Fakultät für Informatik
der Technischen Universität München

# Density-based clustering in large-scale networks

Klaus Holzapfel

Institut für Informatik der Technischen Universität München
Lehrstuhl für Effiziente Algorithmen

# Density-based clustering in large-scale networks

Klaus Holzapfel

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

| | |
|---|---|
| Vorsitzender: | Univ.-Prof. Dr. Dr. h.c. mult. Wilfried Brauer |
| Prüfer der Dissertation: | 1. Univ.-Prof. Dr. Ernst W. Mayr |
| | 2. Univ.-Prof. Anja Feldmann, Ph.D. |

Die Dissertation wurde am 21.01.2004 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 07.02.2006 angenommen.

# Zusammenfassung

Die Analyse großer Netzwerke mit bis zu Milliarden von Elementen (z.B. das Internet, biochemische Netze oder große soziale Strukturen) bedient sich oft abstrahierender Darstellungen. Dichte-basiertes Clustern ist eine wichtige Technik, um diese Abstraktionen zu erzielen.

Die vorliegende Arbeit diskutiert die Komplexität des Auffindens dichte-basierter Substrukturen in großen Netzwerken, die als einfache, ungerichtete Graphen dargestellt werden. Der Schwerpunkt der Arbeit liegt in der Analyse eines entsprechenden "fixed-parameter" Entscheidungsproblems zum Finden dichter Subgraphen. Es wird eine Klassifizierung des Problems erzielt, das entscheidet, ob ein gegebener Graph einen Subgraphen auf genau $k$ Knoten mit einer bestimmten Mindestanzahl von Kanten enthält (in Abhängigkeit einer festen Funktion $\gamma$ über $k$). In Abhängigkeit des Wachstums der Funktion $\gamma$ wird eine obere Schranke für die Zugehörigkeit zur Klasse der polynomiell lösbaren Probleme und eine untere Schranke für eine **NP**-Vollständigkeit angegeben.

Eine Vielzahl der realen, großen Netzwerke kann gut durch so genannte Power-Law Graphen beschrieben werden, die durch ihre typische Gradverteilung charakterisiert sind. Entsprechend dieser Eigenschaft wird in der Arbeit auch die Komplexität des oben genannten Entscheidungsprobelms betrachtet, wenn die Menge der Eingabegraphen auf diese Graphklasse beschränkt wird. Ähnlich zur vorhergehenden Klassifizierung werden entschrechende Schranken ermittelt.

Abschließend wird die Approximierbarkeit eines zu Grunde liegenden Optimierungsproblems untersucht. Um Möglichkeiten aufzuzeigen, wie die schwache, bestbekannt Approximationsgüte verbessert werden kann, werden für Netzwerke zur Darstellung der Hyperlinkstruktur des World Wide Web einige gängige Heuristiken erläutert.

# Abstract

The analysis of large networks with up to billions of entities (e.g., the Internet, biochemical pathways, or huge social structures) is often based on abstract representations. Density-based clustering is an important technique for deriving these abstractions.

In this thesis we dicuss the complexity of finding density-based substructures in large-scale networks, represented as simple, undirected graphs. The main focus is on the analysis of a corresponding fixed parameter dense subgraph problem. We derive a classification for the decision problem that decides whether a given graph contains a subgraph on exactly $k$ vertices and a given minimum number of edges (calculated by some fixed function $\gamma$ of $k$). Based on the magnitude of function $f$ we state an upper bound for membership in the class of polynomially solvable problems and a lower bound for completeness in **NP**.

A large fraction of real world networks can be sufficiently described in terms of so called power-law graphs, which are characterized by their typical degree sequence. Based on this property, we also discuss the complexity of the above decision problem, when restricted to this class of input graphs. Similarly to the previous classification, we derive corresponding bounds.

Finally, we investigate the approximability of an underlying optimization problem. Further, in order to outline some possibilities to overcome the poor best known approximation ratio, we discuss some commonly used heuristics for networks representing the hyperlink structure of the World Wide Web.

# Acknowledgments

There are numerous people I want to thank for their advice, help and motivation during the last four years.

First of all, I would like to thank my supervisor Prof. Dr. Ernst W. Mayr for the opportunity to be part of his research group while working on my PHD thesis, and for all his support and encouragement. I had the chance to carry out my research freely and to attend international research meetings which makes it possible for me to look back confident on my academic life so far. I also want to thank my second anonymous referee for kindly agreeing to examine my thesis.

Further, I want to express thanks to all my dear colleagues. Special thank goes to Hanjo, Moritz, Stefan and Sven from NetLEA for the pleasant atmosphere and friendship within our research group, and all the hours of fruitful discussions, inspiration and final proof reading. I also want to thank Angelika, Steffi, Mark, Martin, Thomas, and Volker who left LEA earlier and helped me to understand what research is really about and that work must be correlated with fun in order to please.

Finally, I want to address my family. Metaphorically speaking, thank you for having carried me in those times where there was only one set of footprints in the sand. Love is the highest gift, thank you Eva.

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Motivation

Due to enormous increase of computational power in the last decades, it has been possible to store and process information that originates from large-scale networks, which consist of upto billions of entities. While these networks share the common property of large size, they however come from numerous different and uncorrelated fields, such as

- collections of information (e.g., World Wide Web (referred to as WWW), digital libraries, or product catalogs),
- social structures (e.g., acquaintanceship networks, supervisor hierarchies, or dependency structures),
- technical connectivity networks (e.g., Internet structure, power systems, or VLSI design problems), or
- human-independent natural systems (e.g., biochemical pathways, neuronal systems, or food webs).

