LF | (Guo et al. 2014) | 874 | 1309 | 0.03430 |
| petster-frienships-hamster-uniq | PF | (Rossi and Ahmed 2015) | 1858 | 12534 | 0.07260 |
| musae-facebook-edges | MFb | (Rozemberczki et al. 2021) | 22470 | 171002 | 0.00670 |
| Deezer-HR-edges | DHR | (Rozemberczki et al. 2019) | 54574 | 498202 | 0.00330 |
| Deezer-RO-edges | DRO | (Rozemberczki et al. 2019) | 41774 | 125826 | 0.00140 |
| Deezer-HU-edges | DHU | (Rozemberczki et al. 2019) | 47539 | 222887 | 0.00190 |
| musae-git-edges | MG | (Rozemberczki et al. 2021) | 37700 | 289003 | 0.00400 |
| lastfm-asia-edges | L | (Rozemberczki and Sarkar 2020) | 7624 | 27806 | 0.00950 |
| fb-artist-edges | FbA | (Rozemberczki et al. 2019) | 50516 | 819306 | 0.00640 |
| fb-athletes-edges | FbAT | (Rozemberczki et al. 2019) | 13867 | 86858 | 0.00900 |
| fb-government-edges | FbG | (Rozemberczki et al. 2019) | 7058 | 89455 | 0.03590 |
| fb-new-sites-edges | FbN | (Rozemberczki et al. 2019) | 27918 | 206259 | 0.00530 |
| fb-politician-edges | FbP | (Rozemberczki et al. 2019) | 5909 | 41729 | 0.02390 |
| fb-public-figure-edges | FbPF | (Rozemberczki et al. 2019) | 11566 | 67114 | 0.01003 |
| fb-tvshow-edges | FbT | (Rozemberczki et al. 2019) | 3893 | 17262 | 0.02280 |

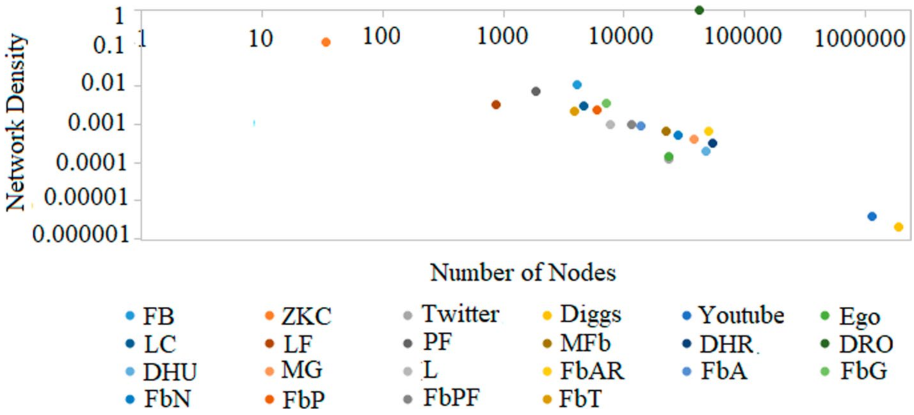


**Fig. 3** Network density verses number of nodes in social networks (adapted from Sadaf et al. 2021)

models as well as real social networks. For comparison purposes, we generated all the networks with same number of nodes and edges. In order to achieve that, we used the method previously applied in Wahid-Ul-Ashraf et al. (2018). We generated 720 networks with 100, 200, 300, 400 and 500 each for Random, Small-World and Scale-Free. More details of the networks is also given in our previous work (Sadaf et al. 2021).

For social networks we used twenty two social networks available in SNAP library (Leskovec and Krevl 2014).

2 Traditional Seed Selection: The methods that are being used in this section are, random seed selection (R), degree seed selection (D), closeness centrality seed selection (C), betweenness centrality seed selection (B), Kempe seed selection (K) and additionally the degree, closeness and betweenness centrality seed selection (DCB) where a top percentage of number of nodes is selected as seeds based upon their average of higher degree, closeness and betweenness centrality values. R, D, C, B and Kempe are most commonly used seed selection methods. We used these methods to find out their usefulness in comparison to the new methods.

3 Driver Seed Selection: One of the contributions of this study is a creation of novel driver-based seed selection strategies. It is a methodological advancement where network control concepts are used in the influence modelling space. First, we identify the driver nodes for each of the network and then we use those driver nodes to define the seed nodes set. The driver nodes are identified by using the Minimum Dominating Set (MDS) method. MDS has been calculated to show the number of driver nodes in the network using MDS method as described in Nacher and Akutsu (2012). Although MDS is a NP-hard problem, reduction rules are a great way to obtain a reduced minimum dominating set, a.k.a Branch and Reduce Algorithm (Weihe 1998). By applying this algorithm, we get a reduced minimum dominating set a.k.a driver nodes. Ranking of driver nodes been previously shown that influential nodes often have higher centrality values (Chen et al. 2012). We focus on ranking driver nodes using various centrality values. Additionally we use Kempe approach to rank driver nodes and as a baseline we use random approach. All proposed ranking strate-

gies are outlined below. In Driver–Random Seed Selection (DR), we select nodes at random from all the driver nodes and they create seed set. In Driver–Degree Seed Selection (DD), we rank the driver nodes in by their degree values and the top ranked nodes become seed nodes. In Driver–Closeness Seed Selection (DC), we rank the driver nodes by their closeness centrality values and the top ranked nodes become seed nodes. In Driver–Betweenness Seed Selection (DB), we rank the driver nodes by their betweenness centrality values and the top ranked nodes become seed nodes. In Driver–Degree–Closeness–Betweenness Seed Selection (DDCB), we rank the driver nodes by averaging the sum of each node's degree, closeness and betweenness centrality values. In Driver–Kempe Seed Selection (DK), we rank the driver nodes based upon their potential to influence the network. The node which is able to spread influence to more number of nodes is ranked higher and in each iteration every new node is evaluated together with those already in the seed set. At the end, the nodes which are able to spread influence to maximum number of nodes are remained in the final seed set, which is pre-defined for the further analysis.

4 Simulating Influence Spread by using LTM: We use both traditional and driver-based seed selection methods to obtain the seed sets and we use those sets as the input to the LTM model to investigate how the spread progresses. In LTM, each agent activates if the number of its active neighbours is bigger or equal than its current activation threshold. We used Bootstrap Percolation to determine the thresholds for LTM. Bootstrap percolation is a process of spread of "activation" on a given network with a given number of initially active nodes. At each step those vertices which have not been active but have at least $\geq 2$ active neighbours become active as well (Janson et al. 2012).

## Results and analysis

Eleven seed selection methods (i.e. Random, Degree, Closeness, Betweenness, Degree–Closeness–Betweenness, Kempe, Driver–Random, Driver–Degree, Driver–Closeness, Driver–Betweenness, Driver–Kempe and Driver–Degree–Closeness–Betweenness) have been tested on synthetic and real world networks. LTM was used to ensure consistency of the results across the board. The findings are discussed from the perspective of (i) synthetic and (ii) real networks.

### Results from synthetic networks

Results from synthetic networks include impact of network density on percentage of nodes influenced and a comparison of all seed selection methods with respect to number of nodes influenced within given time budget and time needed to achieve 100% coverage.

#### *Impact of network density on no. of influenced nodes*

We analysed the influence in networks through the lens of the network density. Figure 2 shows size and density for all the generated networks. We can see that we have networks from lowest to highest densities for each of the given sizes i.e.*number of nodes*. Table 2 shows a comparison of the low and medium density networks with nodes from 100-500 with seed sizes 1%, 10%, 20%, 30%, 40% and 50% in all Random, Small-World

**Table 2** The average percentage of influenced nodes in all generated networks, with seed sizes up to 50% after 20 iterations

| **Seed size** | | | **1%** | **10%** | **20%** | **30%** | **40%** | **50%** |
|---|---|---|---|---|---|---|---|---|
| **N** | **Net.** | **D** | **Percentage of influenced nodes** | | | | | |
| 100 | R | L | 20.0 | 33.0 | 49.0 | 67.0 | 78.0 | 89.0 |
| | | M | 59.0 | 88.0 | 97.0 | **100.0** | **100.0** | **100.0** |
| | SW | L | 30.0 | 49.0 | 58.0 | 64.0 | 82.0 | 93.0 |
| | | M | 57.0 | 92.0 | 99.0 | **100.0** | **100** | **100.0** |
| | SF | L | 18.0 | 51.0 | 67.0 | 79.0 | 87.0 | **100.0** |
| | | M | 62.0 | **100.0** | **100.0** | **100.0** | **100.0** | **100.0** |
| 200 | R | L | 16.0 | 30.0 | 42.0 | 53.0 | 78.0 | 87.0 |
| | | M | 67.0 | 79.0 | 82.0 | 98.0 | **100.0** | **100.0** |
| | SW | L | 16.5 | 40.0 | 54.0 | 67.0 | 78.0 | 89.0 |
| | | M | 67.0 | 91.0 | **100.0** | **100.0** | **100.0** | **100.0** |
| | SF | L | 15.5 | 40.0 | 56.0 | 67.0 | 79.0 | 98.0 |
| | | M | 88.0 | **100.0** | **100.0** | **100.0** | **100.0** | **100.0** |
| 300 | R | L | 25.0 | 46.0 | 57.0 | 68.0 | 78.0 | 82.0 |
| | | M | 64.0 | 88.0 | 97.0 | **100.0** | **100.0** | **100.0** |
| | SW | L | 26.0 | 46.0 | 67.0 | 73.0 | 84.0 | 92.0 |
| | | M | 60.0 | 84.0 | 98.0 | **100.0** | **100.0** | **100.0** |
| | SF | L | 24.6 | 49.0 | 57.0 | 66.0 | 79.0 | 87.0 |
| | | M | 87.0 | 96.0 | **100.0** | **100.0** | **100.0** | **100.0** |
| 400 | R | L | 21.0 | 44.0 | 55.0 | 61.0 | 87.0 | 92.0 |
| | | M | 77.0 | **100.0** | 82.0 | 97.0 | **100.0** | **100.0** |
| | SW | L | 22.0 | 45.0 | 56.0 | 68.0 | 72.0 | 88.0 |
| | | M | 71.0 | 82.0 | 98.0 | **100.0** | **100.0** | **100.0** |
| | SF | L | 28.0 | 46.0 | 64.0 | 71.0 | 87.0 | 99.0 |
| | | M | 76.0 | **100.0** | **100.0** | **100.0** | **100.0** | **100.0** |
| 500 | R | L | 25.0 | 42.0 | 57.0 | 61.0 | 78.0 | 83.0 |
| | | M | 77.0 | 88.0 | 98.0 | **100.0** | **100.0** | **100.0** |
| | SW | L | 26.0 | 35.0 | 65.0 | 76.0 | 78.0 | 87.0 |
| | | M | 60.0 | 72.0 | 78.0 | 89.0 | 92.0 | **100.0** |
| | SF | L | 29.3 | 39.0 | 56.0 | 63.0 | 78.0 | 87.0 |
| | | M | 64.0 | 75.0 | **100.0** | **100.0** | **100.0** | **100.0** |

N - number of nodes, Net. - Network model, D - network density, L - low and M - medium density. The 100% influence is shown in bold font

and Scale-Free networks. The range of low densities is from 0.12 to 0.16, and the range for medium densities is from 0.60 to 0.64. We considered 20 iterations for LTM as a benchmark, because most of the networks reach 100% influence within 20 iterations for networks with medium densities. For complete graphs (density equal to *1*), it is observed that all networks reached the maximum influence in less than 20 iterations, regardless of the seed selection method. Table 3 shows the percentage of influenced nodes in Random, Small-World and Scale-Free networks when the density is low. We can see that for the lowest tested density, i.e., *0.1* the range of level of influence for different network sizes is between $15.5 - -25\%$ when the seed size is 1%. For medium density networks it lies between 57% and 79% for all the methods when the seed size is 1% (Table 4) for network with $N = 100$. That means, more iterations are required with 1% seed size to achieve a 100% influence in all network types and sizes i.e. from 100 to 500 nodes.

**Table 3** Percentage of influenced nodes when the density is low and seed size is 1%

| **Nodes** | | **100** | **200** | **300** | **400** | **500** |
|---|---|---|---|---|---|---|
| **Networks** | **Methods** | **Percentage of influenced nodes** | | | | |
| R | R | 20.0 | 16.0 | 25.0 | 21.7 | 14.6 |
| | D | 23 | 17.5 | 25.7 | 22.2 | 15.2 |
| | C | 25.0 | 17.5 | 23.7 | 22.7 | 15.8 |
| | B | 27.0 | 18.5 | 24.7 | 23.2 | 16.2 |
| | DCB | 27.0 | 18.5 | 25.67 | 23.25 | 16.2 |
| | K | 28.0 | 19.0 | 26.0 | 23.5 | 16.4 |
| | DR | 23 | 17.5 | 25.7 | 22.2 | 15.2 |
| | DD | 29.0 | 20.5 | 25.7 | 34.75 | 37.8 |
| | DC | 31.0 | 21.5 | 28.7 | 35.2 | 38.2 |
| | DB | 32.0 | 22.0 | 29.0 | 35.5 | 38.4 |
| | DK | 33 | 22.5 | 29.3 | 35.7 | 38.6 |
| | **DDCB** | **37.0** | **24.5** | **30.7** | **36.7** | **39.4** |
| SW | R | 27.0 | 16.5 | 26.0 | 22.0 | 15.2 |
| | D | 27.0 | 17.5 | 25.7 | 22.25 | 15.6 |
| | C | 27.0 | 18.5 | 24.3 | 23.25 | 16.2 |
| | B | 28.0 | 19.5 | 24.3 | 23.0 | 16.2 |
| | DCB | 28.0 | 19.5 | 26.3 | 23.0 | 16.2 |
| | K | 30.0 | 20.5 | 27.0 | 23.5 | 16.6 |
| | DR | 30.0 | 18.5 | 24.7 | 23.0 | 16.2 |
| | DD | 35.0 | 20.0 | 25.33 | 37.8 | 39.0 |
| | DC | 41.0 | 21.0 | 28.0 | 24.2 | 17.2 |
| | DB | 43.0 | 22.0 | 28.7 | 24.7 | 17.6 |
| | DK | 42.0 | 21.5 | 28.3 | 24.5 | 17.4 |
| | **DDCB** | **46.0** | **23.0** | **29.3** | **39.4** | **40.0** |
| SF | R | 18.0 | 15.5 | 24.7 | 28.0 | 13.6 |
| | D | 20.0 | 16.5 | 25.3 | 28.7 | 14.2 |
| | C | 21.0 | 17.0 | 23.3 | 21.75 | 14.6 |
| | B | 22.0 | 17.5 | 23.7 | 22.2 | 15.4 |
| | DCB | 22.0 | 17.0 | 25.7 | 21.0 | 14.0 |
| | K | 23.0 | 18.0 | 26.3 | 22.5 | 15.6 |
| | DR | 22.0 | 17.5 | 26.0 | 29.5 | 25.2 |
| | DD | 30.0 | 21.5 | 26.3 | 24.2 | 17.0 |
| | DC | 32.0 | 22.5 | 29.3 | 24.7 | 17.4 |
| | DB | 33.0 | 23.0 | 29.7 | 25.0 | 17.6 |
| | DK | 31.0 | 22.0 | 29.0 | 24.5 | 17.2 |
| | **DDCB** | **37.0** | **25.0** | **31.0** | **33.25** | **28.2** |

The highest percentage is bolded

Table 4 shows the percentage of nodes influenced for seed selection methods in R, SW, and SF networks when density is medium and seed size is 1%. For complete networks all nodes in all networks are influenced. As we increase the seed set size we can see that the percentage of nodes influenced are started to increase regardless of density or size(number of nodes) of the network. But for maximum seed size 50%, the 100% influence is reached in medium as well as low densities networks within 20 iterations. It is also important to point out that in synthetic networks, the small size networks are able to reach influence faster as compared to large size networks. A point of interest is the