While, in former times, analysis of corresponding data had to be empirical and based on small test sets, nowadays, it is possible to collect all or at least an comprising majority of the data of interest. When working on this networks, one of the main tasks of sociologists, biologists, engineers, and theoreticians is to analyze and structure this huge amount of data in order to derive knowledge that helps to understand the underlying systems and to improve existing or develop new strategies and solutions for information retrieval problems. Regardless of the above mentioned increase of computational power the applied methods still have to be simple and efficient. More precisely, in most cases, we have to require (pseudo)linear time complexity for corresponding algorithms. However, dependent on the specific questions and desired solutions, this property is often either not achievable (e.g., due to computational hardness) or there is no suitable solution known, so far. Further, most of these systems evolve over time and thus the data cannot be collected nor represented in a consistent state. Therefore, it is impossible to compute optimal solutions for the underlying systems.

As a consequence, in most cases, it is suitable and also computational more efficient to work with approximation results. Using structures or properties that have been observed for the networks, it is possible to build abstractions and thus to decrease data size by grouping entities that share some common property. Finally, based on these reduced data sets, the approximative solutions of the problems can be computed faster (due to the decrease of input size, it is sometimes even possible to disregard the restriction on (quasi-)polynomial algorithms). Especially, when considering pure information retrieval problems, the derived abstractions themselves already represent suitable solutions, in most cases. However, the usage of comprehensive data representations are obviously applicable in various other scenarios, e.g., for search processes in information bases, visualization of geometric data, statistics on huge data sets, or design of communications systems.

The overall technique of deriving abstractive data representations is often referred to as clustering and is briefly discussed in the next section. This discussion is followed by an overview on the content of this thesis. Within this overview we state a specific clustering problem whose complexity is extensively discussed in this thesis. Based on the growing interest on large-scale systems, we turn our attention to cluster problems with inputs chosen from this type of network.

## 1.2 Background on Clustering

In general, clustering is the problem of grouping entities of some input according to some objective function that depends on the properties of the entities [JD88]. Dependent of the choice of clusters (disjoint vs. overlapping, complete vs. partial, etc.) and the type of objective functions (general distance functions vs. metrics, properties of isolated entities vs. sets of entities, etc.), there exists a huge variety of different clustering problems. Two very prominent problems, which are based on distance functions, ask to partition the elements in two (equal-sized) sets in such a way that the sum of the squared distances within the sets is minimized or the sum of distances between entities in different sets is maximized. Unfortunately, similarly to most clustering problems, the two mentioned problems are **NP**-hard. In the following we restrict our discussion on graph-based clustering, i.e. we intend to cluster the vertices of some input graph with respect to the adjacency matrix (or some additional weight functions on the sets of vertices and edges). Information on other cluster categories such as text-based clustering (mainly using vector space models such as latent semantic indexing [BDO95, BDJ99]), geometric-clustering, or cluster-based compression techniques is found in standard textbooks and comprehensive articles [AK95, JD88, Per02, Ber03].

Graph-based clustering, on the one hand side, is important for grouping edge-weighted representations of geometric data or physical networks (e.g., VLSI design problems [Len90]), and on the other hand side, has become more and more important to improve text-based clustering in hyperlinked environments (e.g., see [Cha00]). Nowadays search engines for the WWW, for instance, are highly dependent on the link structure (e.g., search algorithm such as HITS [Kle99], PageRank [BP98], Salsa [LM00], etc.). Within graph-based clustering most problems are partitioning problems, i.e. the resulting subset of vertices are disjoint. Further, the problems can be defined on directed or undirected graphs, respectively. Typically, the directed and undirected problems only differ in minor changes of the objective functions. Similarly, as already mentioned above, it is possible to incorporate weight functions on the edges or vertices.

Despite of this variety of different kinds of graph-based cluster problems, almost all problems aim to optimize either correlation (i.e., high objective value) within the clusters or separation (i.e., low objective value) between clusters, or sometimes even both of them. These two approaches have similar results, if it is possible to partition a graph in a way that the resulting sets of vertices have many intra-cluster edges but only few inter-cluster edges. However, when such a structure is missing or occurs only occasionally (e.g., in semi-structured data), the results may differ extremely, dependent whether optimizing correlation or separation.

There exists a huge variety of objective functions in order to optimize the above mentioned targets. For maximizing the correlation within the clusters it is possible to consider edge-connectivity [GPS90, HS00], induced degrees [BMZ99], the diameter [ESB99], or the number of edges [HK95, SW03], etc. (most of these problems are again **NP**-hard [GJ79]). When minimizing the correlation between vertices of different clusters, the objective function can be described in terms of minimum cuts (e.g., [vD00, FLGC02]). While the minimum cut can be computed in polynomial time by using maximum flow calculations [FF56], in most cases, it is necessary to include size constraints in order to avoid artificial solutions (e.g., see [FLG00]). A very common objective function that uses this idea is known as ratio-cut, where the cut is scaled with the size of the single clusters. Doing so, it is possible to admit somehow unequal cluster sizes in order to get significant smaller cuts. However, when including size restrictions most problems once again get **NP**-hard [GJ79].

Finally, due to the computational hardness of the problem, there are several approximation solutions that use techniques such as greedy algorithms [AITT00, FKP01], move-based approaches [KL70, FM82], spectral techniques [CSZ93], random walks [CHK91], etc. Further, some existing heuristics are based on finding specific substructures that can be expanded to complete clusters (for an example on finding clusters in the hyperlink graph of the WWW, see section 5.3).

## 1.3 Overview and Results

In this thesis, based on the above mentioned necessity of abstracting data in large-scale systems, we analyze the complexity of density-based clustering within corresponding networks. Doing so, we represent the systems as undirected graphs. In order to model large-scale systems, we require that these graphs follow some characteristic degree sequence (referred to as power-law structure) that has been observed for a wide variety of real-world data (for a detailed discussion on large-scale systems we refer to the latter corresponding chapter). Further, the objective function is defined to count the number of edges within a cluster. This choice is based on the observation that many real-world networks follow the small-world behavior. This property, among other things, gives strong evidence for the existence of dense subgraphs (i.e., subgraphs with large average degrees resp. number of edges), which can be used as clusters.

The analysis of density-based clustering is reduced to the basic problem of detecting dense subgraphs within some given input instance. The clustering itself can be achieved by iteratively extracting isolated clusters (see also [AK95, HK95]). Thus, the key algorithmic challenge of the clustering process is condensed of the above described problem of detecting dense subgraphs. In this thesis, we focus on the analysis of the computational complexity of this problem. The discussion itself can be split in two main parts.

1. We analyze the corresponding decision problem on the existence of a subgraph of given size and number of edges, where the number of required edges is calculated by some fixed function $\gamma$ of the number of vertices of the subgraph. Dependent on the fixed function $\gamma$, we achieve to state some intrinsic classification of the complexity of detecting dense subgraphs.

2. Based on the computational hardness that is derived in the first part, we discuss the approximability of a densest subgraph on a given number of vertices.

The discussion of both parts initially analyzes the problem on general input graphs and proceeds with restricting to the above mentioned abstraction of large-scale networks (i.e., power-law graphs). Doing so, we can refine general prove techniques, on the one hand side, and show similar or closely related complexity of both problems, on the other.

The main results can be summarized as follows. Let $\gamma$ be the function that described the number of edges that is required for a subgraph on given number of vertices in order to be a valid solution (throughout the thesis we do not require induced subgraphs, unless stated otherwise; for preliminaries on graphs and complexity notations see appendix A). Dependent on $\gamma$ we can state the following, up to some small gap, complete categorization of the complexity of the subgraph problem. If $\gamma(k) \in k + O(1)$ we state an decision algorithm that runs in time polynomial in the number of vertices of the input graph. If $\gamma(k) \in k + \Omega(k^{\varepsilon})$, for some $\varepsilon > 0$, we prove, for general input graphs, that the problem is **NP**-complete. When restricting to a graph class that abstracts large-scale systems, we prove that the lower bound for **NP**-completeness is linear in $k$, i.e., $\gamma(k) \geq \frac{15}{11}\delta \cdot k$ (where $\delta$ describes the (constant) maximum average degree of the specific class of input graphs).

After the classification of the decision problems, the next part of this thesis analyzes the approximability of finding dense subgraphs of a given size. In the corresponding discussion, we prove equivalence of the problem, when restricting to abstractions of large-scale networks, and the problem for general input graphs, which has been extensively discussed in the literature (a detailed overview one the results is stated in the latter discussion). Thus, we can transfer the approximation results for the general problem to the restricted version and derive some so far best known approximation ratio of $n^{\frac{1}{3}-\varepsilon}$ (with $n$ the number of vertices of the input graph) and strong evidence for the non-existence of a constant approximation ratio [FKP01]. Finally, we get that both, detecting and approximating dense subgraphs within large-scale networks, is computational difficult.

In order to give a comprehensive discussion on density-based clustering, we state some heuristics that try to overcome the poor best known approximation ratio by enumerating dense subgraphs in polynomial time. These heuristics are based on further analysis of the structure of specific large-scale systems. We choose the hyperlink structure of the WWW for illustrating this approach. Based on the large expected number of dense bipartite graphs within the hyperlink structure, we present two polynomial algorithms [KPRT99b, KRK01a] for detecting sets of webpages, that belong to the same cluster and concern some common topic within the WWW.

The remainder of this thesis is structured as follows. In chapter 2 we give a short introduction to large-scale networks. Within this overview, we discuss the two above mentioned properties of this type of networks, namely the small-world characteristic and the power-law behavior. Further, we outline some additional observed properties of the hyperlink structure of the WWW. In chapter 3 we state four graph transformations that are used within the **NP**-completeness proofs. In chapter 4, after a brief discussion on density-based subgraph problems, we state the above mentioned polynomial-time algorithm for detecting subgraphs with small density. Then, we prove the lower bounds for **NP**-completeness of the problem, when using either general input graphs or input graphs that have a power-law degree sequence. At the end of the chapter we complete the analysis with a discussion of the remaining gap when using power-law input graphs. In the subsequent chapter 5, we analyze the approximability of dense subgraph problems and state the above mentioned heuristics. Finally, in chapter 6, we summarize our results and finalize with a discussion on future work.

# Chapter 2

# Large-scale networks

In this chapter, we present an overview on the graph structure of large-scale networks, which are present in almost all areas of natural life. In order to give a rough idea on the huge variety of these networks we only name some of them: the Internet (e.g., the pure physical connections, the AS (autonomous systems) graph, the hyperlink graph of the WWW), social networks (who-knows-who, telephone call graphs, sexual contacts), scientific citation networks, power-grid connections, biochemical networks, food webs, neural connections, etc.

Although these networks are harbored in uncorrelated areas, there are some common properties (e.g., small-world property and power-law behavior), when representing these networks as directed or undirected graphs. In this thesis, we mainly work on representations using undirected simple graphs. The construction of these graphs is straight forward. Every entity of the network is matched to some vertex and any connection between two entities results in an edge connecting corresponding vertices. All multiple edges that occur during construction as well as self loops are removed in order to get simple undirected graphs (for notations on graphs we refer to appendix A.1).

This chapter is organized as follows. First of all, in section 2.1, we describe some generalized property of these networks that is known as the small-world characteristic. This property is one of the first attempts to categorize this class of networks. Then, in section 2.2, we discuss the overall structure of the degree sequence of the underlying graphs. Within the last years, using the increasing computational power, this new graph model for large real-world data, referred to as power-law graphs, has replaced the random graph model $G_{n,p}$, which has been often used before. Finally, in section 2.3, we discuss the hyperlink graph of the World Wide Web and outline differences and similarities to the previously described global characterizations. Further, we list some properties of this network that are used within heuristics in order to overcome the large computational complexity of density-based clustering (see section 5.3). A more extensive introduction to this class of complex networks is given in corresponding comprehensive overviews (e.g., [AB02, BS02, New03]).

## 2.1 Small-world networks

In this section, we discuss the so called small-world property that is attributed to several of the above mentioned networks.