**Table 4** Percentage of influenced nodes when the density is medium and seed size is 1%

| **Nodes** | | **100** | **200** | **300** | **400** | **500** |
|---|---|---|---|---|---|---|
| **Networks** | **Methods** | **Percentage of influenced nodes** | | | | |
| R | R | 59.0 | 66.0 | 56.3 | 47.2 | 50.6 |
| | D | 61.0 | 67.0 | 56.7 | 48.0 | 51.4 |
| | C | 61.0 | 67.5 | 57.0 | 48.2 | 52.0 |
| | B | 63.0 | 68.5 | 57.7 | 48.7 | 52.4 |
| | DCB | 63.0 | 68.0 | 58.0 | 49.0 | 52.0 |
| | K | 64.0 | 69.0 | 58.0 | 49.0 | 52.6 |
| | DR | 62.0 | 68.0 | 57.7 | 48.5 | 52.2 |
| | DD | 70.0 | 71.5 | 61.3 | 95.2 | 75.8 |
| | DC | 72.0 | 72.5 | 62.0 | 95.7 | 76.2 |
| | DB | 73.0 | 73.0 | 62.3 | 96.0 | 76.4 |
| | DK | 74.0 | 73.5 | 62.7 | 96.2 | 76.6 |
| | **DDCB** | **78.0** | **75.5** | **64.0** | **97.2** | **77.4** |
| SW | R | 57.0 | 65.5 | 56.3 | 47.2 | 52.0 |
| | D | 62.0 | 67.0 | 56.7 | 48.0 | 52.4 |
| | C | 63.0 | 68.5 | 57.7 | 48.7 | 52.4 |
| | B | 64.0 | 69.0 | 57.7 | 48.5 | 53.0 |
| | DCB | 64.0 | 69.0 | 58.0 | 48.0 | 53.0 |
| | K | 66.0 | 70.0 | 58.3 | 49.0 | 53.4 |
| | DR | 62.0 | 68.0 | 57.7 | 48.5 | 53.0 |
| | DD | 65.0 | 69.5 | 58.7 | 49.2 | 59.0 |
| | DC | 67.0 | 70.5 | 59.3 | 49.7 | 59.4 |
| | DB | 69.0 | 71.5 | 60.0 | 50.2 | 59.8 |
| | DK | 68.0 | 71.0 | 59.7 | 50.0 | 59.6 |
| | **DDCB** | **71.0** | **72.5** | **60.7** | **50.7** | **60.2** |
| SF | R | 59.0 | 65.5 | 55.7 | 46.5 | 60.2 |
| | D | 61.0 | 66.5 | 56.3 | 47.2 | 60.4 |
| | C | 62.0 | 67.0 | 56.7 | 47.5 | 60.6 |
| | B | 63.0 | 67.5 | 57.0 | 47.7 | 61.0 |
| | DCB | 63.0 | 67.5 | 56.0 | 47.5 | 60.0 |
| | K | 64.0 | 68.0 | 57.3 | 48.0 | 61.2 |
| | DR | 63.0 | 67.5 | 57.0 | 47.7 | 61.0 |
| | DD | 71.0 | 71.5 | 59.7 | 49.7 | 62.6 |
| | DC | 73 | 72.5 | 60.3 | 50.2 | 63.0 |
| | DB | 74.0 | 73.0 | 60.7 | 50.5 | 63.2 |
| | DK | 72.0 | 72.0 | 60.0 | 50.0 | 62.8 |
| | **DDCB** | **78.0** | **75.0** | **62.0** | **51.5** | **64.0** |

The highest percentage is bolded

increase and decrease in density, which allows the methods to remain efficient regardless of the increase in number of nodes if the density is also increasing. But if the size (i.e. number of nodes) are increased but density is decreased that the influence process will be effected greatly as we can see from the Tables 3 and 4. A few observations from the experiment suggest that firstly, it takes more iterations when the seed size is smaller, i.e., 1% of the total number of nodes. Secondly, fewer iterations are required to achieve more influence when the densities are higher, regardless of the network topology. Lastly, for complete graphs, we need fewer iterations regardless of the type of network or seed selection method used. From the above observations, we can say that density and seed set size play an important role in determining the efficiency of influence spread in terms of the percentage of influenced nodes. However, as we can see from the results, the seed selection method also matters.

### *Percentage of nodes influenced*

We present a comparison of seed selection methods in the form of the percentage of *gain* in the influence of DDCB over Random, Small-World and Scale-Free networks in Table 5. The percentage of gain of influence for the DDCB method is calculated by subtracting from the percentage of nodes influenced by DDCB the percentage of nodes

**Table 5** Average percentage gain of DDCB method over R, D, C, B, DCB, K, DR, DD, DC, DB, and DK methods in Random (R), Small-World (SW) and Scale-Free (SF) networks

| | | **Seed Size** | | | | | |
|---|---|---|---|---|---|---|---|
| **N** | **M** | **1%** | **10%** | **20%** | **30%** | **40%** | **50%** |
| R | R | 1.51±.11 | 8.2±.06 | 6.18±.03 | 4.87±.03 | 3.31±.03 | 1.49±.02 |
| | D | **9.74±.11** | 7.56±.06 | 5.64±.03 | 4.41±.03 | 2.97±.03 | 1.26±.02 |
| | C | **9.64±.10** | 7.74±.05 | 5.72±.03 | 4.49±.03 | 2.91±.02 | 1.21±.02 |
| | B | **9.08±.05** | 7.46±.02 | 5.43±.01 | 4.02±.01 | 2.64±.01 | 1.02±.01 |
| | DCB | **9.08±.01** | 7.74±.07 | 5.44±.04 | 4.49±.03 | 2.64±.06 | 1.21±.03 |
| | K | 8.79±.55 | 7.18±.53 | 5.15±.21 | 3.79±.45 | 2.46±.03 | .92±.02 |
| | DR | 8.95±.11 | 7.00±.06 | 5.11±.03 | 3.95±.03 | 2.64±.03 | 1.02±.02 |
| | DD | 2.26±.06 | 2.26±.06 | 2.05±.01 | 2.07±.01 | 1.07±.01 | .77±.01 |
| | DC | 1.69±.08 | 1.69±.09 | 1.54±.02 | 1.61±.04 | .77±.05 | .61±.51 |
| | DB | 1.41±.03 | 1.41±.04 | 1.28±.05 | 1.38±.09 | .61±.45 | .54±.35 |
| | DK | 1.12±.29 | 1.13±.45 | 1.02±.35 | 1.15±.85 | .46±.65 | .46±.07 |
| SW | R | 5.33±.05 | 3.97±.02 | 4.02±.02 | 4.59±.04 | 2.21±.03 | 1.08±.02 |
| | D | 4.7±.05 | 3.41±.02 | 3.49±.02 | 4.13±.04 | 1.9±.02 | .95±.01 |
| | C | 4.08±.02 | 3.92±.01 | 5.41±.01 | 4.82±.02 | 3.26±.03 | .91±.01 |
| | B | 3.95±.03 | 3.11±.01 | 3.33±.01 | 3.81±.03 | 1.15±.01 | .71±.01 |
| | DCB | 3.95±.03 | 3.41±.03 | 5.41±.01 | 3.8±.02 | 1.15±.05 | .90±.04 |
| | K | 3.4±.04 | 2.51±.01 | 2.8±.02 | 3.30±.03 | .90±.05 | .54±.06 |
| | DR | 4.26±.05 | 2.85±.02 | 2.95±.02 | 3.80±.04 | 1.15±.02 | .70±.01 |
| | DD | 1.69±.06 | 1.71±.06 | 1.31±.01 | 2.05±.01 | 1.02±.01 | .54±.02 |
| | DC | 1.12±.91 | 1.12±.34 | 1.02±.04 | 1.64±.36 | .82±.34 | .38±.57 |
| | DB | .57±.21 | .57±.04 | .51±.11 | 1.23±.34 | .61±.28 | .23±.43 |
| | DK | .85±.04 | .85±.38 | .77±.08 | 1.43±.91 | .72±.54 | .31±.59 |
| SF | R | 6.97±.06 | 5.97±.03 | 5.13±.02 | 4.36±.03 | 3.43±.04 | 1.61±.02 |
| | D | 6.36±.06 | 5.41±.03 | 4.56±.02 | 3.90±.03 | 3.15±.04 | .14±.02 |
| | C | 5.79±.04 | 5.13±.03 | 4.00±.02 | 3.44±.02 | 2.99±.03 | 1.38±.02 |
| | B | 5.49±.06 | 4.85±.02 | 4.00±.01 | 3.46±.02 | 2.87±.03 | 1.15±.02 |
| | DCB | 5.79±.91 | 5.13±.65 | 4.00±.37 | 3.46±.91 | .87±.11 | .91±.31 |
| | K | 5.21±.55 | 4.57±.87 | 3.72±.84 | 3.23±.04 | 2.72±.06 | 1.05±.13 |
| | DR | 6.05±.06 | 4.85±.03 | 4.00±.02 | 3.46±.02 | 2.87±.03 | 1.15±.02 |
| | DD | 2.15±.06 | 2.15±.06 | 2.26±.05 | 1.79±.04 | .92±.01 | .61±.01 |
| | DC | 1.54±.04 | 1.54±.08 | 1.70±.93 | 1.30±.08 | .72±.11 | .50±.26 |
| | DB | 1.22±.33 | 1.23±.54 | 1.41±.09 | 1.02±.06 | .61±.73 | .40±.34 |
| | DK | 1.85±.11 | 1.84±.03 | 1.97±.93 | 1.54±.05 | .82±.17 | .53±.66 |

Seed set size is expressed as % of the total number of nodes. Each cell in the table shows the average and standard deviation of percentage gain of DDCB over other methods influenced when using other methods (i.e. R, D, C, B, DCB, K, DR, DD, DC, DB, DK, DDCB). The overall gain for Random, Small-World and Scale-Free networks for a particular number of nodes is calculated by taking the average gain over all generated networks of one size (*i.e* $N = 300$). We compute the gain using the level of influence after 20 iterations, as this is the earliest point that the DDCB (and hence any) seed selection method reaches 100% influence. We noted that, as expected, the percentage gain of DDCB method is the highest over Random seed selection method (i.e. 10.51%) in Random networks. However, Table 5 shows that the DDCB method outperforms all evaluated seed selection methods.