### 2.1.1 History

Due to the huge variety of applications the analysis of large networks has attracted the interest of different research groups. Regardless whether considering transportation problems, spreading of information or diseases, one of the main tasks can be boiled down to the detection of a short connection between two arbitrarily chosen entities within the network. The overall observation for the graphs corresponding to the previously mentioned networks is the short average distance between most of the possible pairs of vertices. This distance appears to have order of $O(\log |V(G)|)$. E.g., there exist several experiments on the who-knows-who graph (i.e., vertices correspond to people, and edges to two people knowing each other) that have shown the existence of short connections. One of the earliest and most cited experiment was performed by Milgram in the 1960s [Mil67]. Within his experiment, he sent letters to persons, referred to as starting persons (either located in Wichita, Kansas or Omaha, Nebraska), who were asked to deliver the letter to some target person living in Cambridge. If a person does not know the target person on a personal basis, he was asked to send the letter to someone (who must be known on a first-name basis) who is more likely to know the target person, and so on. The overall result of the experiment can be summarized as follows: If some letter arrived at the target person (e.g., only 44 of the 160 chains that started in Nebraska, were completed) there were at most ten intermediate persons on the chains with an average number of five. Based on the assumption that other chains were broken due to lack of interest or other difficulties and not due to the non-existence of a chain, one can observe some rather short expected path length (compared to the large amount of people living between Nebraska and Cambridge). Regardless if this assumption is admissible (for an extensive discussion see [Kle02]), we get some idea on the property of this type of network. Further, there are other studies, which underline the observation of the existence of short connecting paths. Within the Hollywood graph, which links actors who played together in a film, there exists the so called Bacon-distance, which states the distance to the "center actor" Kevin Bacon. It has been observed that this distance of actors to Kevin Bacon is less than six [Wat99a]. A similar distance within scientific networks is represented by the Erdös-Number [DCG99], which is also bounded by six. Based on these observations the phrase of "six degrees of separation" is widely used, which assumes that the average distance between any two people in the who-knows-who network is at most six.

Similarly to these general results, almost everyone has already had the experience to meet someone he had never seen before and then occasionally to realize to have a friend in common. Very often this observation is accompanied with the astonished sentence “Isn’t it a small world”. Due to this phrase, Milgram initiated the name small-world networks (for a precise definition of small-world networks, we refer to the next subsection). The existence of short paths has also been observed in other large-scale networks (see general articles such as [Str01, New03]). At this point we want to mention that Albert et al. have proposed some small average distance for the hyperlink graph of the WWW [ABJ99]. However, this result does not hold in this generality but must be restricted to the undirected graph or some appropriate subset of the directed graph (see section 2.3).

### 2.1.2 Small-world graphs

In the following, we define small-world networks in terms of simple undirected graphs. Most definitions of small-world networks do not quantify the required properties but only state the basic intuition (see e.g. [Wat99b, Hay00, New03]). Similarly, Definition 2.1 only formalizes some general properties of graphs. Despite, the first and the third property of the definition are not required to guarantee the existence of short paths, and thus may not be assumed to belong to the definition, they are included since they represent an additional good characterization of the intended graphs class.

**Definition 2.1** *A network is defined to be a small-world network, if the underlying graph structure $G$ has the following three conditions:*

- *$G$ is a sparse graph.*
- *$G$ has small diameter.*
- *$G$ has high cluster tendency.*

In general, the first property is interpreted in such a way that the graph has constant average degree, independent of its size. The second property describes that the (average) diameter of the graph is often assumed to be logarithmic in its number of vertices. Finally, the third parameter models the following property that is often observed in these networks. If a vertex is connected to two neighbors $v$ and $w$, it is very likely that the edges $\{v, w\}$ is also present. This tendency can be measured by the clustering coefficient[1] $C$ with

$$C = \frac{3\ \times\ \text{number of triangles in the network}}{\text{number of connected triples of vertices}}.$$

[1] In the literature, there also exist other definitions on the cluster coefficient. E.g. Watts and Strogatz [WS98] define the clustering coefficient for every individual vertex and measure the average value of all vertices of the graph. The differences of these two definitions and other possible measures are discussed in [New03].

| | $\mathrm{diam}_{\text{real-world}}$ | $\mathrm{diam}_{G_{n,p}}$ | $C_{\text{real-world}}$ | $C_{G_{n,p}}$ |
|---|---|---|---|---|
| Film actors | 3.65 | 2.99 | 0.79 | 0.00027 |
| Power grid | 18.7 | 12.4 | 0.080 | 0.005 |
| C.elegans | 2.65 | 2.25 | 0.28 | 0.05 |

Table 2.1: Comparison of diameter (diam) and cluster coefficient ($C$) in real-world graphs and corresponding $G_{n,p}$ [WS98]

The last property required for small-world networks, describing the cluster tendency, excludes the classes of general random graphs $G_{n,p}$ resp. $G_{n,m}$ that have been introduced by Gilbert[Gil59], and Erdös and Renyi [ER59] (for a comprehensive overview on random graphs we refer to [Bol85]). While it is possible to choose $p$ resp. $m$ in such a way that the graphs fit to the first two properties, the third property does not hold. Assuming constant average degree $d$ (i.e. $p = \frac{p}{n-1}$ resp. $m = \frac{d}{2}n$) almost all of these graphs have diameter logarithmic in the number of vertices [CL01]. However, the cluster coefficient scales with $O(n^{-1})$ and thus gets arbitrary small for large graphs what contradicts the third property. In Table 2.1 we state the different values of the diameter and cluster coefficient for three large-scale networks compared to the average values of graphs $G_{n,p}$ with corresponding edge density.

### 2.1.3 Graph models

In this subsections, we present two models for small-world graphs. These models guarantee the above required properties but do not match the degree sequence that has been observed for real-world date (see section 2.2).

One of the first small-world models has been stated by Watts and Strogatz [WS98]. Starting with a ring lattice on $n$ vertices with even degree $k$ (i.e., $n$ circular ordered vertices, where each vertex is connected to its $\frac{k}{2}$ left and $\frac{k}{2}$ right neighbors) each edge is rewired (i.e., replaced by some arbitrary so far non-existing edge) with some small probability $p$ (see Figure 2.1).

For $p = 0$ the network equals the original lattice with high cluster coefficient but diameter linear in the number of vertices. For small values of $p$ the diameter (for an analysis of the average path length see [NMW00]) decreases fast, while the cluster coefficient only reduces slightly. Finally, for $p = 1$ the final graph is equivalent to some random graph $G_{n,m}$ with small diameter and small cluster coefficient.

Similarly to the above model, Kleinberg investigated the small-world phenomenon on a graph model that is based on a $n \times n$-grid with additional edges connecting arbitrary vertices (according to some variable probability distribution) [Kle00]. Within the graph every vertex $v$ is connected to its neighbors within grid distance at most $p$ (denoted as local contacts; these local contacts guarantee large cluster

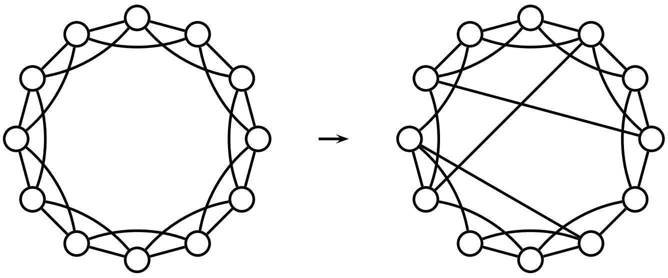

Figure 2.1: Construction of a small-world graph according to the model of Watts and Strogatz (12 vertices, degree 4, and 3 rewired edges)

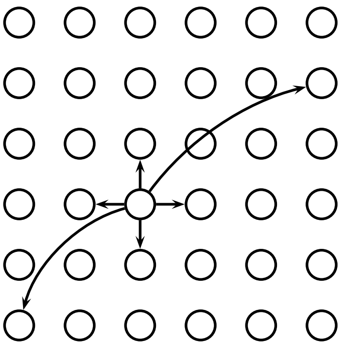

Figure 2.2: Construction of a small-world graph according to the model of Kleinberg ($p = 1$, $q = 2$, n=6)[Kle00]

coefficient). Further, every vertex has $q$ so called long-range contacts, where an edge $\{v, w\}$ is chosen as long-range contact with probability proportional to $d(v, w)^{-r}$, for some global constant $r$ and grid distance $d(v, w)$ of $v$ and $w$. This idea of choosing the connections is illustrated in Figure 2.2.

Based on this definition, Kleinberg has investigated the expected delivery time for some decentral transportation problem (i.e., delivering a message from some vertex $v$ to some different vertex $w$, where the choice of the route is only based on local information, i.e. grid position of the actual neighbors and the target). Kleinberg suggested an algorithm that (starting at $v$) sends the message to that neighbor having minimum grid distance to the target. This decentral algorithm has expected delivery time of at most $O((\log n)^2)$ which is proven to be optimal for the choice of $r = 2$. Expanding this model to higher dimensional grids, it is possible to derive similar results.

### 2.1.4 Summary for small-world networks

In this section we have characterized a wide set of natural large-scale networks using the small-world property. This property abstracts sparse graphs which have high cluster ratio and small diameter, a property that could not be achieved by using random graphs on the same number of vertices and edges. While the cluster ratio must not be confused with the appearance of dense and well separated clusters within the graphs, this value however appears to be some good indicator to do so. This assumption is based on the observation that for most real-world networks with small-world property it is possible to apply clustering techniques to derive dense substructures (e.g., see [SW03]).

Nevertheless, we want to stress that the small-world property only describes tendencies within graphs. Especially, when considering the diameter, the definition has to be restricted to the isolated connected components and it is reasonable to argue on the average path length instead of the maximum occurring value (this allows to deal with possibly occurring artefacts).

## 2.2 Power-law graphs

As mentioned in the previous section, a wide set of large-scale networks is classified to have the small-world characteristic. Despite this property that does not hold for all large-scale systems, the class of corresponding graphs share some further abstracting property regarding their degree sequence. This property that is referred to as power-law degree distribution (or, due to the considerably large fraction of high degree vertices, also heavy-tail distribution), states that the number of vertices with degree $i$ is proportional to $i^{-\beta}$, for some network-typical constant $\beta$. This distribution corresponds to a straight line with slope $-\beta$ in the log-log plot of the cumulative degree distribution. Similarly, when considering random graphs, we can require that the probability of a vertex having degree $i$ is proportional to $i^{-\beta}$. For most large-scale networks it has been observed that $\beta \in \,]\,2\,..\,3\,]$ [New03].

Already in 1896, Pareto observed a similar degree distribution, when analyzing peoples' income [Par96]. Later in 1949, Zipf also received a corresponding distribution for the frequencies of English words [Zip49]. Due to these early observations, the terms Pareto distribution and Zipf law are also often used instead of power-law distribution (however there are minor differences between the precise definitions). In the last decade a power-law degree distribution has been observed for a wide range of large-scale graphs, e.g. for the Hollywood graph [BA99, ASBS00], the connectivity graph of the autonomous systems in the Internet [FFF99, MP01, SFFF03], the hyperlink graph of the Internet [BAJ00], metabolic networks [JTA$^{+}$00], or the telephone call graph [ACL02].

Similarly to the power-law distributions, other characterizations of the degree sequence have been proposed, e.g. the double Pareto distribution [Ree03], the Weibull distribution [Fel00], or the log-normal distribution [Mit03]. The double Pareto distribution is a combination of two power-laws, the Weibull distribution models a variety of life behaviors dependent on some shape parameter $\beta$, and within the log-normal distribution the number of items (e.g., vertices with degree $i$) follows a normal (Gaussian) distribution. Considering the log-log plot of the cumulative degree distribution, these alternatives have similar appearance compared to the standard definition of power-law graphs, for a range of several magnitudes and can be approximated by a power-law distribution, when allowing some corresponding error term.