Additionally, the results in Tables 3 and 4 show that the traditional seed selection methods do not perform as well as their 'sibling' driver based methods. By 'sibling' method, we denote a pair of methods where ranking is done using the same approach, but one is a driver-based method (only driver nodes are ranked) and the other is not (all nodes are ranked). It could be because none of the ranking methods incorporates the fact that even the highest degree node can be clustered. As a cluster in a network is a set of densely connected nodes that is sparsely connected to other clusters in the network, so it is unnecessary to target all the highest degree nodes that may be in only one or few clusters. That is why, we have more influenced nodes in driver based methods overall. The driver nodes are selected in a way to enable control over the whole network and not only its parts, so they provide better coverage of the network. These observations suggest that if we rank driver nodes based on centrality measures when they are to be used as seeds, the influence spread process produces better results than the benchmark methods such as randomly generated seeds or most commonly used degree based methods. Table 5 shows the results when 1%, 10%, 20%, 30%, 40% and 50% of driver nodes are selected as seeds. With the increase of the seed set size, the difference in the percentage of nodes influenced by different methods becomes negligible. Thus, the advantage of the DDCB method is more critical when we have low budget for seed selection, and we can target only small number of nodes which, arguably, is a case in majority of situations. The highest percentages (greater than 9 percent) are highlighted in bold font in Table 5.

### *Critical difference diagrams*

Figure 4 shows a comparison between all the methods used to generate influence over all of the generated networks. The critical difference diagram shows whether the results (expressed as % of nodes influenced) for various methods are significantly different from each other. The confidence level used is $\alpha = 0.05$. Critical difference diagrams use the

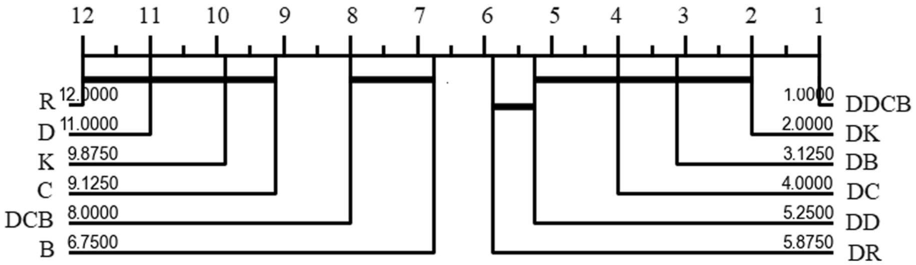


**Fig. 4** Critical difference diagram for generated networks

Wilcoxon-Holm method (Holm 1979) to determine the statistical significance of the results. The lower the rank (further to the right) the better performance of a model under the particular masking rate compared to the others on average. Horizontal line segments group together methods with ranks that are not significantly different in terms of spreading influence. The percentage of number of nodes influenced is calculated for each method for seed sizes 1%, 10%, 20%, 30%, 40% and 50%. The diagrams show that, in generated networks, driver based methods are critically different from traditional seed selection methods in terms of percentage of influenced nodes. Moreover, the DDCB method consistently outperforms other methods and ranks as no. 1 across the board. Other driver–based methods, with exception of random approach (DR), although they outperform traditional methods, are not statistically significantly different between each other. When looking at the traditional methods there is more statistically significant difference between centrality based methods, e.g., degree and closeness centrality ranking methods are worse than betweenness centrality.

This indicates that the key to good seed selection method is rather the fact that we first select driver nodes and rank those than the ranking method itself. Selecting driver nodes enables to effectively reduce number of nodes to be ranked and in the same time ensures that selected nodes are good influencers as they can control the underlying structure.

## Results from real social networks

This section contains the results from real social networks.

### *Percentage of influence gain by DDCB method over other methods in the social networks*

Table 6 shows the percentage of influence gained over all other methods by DDCB method after 100 iterations. We can see that over Random method, percentage gain is the highest. It can be seen, that DDCB method gained more influence over traditional methods as compared to driver based methods. This means that driver based methods, regardless the applied ranking method, do increase the spread of influence over a network. We observe an increase of number of influenced nodes as the process progresses. This leads to the reduction of gain achieved by DDCB over other methods and eventually, when 100% of influenced nodes is achieved, there is no gain.

### *Percentage of nodes influenced*

We can see from the results that driver–based methods of seed selection (DR, DD, DC, DB, DK, DDCB) are able to achieve full influence i.e. *when all nodes are activated* in less iterations than traditional methods (R, D, C, B, K). With 25% nodes selected as seed nodes for DDCB method, in all the networks, all nodes are activated in 100 or fewer iterations. For R and D methods, for Twitter, FbN, DRO, DHU, FbA, DHR, YouTube and Diggs networks it took more than 100 iterations to achieve 100% influence, additionally, for R method and Twitter, DRO, FbA, DHU, DHR, YouTube, and Diggs natworks it took more than 100 iterations to reach 100% influence. The networks are distributed to be represented in three figures, which are divided on the basis of the similarities between their network densities. We can also notice in Figs. 6 and 7 that for such networks as YouTube, Diggs, DRO, DHU, FbT and FbN the spreading dynamic (trend-lines) based on R and D seed selection start slowly (below $y = x$ line) but then pick up as number of

**Table 6** Gain of DDCB over R, D, C, B, DCB, K, DR, DD, DC, DB and DK in real social networks

| | **R** | **D** | **C** | **B** | **DCB** | **K** | **DR** | **DD** | **DC** | **DB** | **DK** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FB | 24.68 | 23.03 | 23.94 | 24.94 | 24.15 | 23.59 | 20.59 | 20.59 | 21.19 | 21.28 | 20.14 |
| ZKC | 8.18 | 2.00 | 1.82 | 1.09 | 0.95 | 0.73 | 0.18 | 0.18 | 0.27 | 0.00 | 0.00 |
| Twitter | 33.81 | 25.83 | 25.8 | 25.78 | 19.16 | 25.77 | 22.81 | 22.8 | 22.74 | 22.06 | 20.22 |
| Diggs | 38.49 | 36.05 | 35.76 | 35.47 | 35.37 | 38.21 | 19.11 | 17.89 | 16.67 | 15.53 | 18.85 |
| Youtube | 39.00 | 33.02 | 32.12 | 31.79 | 31.59 | 32.92 | 2.51 | 1.71 | 0.91 | 0.11 | 2.45 |
| Ego | 20.83 | 13.34 | 13.33 | 13.33 | 16.15 | 20.81 | 8.64 | 8.62 | 8.14 | 8.05 | 7.89 |
| LC | 30.84 | 24.62 | 24.61 | 24.61 | 24.52 | 30.81 | 21.4 | 21.23 | 20.98 | 20.65 | 21.06 |
| LF | 15.29 | 8.62 | 8.56 | 8.34 | 8.25 | 8.33 | 7.38 | 7.35 | 7.20 | 6.86 | 8.11 |
| PF | 7.62 | 4.66 | 4.43 | 4.21 | 4.13 | 4.25 | 1.94 | 1.78 | 1.64 | 1.60 | 1.71 |
| MFb | 21.44 | 20.16 | 20.11 | 20.11 | 20.10 | 20.11 | 14.07 | 13.80 | 19.70 | 19.43 | 13.80 |
| DHR | 36.77 | 33.42 | 32.21 | 31.00 | 30.90 | 33.20 | 5.78 | 5.26 | 4.73 | 4.21 | 5.01 |
| DRO | 38.43 | 33.74 | 33.42 | 33.22 | 33.12 | 33.45 | 12.50 | 12.40 | 12.13 | 11.94 | 33.18 |
| DHU | 42.4 | 33.77 | 33.52 | 33.33 | 33.13 | 37.85 | 25.35 | 25.02 | 24.84 | 24.61 | 24.33 |
| MG | 27.54 | 25.43 | 25.07 | 25.25 | 25.07 | 25.34 | 15.14 | 14.49 | 14.05 | 9.35 | 13.86 |
| L | 24.55 | 23.25 | 23.04 | 22.82 | 22.79 | 22.81 | 17.34 | 17.11 | 16.75 | 16.71 | 16.70 |
| FbAR | 37.97 | 30.40 | 30.18 | 29.95 | 29.93 | 30.30 | 28.43 | 28.14 | 27.85 | 27.56 | 28.28 |
| FbA | 45.29 | 30.45 | 30.05 | 29.55 | 39.87 | 44.89 | 31.28 | 30.83 | 30.28 | 29.46 | 31.01 |
| FbG | 19.95 | 18.22 | 17.93 | 17.71 | 18.18 | 18.20 | 12.97 | 12.75 | 12.39 | 12.13 | 12.68 |
| FbN | 28.82 | 21.03 | 21.00 | 20.95 | 20.96 | 21.01 | 11.85 | 11.64 | 11.18 | 11.10 | 11.40 |
| FbP | 26.73 | 20.90 | 20.76 | 20.40 | 20.87 | 20.89 | 14.89 | 14.47 | 14.15 | 13.90 | 14.31 |
| FbPF | 34.61 | 30.21 | 29.85 | 29.57 | 29.39 | 29.48 | 25.30 | 25.12 | 24.85 | 24.21 | 25.21 |
| FbT | 23.93 | 20.84 | 23.70 | 23.29 | 16.73 | 16.77 | 18.37 | 17.46 | 12.71 | 12.36 | 16.63 |

Seed set size is $25\%$ of the all nodes. Each cell in the table shows the percentage gain of DDCB over other methods after 100 iterations

iterations increases. This clearly means that if we are looking for fast influence spread, in less number of iterations and with small percentage of seed size, we cannot just rely on Random or Degree based methods. Spreading dynamic for driver–based methods shows faster influence spread as compared to traditional seed selection methods. That means, driver nodes are the influential nodes and then ranking of those nodes based on centrality measures enabled to extract the most influential ones. It means that driver nodes that were originally used as nodes that can control network can also be used as seeds and can actually provide faster influence spread.

The DDCB method shows promising results, but this could also be due to the network structural measures, e.g. density of the underlying social network. Thus, we look into the densities of the selected networks. We can see in case of lower density networks, such as Diggs (0.000002), YouTube (0.000004), Twitter (0.00012) and Ego (0.00014), the trend-line for all the methods starts lower than for the networks with high density such as FB (0.01) or ZKC (0.13). It is worth noting that for all networks, for seed size 25%, DDCB method achieved 100% influence in less than 100 iterations, which can be seen from the trend-lines in Figs. 5, 6 and 7. The point to be highlighted here is: networks with higher densities and smaller size (e.g. ZKC) show a trend-line which depicts the quickest possible influence spread as compared to the rest of the networks. Figures 5, 6 and 7 show the percentage of nodes influenced as a function of number of iterations of spreading process for different seed selection methods and seed set size of 25%. The results show

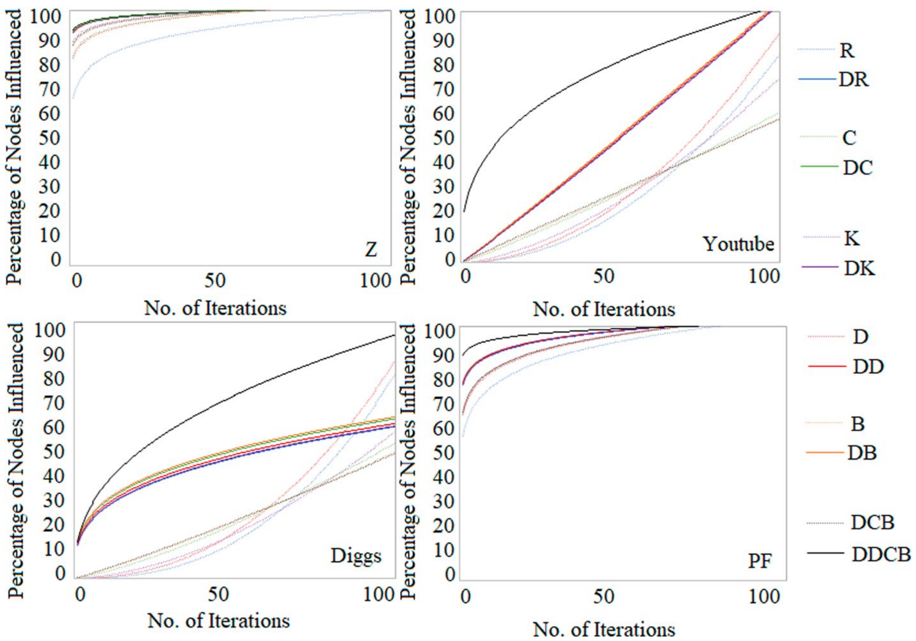


**Fig. 5** The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for Z, Youtube, Diggs, and PF networks. Seed set size is 25%

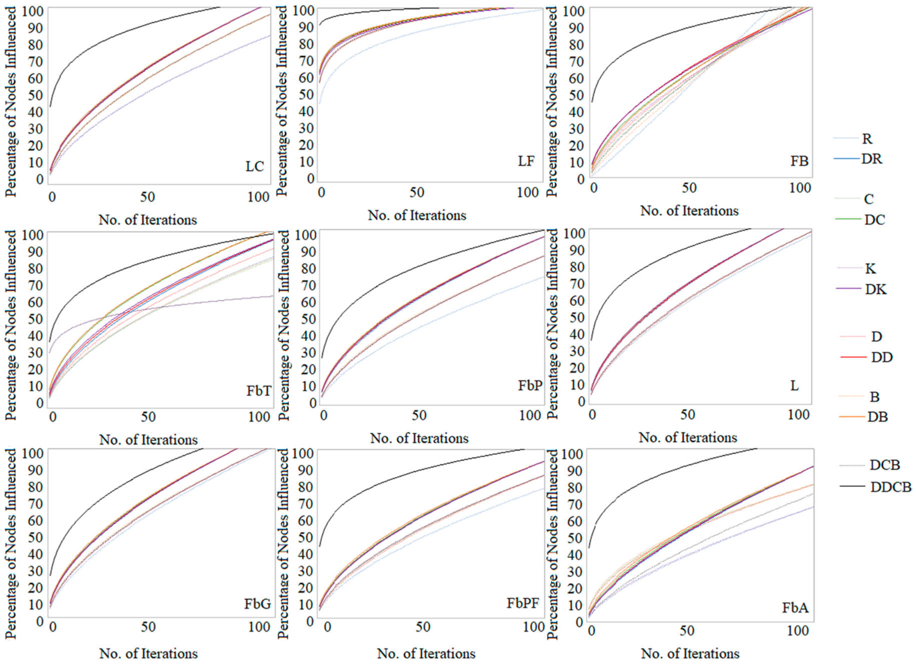


**Fig. 6** The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for LF, FB, FbT, LC, FbP, FbG, L, FbPF, and FbA networks. Seed set size is 25%

that DDCB obtained momentum from the very start of influence spread in all the networks, despite their size. That clearly means, that if we aim to achieve faster influence, i.e., more nodes activated in shorter time, then DDCB is the right choice. That will allow

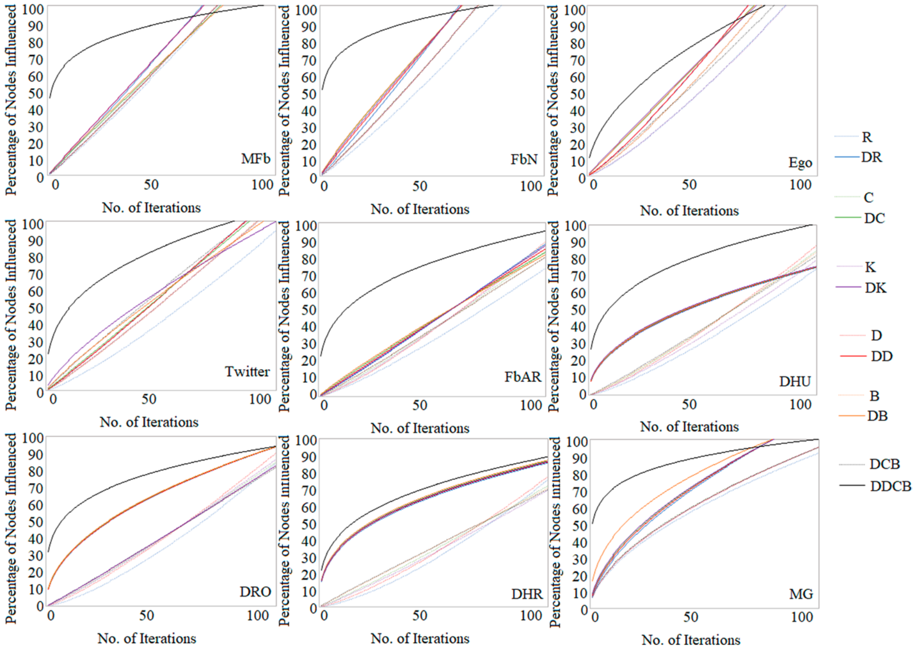


**Fig. 7** The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for MFb, FbN, Ego, Twitter, FbAR, DHU, DRO, DHR, and MG networks. Seed set size is 25%