In this thesis, we use the power-law distribution of the degrees in order to abstract some common characteristic of power-law graphs. In chapter 4 when discussing the complexity of detecting density based clusters, this very general description is used when analyzing the corresponding complexity for this class of graphs. Within this analysis, different so some random graph models, we use a deterministic abstraction that is formalized in Definition 2.2. Due to this simple definition, we are able to set up several general results on the subsets of these graphs (see subsection 2.2.3) that are required within the complexity proofs.

**Definition 2.2** *A $(N,\beta)$-power-law graph (referred to as $(N,\beta)$-PL) is an undirected graph $G=(V,E)$ with the following property*

$$\#_i =_{\text{def}} |\{v \in V \mid \deg(v)=i\}| = \lfloor N \cdot i^{-\beta} \rfloor$$

*where $\#_i$ denotes the number of vertices $v \in V$ with degree $\deg(v)=i$. In order to guarantee that the sum of degrees is even, there might exist an an additional vertex of degree 1. Its occurrence is indicated by variable $\alpha_{N,\beta}$, which is assigned the corresponding value 0 or 1. Thus, $\#_1$ results to $\#_1 = \lfloor N \cdot 1^{-\beta} \rfloor + \alpha_{N,\beta}$.*

*The set of all instances of $(N,\beta)$-PL graphs for some constant value $\beta$ is referred to as $\beta$-power-law graphs or $\beta$-PL.*

Before analyzing the degree sequence corresponding to the above definition, we discuss several theoretical and empirical results for random power-law graphs stated in the literature.

### 2.2.1 Properties of random power-law graphs

There are two main ares of research on $\beta$-PL graphs. First of all, there are several studies in order to determine the empirical properties of real-world large-scale networks with $\beta$-PL degree sequence. Second, some research projects define mathematically strict models for the family of $\beta$-PL graphs. E.g. Aiello, Chung and Lu analyze a model for random graphs, which they call the $(\alpha,\beta)$-graph [ACL01]. In this model the number $y$ of vertices with degree $x$ satisfies $\log y = \alpha - \beta \log x$

(i.e., $y = \frac{e^\alpha}{x^\beta}$). This model, up to rounding and the additive value $\alpha_{N,\beta}$, is equivalent to Definition 2.2. Other mathematical models are based on the configuration model [Bol80, Luc92], the scale-free model [BA99, BAJ00], generating functions [NSW01], or the so called LCD model [BR02b]. Some of these models try to rebuild the dynamical growth of the underlying networks (see subsection 2.2.2). In the following, we state results on the properties of power-law graphs derived by either empirical studies or mathematical analysis.

**Average degree** Using the definition of the power-law degree distribution, it is easily seen that for values $\beta > 2$ the (expected) average degree of the corresponding graph is bounded by some constant. Further, for large graphs the average degree converges and thus is often assumed to be constant [ACL01]. Similarly, for $\beta > 3$, the average degree is less than two. These results match the observation of constant average degree for large-scale networks (with small-world property) [New03].

**Diameter** From the discussion of small-world graphs we know that the diameter of this graph class is logarithmic in the number of vertices. Further we know that many large-scale networks (especially those with $\beta$-PL degree sequence) have been observed to be small-world networks. As mentioned in the previous discussion, the diameter has to be considered within the separate connected components, unless the graph is connected. Sometimes it is even worthwhile to use the average diameter (see the latter discussion on the hyperlink graph of the WWW).

This empirical observation [New03] is well represented in the models of $\beta$-PL graphs. While empirical studies can only state some assumed logarithmic scaling, the mathematical models state more precise diameters. Newman et al. state an analysis on $\beta$-PL and derive that the typical length $l$ of a shortest path between two randomly chosen vertices is $l = \frac{\log N + \log c_1}{\log c_2} + 1$, where $c_1$ and $c_2$ are constants depending on the choice of $\beta$ [NSW01]. Bollobás and Riordan [BR02a] state that the diameter of their model $G_m^{(n)}$, a $\beta$-PL graph on $n$ vertices, satisfies

$$(1-\varepsilon)\frac{\log n}{\log\log n} \ \leq \ \mathrm{diam}(G_m^{(n)}) \ \leq \ (1-\varepsilon)\frac{\log n}{\log\log n}.$$

Further, they point out (independently also stated by Cohen and Havlin [CH03]) that a heuristic argument, using neighborhood expansion, also gives the correct diameter proportional to $\frac{\log n}{\log\log n}$.

**Clustering Coefficient** The clustering coefficient, that was introduced when discussing small-world graphs, has been analyzed empirically by Albert and Barabasi [AB02]. Different to the small-world models discussed in section 2.1 the experimental value $C$ is not constant, but proportional to $n^{-0.75}$. However, analyzing a similar mathematical model Bollobás and Riordan have derived an expectation proportional to $\frac{(\log n)^2}{n}$ and therefore have contradicted the empirical value. Thus, based on the mathematical model, the clustering coefficient of general $\beta$-PL graphs is asymptotically similar to that of random graphs (note that the constant appears to be larger for $\beta$-PL graphs). Therefore, the observed cluster tendency for real-world large-scale systems seams to origin from different properties than the pure power-law degree distribution.