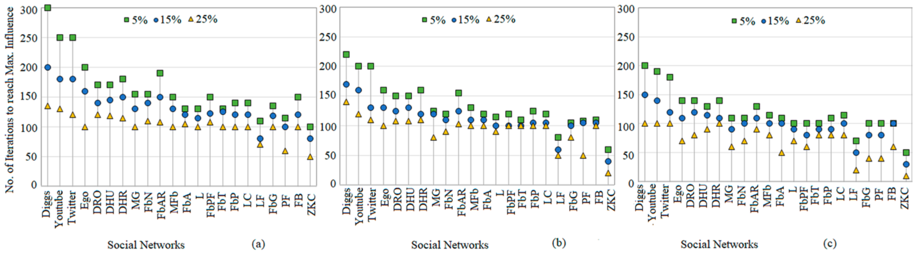


**Fig. 8** Number of iterations needed to reach 100% of influenced nodes using different seed selection strategies: **a** Traditional Seed Selection, **b** Driver Seed Selection, **c** DDCB Seed Selection. Seed set sizes are 5%, 15% and 25%

us to decrease our time (the number of iterations) needed to spread influence in the network, and we can achieve faster influence by using smaller seed set size. If we look at the shape of trend-lines from Figures S6, S7 and S8, we see a lift-off in trend-lines of DDCB method.

### *Number of iterations needed to influence the network*

Figure 8 shows different seed set sizes and the number of iterations each method needs to achieve 100% influence. We sorted the networks in ascending order of their densities to see clearly that the sparser networks needs more iterations to complete the process, irrespective of the seed set size. We worked on achieving the maximum influence by continuing the influence spread to whatever number of iterations is needed. Overall, the number of iterations starts to reduce in all networks when the seed set size increases from 5% to 15% and to 25%. We can see a drop in the number of iterations as we increase the seed size and also as the density of a network increases. This means that some of the strategies to get 100% of nodes activated are (i) to increase the seed size or (ii) to run the process longer. But this is not always possible, for example due to resource limitation.

In Fig. 8, we can see that DDCB method, outperforms both driver based and traditional seed selection methods, and can be used when we want to see more nodes influenced in less time (iterations). We can also see that even driver method, where seeds are ranked in the highest degree, helps propagate influence faster than when using traditional seed selection methods. From the same figure, we can see that Diggs network takes maximum iterations, i.e. up to 300 when seed size is 5%, to achieve maximum influence when we use traditional seed selection methods. What is more, when we increase the seed size to 15% or 25%, we see a sudden drop in number of iterations needed. But still the number of iterations remain higher in Diggs than the rest of the networks. Since Diggs is the lowest density (0.000002) network, we can say our results from simulated networks are relevant here as well. Where we say that the denser the networks, the faster the influence spread. We can see a drop in number of iterations for the same network from Fig. 8, where we compared different seed sizes for driver based seed selection methods. This result indicates that, even in networks with varying structures, driver based methods outperforms the traditional seed selection methods.

### *Critical difference diagram for social networks*

We see similar results (Fig. 9) as we have seen earlier for the generated networks. We found out that all driver–based methods yield statistically better results as compared to their traditional counterparts. This can mean that, if we identify nodes as driver nodes first before selecting them as seeds, it increases their potential to influence more nodes in the network in less iterations as compared to traditional seed selection methods. Not all traditional methods are significantly different from each other. We can see high resemblance in the results of D, K and B methods. If we look at their counterparts methods based on driver nodes DR, DK and DB respectively they are significantly improved than them.

### Time complexity and execution times

From the experiments results, we can see that driver based methods can influence more number of nodes in the networks under consideration. However, there is complexity connected with calculating the driver nodes and ranking. Since our main focus is on calculating the percentage of the nodes influenced, that is why we talk about in how many

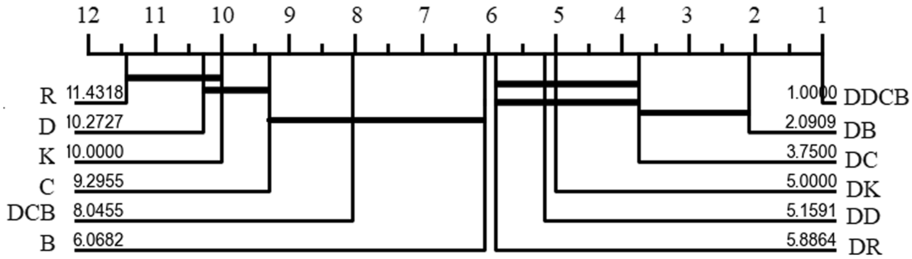


**Fig. 9** Critical difference diagram for real networks

iterations we can reach the maximum influence. There is complexity connected with calculating the driver nodes and ranking. But our main focus is on calculating the percentage of the nodes influenced. We have analysed the execution times for all the seed selection methods in the biggest network, i.e., Youtube. The most important observation is that when comparing all the methods, DDCB method takes fewer iterations (i.e. 26, 20 and 16 at 5%, 15% and 25% seed nodes) to complete the influence process, which is to influence all the nodes. Also, all the driver based methods in comparison to traditional methods require almost half the iterations to influence 100% of the nodes of the networks. All the comparisons are done for 5%, 15% and 25% seed set sizes respectively. DR in comparison to R (77, 74 and 70) takes 37, 32 and 28 iterations to influence the nodes in the networks. DD in comparison to D (75, 74 and 60) takes 35, 30 and 28 iterations. DC in comparison to C (69, 63 and 62) takes 39, 35 and 30 iterations. DB in comparison to B (63, 59 and 55) takes 43, 40 and 32 iterations. DK in comparison to K (61, 57 and 49) takes 38, 32 and 25 iterations. Hence, comparing to all the methods, DDCB method is more efficient than any other method in terms of number of iterations that it takes to influence the overall network nodes. However, despite the fact that driver nodes have higher theoretical complexity, on the other hand, these can be quite useful in the influence spread in synthetic and social networks because driver-based seed selection methods require fewer iterations to influence all nodes.

Table 7, shows the time it takes to execute the algorithms for the methods.

## Conclusion and future work

In this study, we used driver nodes selection methods as seed selection strategies in the influence spreading process to evaluate how they affect the spread time and the influence number of influenced nodes, both in generated and real social networks. This is the first research that brings the fields of control and influence together and proposes new seed selection methods that are inspired by concepts from control theory. We have compared traditional seed selection methods (R, D, C, B, DCB and K) with driver based seed selection methods as their sibling methods (DR, DD, DC, DB, DDCB and DK). We can draw very clear key conclusions based upon the obtained results.

First, based on Sect.  we can say that all driver based seed selection methods outperforms the traditional seed selection methods in terms of percentage of influenced nodes in generated networks as well as real social networks. We further conclude that, if we have a better seed selection set at the beginning of the spreading process, it is high chance that the more number of nodes will be influenced as compared to when we just apply traditional seed selection methods. Moreover, even if we do random seed selection from driver nodes, they perform better than any of the traditional seed selection

**Table 7** Time complexity of calculating different measures

| Centrality | Complexity |
|---|---|
| Degree | $O^2$ |
| Closeness | $O(N * E * d)$, where $d$ is the diameter |
| Betweenness | $O(N * M + N * 2 * logN)$ |
| Driver Nodes | $O(N^2.5)$ |

strategies. The main contribution here is the fact that, when applying driver based seed selection methods, even if the seed size is small those methods are able to achieve higher number of influenced nodes. Hence, if driver based seed selection methods are used, we need smaller seed set for achieving 100% influence in the network. Our results converge from generated networks as well as from social networks. Secondly, we learn that, for sparse networks where density is very low, percentage of influenced nodes is higher in driver based seed selection methods as compared to traditional methods. We see this phenomenon for generated and real networks such as Youtube and Diggs, which are the lowest density networks. We see that even in these networks with small seed sizes driver based methods outperforms their sibling methods. If we complete graph, 100% influence can be achieved, regardless of the seed selection method the influence process is quick. When density is 1, all seed methods work in the same way – they become random. Important conclusion here is when we compared the percentage of influence in lowest and medium density networks for random and as well as social networks. From this comparison /revWhen we analysed the time complexity of seed selection methods, we see that, identifying driver nodes is a very complex task, but we do not need to calculate centrality measures for all nodes and the number of iterations required to reach the 100% influence in a network reduces when we use driver based methods. Thus, actually we are able to save time and resources. To conclude, 100% influence can be achieved, regardless of the seed selection method the influence process is quick. When density is 1, all seed selection methods work in the same way – they become random. The important thing to note is if the network density is very low, like in the case of Diggs network (0.000002), the driver based methods outperforms traditional methods in terms of number of iterations needed to achieve 100% coverage. For synthetic networks, we see the maximum gain that DDCB method has achieved over other techniques is 10.51% which is substantial average gain over Random seed selection method when seed size is 1% as shown in Table 5. The fact that DDCB method outperforms all others for small seed sizes, shows that it has great potential in situations with limited budget where only small number of nodes can be initially activated. This can be concluded based upon the percentage of influenced nodes in all generated networks. Those results are also confirmed by the experiments on real networks.

Our work identifies the relative performance of different seed selection methods in terms of influence spread in a wide variety of network structures, however further work can be done in identifying the characteristics of the individual nodes which lead to them serving as highly effective seed nodes. A deeper understanding of the structural contributions of individual nodes may lead to further improvements to seed selection methods. Driver based methods show improvement over traditional seed selection methods in both synthetic and real–world networks. Results for DDCB are very promising, as this method consistently outperforms other seed selection methods in both kinds of networks. The observed usefulness of our novel approaches addresses the research question "How can the concepts from network control be used in the spread of influence field?" of the research topic.

Finally, we can conclude that in order to achieve maximum influence in fewer iterations, not only density but also seed size and ranking method of driver nodes are important. A further improvement to this may be to identify communities within the network and target seed selection methods at the community rather than at the whole network level.

**Acknowledgements**
This research was partially supported by the Australian Research Council, Grant No. DP190101087: "Dynamics and Control of Complex Social Networks"; the National Science Centre, Poland, under Grant no. 2022/45/B/ST6/04145 and 2021/41/B/HS6/02798; and the EU under the Horizon Europe, grant no. 101086321 OMINO. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the National Science Centre, EU or the European Research Executive Agency.

**Author contributions**
A.S. took the lead in writing the manuscript. All authors provided critical feedback and suggested appropriate research methodology. K.M., L.M., and P.B. helped with their expertise in the areas of complex networks, algorithm design and influence models respectively. All the authors, contributed in analysis of data.

**Availability of data and materials**
The real social networks datasets analysed during the current study are available in the Stanford Large Network Dataset Collection repository, at https://snap.stanford.edu/data. Generated datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Competing interests**
The authors declare that they have no competing of interests.

Received: 26 January 2024 Accepted: 1 July 2024
Published online: 16 July 2024

**References**

Albert R, Barabási A-L (2002) Statistical mechanics of complex networks. Rev Mod Phys 74(1):47
Banerjee S, Jenamani M, Pratihar DK (2020) A survey on influence maximization in a social network. Knowl Inf Syst 62(9):3417–3455
Bass FM (1969) A new product growth for model consumer durables. Manag Sci 15(5):215–227
Bródka P, Musial K, Jankowski J (2020) Interacting spreading processes in multilayer networks: a systematic review. IEEE Access 8:10316–10341
Buldyrev SV, Parshani R, Paul G, Stanley HE, Havlin S (2010) Catastrophic cascade of failures in interdependent networks. Nature 464(7291):1025–1028
Chen G (2022) Controllability robustness of complex networks. J Autom Intell 1(1):100004
Chen D, Lü L, Shang M-S, Zhang Y-C, Zhou T (2012) Identifying influential nodes in complex networks. Phys A 391(4):1777–1787
Chen D-B, Gao H, Lü L, Zhou T (2013) Identifying influential nodes in large-scale directed networks: the role of clustering. PLoS One 8(10):77455
Chen Y-Z, Huang Z-G, Lai Y-C (2014) Controlling extreme events on complex networks. Sci Rep 4:6121
D'Angelo G, Severini L, Velaj Y (2016) Influence maximization in the independent cascade model. In: ICTCS, pp 269–274
David GL (1979) Introduction to dynamic systems: theory, models and applications. Wiley, Chichester
Delpini D, Battiston S, Riccaboni M, Gabbi G, Pammolli F, Caldarelli G (2013) Evolution of controllability in interbank networks. Sci Rep 3:1626
DuanW G (2009) Whinstonab. Inform Cascades Softw Adop Internet 33(1):23
Erlandsson F, Bródka P, Borg A, Johnson H (2016) Finding influential users in social media using association rule learning. Entropy 18(5):164
Erlandsson F, Bródka P, Borg A (2017) Seed selection for information cascade in multilayer networks. In: International conference on complex networks and their applications, Springer pp 426–436
Fath BD, Scharler UM, Ulanowicz RE, Hannon B (2007) Ecological network analysis: network construction. Ecol Model 208(1):49–55
Gates AJ, Rocha LM (2016) Control of complex networks requires both structure and dynamics. Sci Rep 6:24456
Guo W-F, Zhang S-W, Zeng T, Li Y, Gao J, Chen L (2018) A novel structure-based control method for analyzing nonlinear dynamics in biological networks. bioRxiv, 503565
Guo Q, Lei Y, Jiang X, Ma Y, Huo G, Zheng Z (2016) Epidemic spreading with activity-driven awareness diffusion on multiplex network. Chaos Interdiscip J Nonlinear Sci 26(4):043110
Guo G, Zhang J, Thalmann D, Yorke-Smith N (2014) Etaf: An extended trust antecedents framework for trust prediction. In: 2014 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM 2014), pp 540–547. IEEE
Harary F (1972) Recent results on generalized ramsey theory for graphs. In: Graph theory and applications, Springer p 125
Hogg T, Lerman K (2012) Social dynamics of Digg. EPJ Data Sci 1(1):1–26
Holley RA, Liggett TM (1975) Ergodic theorems for weakly interacting infinite systems and the voter model. Ann Prob pp 643–663
Holm S (1979) A simple sequentially rejective multiple test procedure. Scand J Stat pp 65–70

Hopcroft JE, Karp RM (1973) An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. SIAM J Comput 2(4):225–231

Jankowski J, Waniek M, Alshamsi A, Bródka P, Michalski R (2018) Strategic distribution of seeds to support diffusion in complex networks. PLoS One 13(10):0205130

Janson S, Łuczak T, Turova T, Vallier T (2012) Bootstrap percolation on the random graph g—{n, p}. Ann Appl Probab 22(5):1989–2047

Jia T, Barabási A-L (2013) Control capacity and a random sampling method in exploring controllability of complex networks. Sci Rep 3:2354

Kempe D, Kleinberg J, Tardos É (2003) Maximizing the spread of influence through a social network. In: Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining, pp 137–146

Koutsopoulos I, Halkidi M (2018) Efficient and fair item coverage in recommender systems. In: 2018 IEEE 16th international conference on dependable, autonomic and secure computing, 16th international conference on pervasive intelligence and computing, 4th international conference on big data intelligence and computing and cyber science and technology congress (DASC/PiCom/DataCom/CyberSciTech), pp 912–918. IEEE

Kunegis J (2013) Konect: the koblenz network collection. In: Proceedings of the 22nd international conference on world wide web, pp 1343–1350

Leskovec J, Krevl A (2014) SNAP Datasets: stanford large network dataset collection. http://snap.stanford.edu/data

Liu Y-Y, Barabási A-L (2015) Control principles of complex networks. arXiv preprint arXiv:1508.05384

Liu Y-Y, Barabási A-L (2016) Control principles of complex systems. Rev Mod Phys 88(3):035006

Liu Y-Y, Slotine J-J, Barabási A-L (2011) Controllability of complex networks. Nature 473(7346):167

Liu Y-Y, Slotine J-J, Barabási A-L (2012) Control centrality and hierarchical structure in complex networks. PLoS One 7(9):44459

Liu Y, Tang M, Zhou T, Do Y (2015) Improving the accuracy of the k-shell method by removing redundant links: from a perspective of spreading dynamics. Sci Rep 5(1):1

Lombardi A, Hörnquist M (2007) Controllability analysis of networks. Phys Rev E 75(5):056110

Lu P, Dong C (2019) Ranking the spreading influence of nodes in complex networks based on mixing degree centrality and local structure. Int J Mod Phys B 33(32):1950395

Lü L, Zhang Y-C, Yeung CH, Zhou T (2011) Leaders in social networks, the delicious case. PLoS One 6(6):21202