**Connected components** For most large-scale networks is has been observed that there exists a unique giant connected component, while all other connected components are small (e.g., see discussion on the hyperlink graph in section 2.3). This property is also proven by mathematical models. Aiello, Chung, and Lu state the following sizes constraints for $(\alpha, \beta)$-graphs [ACL01]:

1. If $\beta > \beta_0 \approx 3.47875$, a connected component almost surely has size at most $\Theta(n^{2/(\beta)} \log n)$. I.e., there exists no unique giant component.
2. If $\beta < \beta_0$, there is a unique giant component of size $\Theta(n)$.
3. If there exists a unique giant component (i.e., $\beta < \beta_0$) the size of the second largest component decreases when $\beta$ approaches 1:
   (a) If $2 < \beta < \beta_0$, almost surely the size of the second largest component is $\Theta(\log n)$.
   (b) If $\beta = 2$, almost surely the size of the second largest component is $\Theta(\log n / \log \log n)$.
   (c) If $1 \leq \beta < 2$, almost surely the size of the second largest component is $\Theta(1)$.
4. If $0 < \beta < 1$, almost surely the graph is connected.

Based on the observation that most real-world networks have power-law exponent $\beta \in ]2 .. 3]$, the above results of Aiello et al. match the empirical size of the connected components in large-scale networks.

**Maximum Degree** Dependent whether using a probabilistic or a deterministic model for $\beta$-PL graphs the maximum degree of the graphs may vary. Using $(\alpha, \beta)$-graphs (or, similarly, the graph class defined in Definition 2.2) the maximum degree evaluates to some value proportional to $n^{\frac{1}{\beta}}$ (or, $\lfloor N^{\frac{1}{\beta}} \rfloor$, respectively). However, when using randomized models the maximum degree is proportional to $n^{\frac{1}{\beta-1}}$ [ALPH01, CEbAH00]. Drogovtsev et al. have shown that this property also holds for networks generated by using the principle of preferential attachment (see subsection 2.2.2) [DMS01].

Within the latter complexity analysis we assume maximum degree $N^{\frac{1}{\beta}}$. However, we often approximate corresponding summations by integrals with infinite upper bound (instead of $N^{\frac{1}{\beta}}$). Therefore, the possible error that results by the deterministic definition of $\beta$-PL graph is small and has no impact on the overall result.

**Eigenvalues** The spectrum of $\beta$-PL graphs has the following structure [CLV03], where $m$ is the maximum degree of the graphs.

- If $\beta > 2.5$, the largest eigenvalue of a random power-law graph is almost surely $(1 + o(1))\sqrt{m}$. Moreover, the $k$ largest eigenvalues have a power-law distribution with exponent $2\beta - 1$, where $k$ is a function depending on $\beta$, $m$, and the average degree.
- If $2 < \beta < 2.5$, the largest eigenvalue is heavily concentrated at $cm^{3-\beta}$, for some constant $c$ depending on $\beta$ and the average degree. Further, the $k$-th largest eigenvalues is almost surely $(1 + o(1))\sqrt{m_k}$, where $m_k$ is the $k$-th largest expected degree.

The above analysis matches the empirical observation of the power-law distribution of eigenvalues and vertex degrees of Siganos et al. [SFFF03].

A second type of analysis is based on the spectrum of the eigenvalues. Similarly to the spectrum of random graphs $G_n, p$, whose density function follows the semicircle law [Wig58], the density of the spectrum of scale-free graphs (see description in 2.2.2.1) can be characterized as follows [FDBV01]. The density in the central part (i.e., eigenvalues with small absolute value) follows a triangle like structure, while the outer part (i.e., large absolute eigenvalues) has a power-law like tail, both for positive and negative eigenvalues. Further, it has been observed that the triangle gets negligible for $n \to \infty$. Analyzing this distribution and the large value of the principle eigenvalue, we observe that the number of circles with length $l > 4$ increases with $n^{\frac{l}{4}}$ and thus is significantly larger than the number of circles of length $l$ in random graphs, which grow with $O(n)$. However, the number of triangle, i.e. cycles of length 3 is low (as already stated in the above discussion of the clustering coefficient).

**Robustness and Vulnerability** When analyzing the fault tolerance of $\beta$-PL graphs (i.e. the impact on connectivity and path lengths when removing vertices), we have to distinguish two scenarios. Firstly, we can analyze some random node removal, and, secondly, the removal of a set of specific vertices. While the first scenario models some independent failure/disappearance of vertices, the second one can be used to simulate either attacks to the networks, or similarly, to analyze which subset of vertices is almost surely on paths connecting arbitrary vertices (these subset of vertices could be used to control or protect the system).

In the following, we compare these two kinds of node removals for $\beta$-PL graphs and random graphs (for detailed results see [AB02, SCbA$^+$02]). Different to random networks, where every vertex has similar impact to the network, the few vertices with high degree in $\beta$-PL graphs have significant impact on the connectivity and shortest paths. The following results have been observed:

- Random node removal:
  - Within $\beta$-PL graphs it is possible to remove large sets of vertices without choosing too many of the high degree vertices. Therefore, the giant component is reduced successively in size, while the average path length only increases slightly before it also decreases.
  - For random graphs it is possible to determine some threshold behavior. I.e., after removing some specific fraction of vertices the average path length increases significantly, and after removing some further fraction of vertices the giant component disappears entirely.
- Preferential node removal:

  In this cases, both networks behave similar. I.e., if we iteratively remove the vertex with highest degree, both ($\beta$-PL graphs and random graphs) show the threshold phenomenon (described above). Since the number of vertices with high degrees in $\beta$-PL graphs is small compared the total number of vertices, we even observe that the threshold for $\beta$-PL graphs is significantly smaller than that for random graphs.

Based on these results, we can summarize that $\beta$-PL graphs are resilient, when randomly removing vertices, but vulnerable when intentionally attacking the network.

### 2.2.2 Graph models

In the previous subsection, we have summarized several properties of $\beta$-PL graphs based on empirically studies or mathematical precise analysis. To conclude the overview on the literature on this class of graphs, we discuss some of the proposed models to generate power-law graphs. In the above discussion, we have already mentioned the configuration model that dates back to Bollobás [Bol80]. Different to the growth characteristic that is observed for real networks (i.e., the number of vertices increases overtime) this model generates a graph on some fixed number of vertices with given degree sequence. While this model is comfortable to perform mathematical analysis, it is not suitable to represent the evolution of large-scale networks. Therefore, we only concentrate on those models that simulate the so called growth characteristics (some times also (misleadingly) subsumed as the scale-free characteristic; see below).