Lü L, Chen D, Ren X-L, Zhang Q-M, Zhang Y-C, Zhou T (2016) Vital nodes identification in complex networks. Phys Rep 650:1–63

McAuley JJ, Leskovec J (2012) Learning to discover social circles in ego networks. In: NIPS, vol. 2012, pp 548–56. Citeseer

Menichetti G, Dall'Asta L, Bianconi G (2014) Network controllability is determined by the density of low in-degree and out-degree nodes. Phys Rev Lett 113(7):078701

Morone F, Makse HA (2015) Influence maximization in complex networks through optimal percolation. Nature 524(7563):65–68

Musiał K, Kazienko P, Brodka P (2009) User position measures in social networks. In: Proceedings of 3rd workshop on social network mining and analysis, pp 1–9

Nacher JC, Akutsu T (2012) Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control. New J Phys 14(7):073005

Nacher JC, Akutsu T (2013) Structural controllability of unidirectional bipartite networks. Sci Rep 3:1647

Narayanam R, Narahari Y (2010) A shapley value-based approach to discover influential nodes in social networks. IEEE Trans Autom Sci Eng 8(1):130

Newman ME, Barabási A-LE, Watts DJ (2006) The structure and dynamics of networks. Princeton University Press, Princeton

Pasqualetti F, Zampieri S, Bullo F (2014) Controllability metrics, limitations and algorithms for complex networks. IEEE Trans Control Netw Syst 1(1):40–52

Pastor-Satorras R, Vespignani A (2001) Epidemic spreading in scale-free networks. Phys Rev Lett 86(14):3200

Qin T, Duan G, Li A (2023) Detecting the driver nodes of temporal networks. New J Phys 25(8):083031

Rossi RA, Ahmed NK (2015) The network data repository with interactive graph analytics and visualization. In: Proceedings of the twenty-ninth AAAI conference on artificial intelligence. AAAI'15, pp 4292–4293. AAAI Press

Rozemberczki B, Allen C, Sarkar R (2021) Multi-scale attributed node embedding. J Complex Netw 9(2):014

Rozemberczki B, Davies R, Sarkar R, Sutton C (2019) Gemsec: graph embedding with self clustering. In: Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining, pp 65–72

Rozemberczki B, Sarkar R (2020) Characteristic functions on graphs: birds of a feather, from statistical descriptors to parametric models. In: Proceedings of the 29th ACM international conference on information & knowledge management, pp 1325–1334

Ruths J, Ruths D (2014) Control profiles of complex networks. Science 343(6177):1373–1376

Sadaf A, Mathieson L, Musial K (2021) An insight into network structure measures and number of driver nodes. In: Proceedings of the 2021 IEEE/ACM international conference on advances in social networks analysis and mining, pp 471–478

Scatà M, Di Stefano A, Liò P, La Corte A (2016) The impact of heterogeneity and awareness in modeling epidemic spreading on multiplex networks. Sci Rep 6(1):1–13

Strogatz SH (2001) Exploring complex networks. Nature 410(6825):268

Sun PG, Ma X (2017) Understanding the controllability of complex networks from the microcosmic to the macrocosmic. New J Phys 19(1):013022

Ugurlu O (2022) Comparative analysis of centrality measures for identifying critical nodes in complex networks. J Comput Sci 62:101738

Vitoropoulou M, Tsitseklis K, Karyotis V, Papavassiliou S (2021) Cover: An information diffusion aware approach for efficient recommendations under user coverage constraints. IEEE Trans Comput Soc Syst 8(4):894–905

Wahid-Ul-Ashraf A, Budka M, Musial K (2018) Netsim-the framework for complex network generator. Proc Comput Sci 126:547–556

Wang XF, Chen G (2003) Complex networks: small-world, scale-free and beyond. IEEE Circuits Syst Mag 3(1):6–20

Wang B, Gao L, Gao Y (2012) Control range: a controllability-based index for node significance in directed networks. J Stat Mech Theory Exp 2012(04):04011

Wang B, Gao L, Zhang Q, Li A, Deng Y, Guo X (2015) Diversified control paths: a significant way disease genes perturb the human regulatory network. PLoS One 10(8):0135491
Watts DJ, Dodds PS (2007) Influentials, networks, and public opinion formation. J Consum Res 34(4):441–458
Wei B, Liu J, Wei D, Gao C, Deng Y (2015) Weighted k-shell decomposition for complex networks based on potential edge weights. Phys A 420:277–283
Weihe K (1998) Covering trains by stations or the power of data reduction. Proc Algorithms Exp ALEX pp 1–8
Weng J, Lim E-P, Jiang J, He Q (2010) Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the Third ACM international conference on web search and data mining, p 261
Whalen AJ, Brennan SN, Sauer TD, Schiff SJ (2015) Observability and controllability of nonlinear networks: the role of symmetry. Phys Rev X 5(1):011005
Wuchty S (2014) Controllability in protein interaction networks. Proc Natl Acad Sci 111(19):7156–7160
Yan G, Ren J, Lai Y-C, Lai C-H, Li B (2012) Controlling complex networks: how much energy is needed? Phys Rev Lett 108(21):218703
Yang J, Leskovec J (2015) Defining and evaluating network communities based on ground-truth. Knowl Inf Syst 42(1):181–213
Yi-Run R, Song-Yang L, Jun T, Liang B, Yan-Ming G (2022) Node importance ranking method in complex network based on gravity method. ACTA Phys Sinica 71(17)
You L, Hoonlor A, Yin J (2003) Modeling biological systems using dynetica—a simulator of dynamic networks. Bioinformatics 19(3):435–436
Zachary WW (1977) An information flow model for conflict and fission in small groups. J Anthropol Res 33(4):452–473
Zang H (2018) The effects of global awareness on the spreading of epidemics in multiplex networks. Phys A 492:1495–1506
Zañudo JGT, Yang G, Albert R (2017) Structure-based control of complex networks with nonlinear dynamics. Proc Natl Acad Sci 114(28):7234–7239
Zareie A, Sakellariou R (2021) Influence maximization in social networks: a survey of behaviour-aware methods. arXiv preprint arXiv:2108.03438
Zhang X, Lv T, Yang X, Zhang B (2014) Structural controllability of complex networks based on preferential matching. PLoS One 9(11):112039
Zhang J-X, Chen D-B, Dong Q, Zhao Z-D (2016) Identifying a set of influential spreaders in complex networks. Sci Rep 6:27823
Zhang Y, Garas A, Schweitzer F (2019) Control contribution identifies top driver nodes in complex networks. Adv Complex Syst 22(07n08):1950014
Zhou H, Ou-Yang Z-c (2003) Maximum matching on random graphs. arXiv preprint arXiv:cond-mat/0309348

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Abida Sadaf** is a Higher Degree Research Student at Department of Computer Science, Faculty of Information and Engineering Technology, University of Technology Sydney. Her major research work revolves around control and influence in social networks.

**Luke Mathieson** is a lecturer in the School of Computer Science at the University of Technology, Sydney. He completed his undergraduate studies in computer science and chemistry at the University of Newcastle, Australia and his doctoral studies in computational complexity theory at the University of Durham in 2010. His research covers topics in algorithms, heuristics, computational complexity, graph theory, logic and computing education.

**Piotr Bródka** is an Associate Professor at the Department of Artificial Intelligence, Wroclaw University of Science and Technology. He completed MSc in Computer Science in 2008 and PhD in 2012. In 2020 he has received Habilitation (D.Sc.) in Information and Communication Technology. In 2012 he also received MSc in Computer Science from Blekinge Institute of Technology, Sweden. He was a Visiting Scholar at Stanford University in 2013 and Visiting Professor at the University of Technology Sydney in 2018 and 2019. He has authored over 70 research articles in the fields of complex networks and computational network science, focusing on the extraction and dynamics of communities, spreading processes and the analysis of multilayer networks. In 2015 he received three years scholarship for the best young scientists awarded by the Polish Government.

**Katarzyna Musial** received her MSc in Computer Science from Wroclaw University of Science and Technology, Poland, and an MSc in Software Engineering from the Blekinge Institute of Technology, Sweden, both in 2006. She was awarded her PhD in November 2009 from WrUST, and in the same year she was appointed a Senior Visiting Research Fellow at Bournemouth University, where from 2010 She was a Lecturer in Informatics. She joined King's College London in November 2011 as a Lecturer in Computer Science. In September 2015 she returned to Bournemouth University where she was an Associate Professor in Computing as well as a Head of SMART Technology Research Group and a member of Data Science Initiative. Currently, she is working as Professor in Network Science in the School of Computer Science at University of Technology, Sydney and is a co-director of Complex Adaptive Systems Lab.