#### 2.2.2.1 Scale-free characteristic

The term scale-free graphs was initially used by Barabasi and Albert [BA99, BAJ00]. They propose a graph model that is based on the following two properties. First of all, the graphs evolve over time, i.e. starting with some initial graph, at every time step a new vertex is connected to the already existing graph. This property is defined to be the *growth* of the network. Second, the new vertices are connected to the already existing graph by using the so called *preferential attachment*. I.e., every new vertex $v$ is connected using $m$ new edges $\{v, w_i\}$, with $1 \leq i \leq m$, in such a way that $w_i$ is chosen randomly from the existing vertices with probability proportional to the actual degree of these vertices. Thus, vertices with higher degrees are more likely to be chosen as neighbors of $v$ than vertices with smaller degrees. Due to this fact this linking strategy is sometimes also referred to as "the rich get richer".

Based on these two properties, it is possible to start with some initial graph and successively add vertices. Doing so, it is observed that the degree sequence of the resulting graphs converges to a power-law sequence. Thus, ignoring the first few graphs, the resulting graphs share some common structure of the degree sequence independent of their number of vertices. Barabasi and Albert use the term *scale-free* graphs to describe this characteristic.

#### 2.2.2.2 Graph generators

In the following, starting with the above described graph generator or Barabasi and Albert [BA99], we discuss some corresponding growth based graph models.

**Preferential attachment** Using mean-field theory Barabasi, Albert, and Jeong analyzed the above described graph model and derived that the degree sequence fits to a $\beta$-PL distribution with $\beta = 3$ [BAJ99, BAJ00].

Bollobás et al. have stated that the proposed model is not defined mathematically precise [BRST01]. For example, it is left open how to deal with multiple edges and how to choose the initial graph. Bollobás et al. have stated some similar model that also handles these cases (this model generates directed graphs; in order to compare both models we consider only the undirected version of the latter model). Starting with graph $G_1^0$, the empty graph, Bollobás et al. define some graph process $(G_1^t)_{t \geq 0}$, where $G_1^t$ is constructed from $G_1^{t-1}$ by adding a vertex $v_t$ together with a single edge directed from $v_t$ to $v_i$ (assuming vertex set $V(G_1^t) = \{ \ v_i \mid 1 \leq i \leq t \ \}$), where $i$ is chosen randomly with

$$\mathbf{P}(i = s) \quad = \quad \begin{cases} \frac{d_{G_1^{t-1}}(v_s)}{2t-1} & 1 \leq s \leq t-1 \\ \frac{1}{2t-1} & s = t \end{cases}$$

where $d_{G_1^{t-1}}(v_s)$ denotes the degree of $v_s$ in $G_1^{t-1}$. In order to simulate the graphs constructed by Barabasi and Albert, where every vertex was added with $m$ outgoing edges, a similar process $(G_m^t)_{t\geq 0}$ is defined, which is based on the process $G_1^t$ on a sequence of vertices $v_1', v_2, \ldots$ The graph $G_m^t$ is formed from $G_1^{mt}$ by identifying the vertices $v_1', v_2', \ldots, v_m'$ to form $v_1$, identifying $v_{m+1}', v_{m+2}', \ldots, v_{2m}'$ to form $v_2$, and so on.

Using this definition Bollobás et al. have proven the distribution of the number $\#_d$ of vertices of a random graph $G_m^n$ with in-degree equal to $d$ (i.e., with total degree $d' = m + d$), for $0 \leq d \leq n^{\frac{1}{15}}$. Using

$$\alpha_{m,d} = \frac{2m(m+1)}{(d+m)(d+m+1)(d+m+2)} \in \Theta(d'^{-3})$$

they have shown that almost surely $(1-\varepsilon)\alpha_{m,d} \leq \frac{\#_d}{n} \leq (1+\varepsilon)\alpha_{m,d}$, for some fixed $\varepsilon > 0$. This result states a mathematical precise proof of the $\beta$-PL distribution, with $\beta = 3$, that has already been proposed by Barabasi and Albert.

Another enhanced model (proposed by Drinea et al. [DEM01] and Drogovtsev et al. [DMS00]) also includes some further property referred to as initial attractiveness. When building the graph sequences $(G_1^t)$ (resp., $(G_m^t)$), the probability distribution is modified as follows, based on some value $a$ that describes the initial attractiveness:

$$\mathbf{P}(i = s) \quad = \quad \begin{cases} \frac{d_{G_1^{t-1}}(v_s)+a}{(a+1)t-1} & 1 \leq s \leq t-1 \\ \frac{a}{(a+1)t-1} & s = t \end{cases}$$

The mathematical analysis was performed by Buckley and Osthus [BO03]. They have proven that, for $a \geq 1$ and degrees in $[\,0 \,..\, n^{\frac{1}{100(a+1)}}\,]$, the degree sequence follows a $\beta$-PL with $\beta = 2 + \alpha$. Further, for $a = 1$ this model equals the model of Bollobás et al.

**Copying model** Different to the models based on preferential attachment, Kumar et al. proposed a model that, in order to achieve a power-law degree sequence, copies links from so far existing vertices [KRR+00a, KRR+00b]. Doing so, this model tries to simulate the creation of webpages and the corresponding hyperlinks. I.e., the authors propose that when creating a new webpage the creator often copies the links from some other webpage representing similar content. Based on this intuition, the growth process is defined as follows.

At every time step a new vertex is added and linked to the so far existing graph with $d$ out-links, for some constant $d$. In order to determine the out-links, we choose (uniformly at random) a prototype vertex $w$ from all already existing vertices (this vertex is chosen once for all $d$ out-links). For $1 \leq i \leq d$, dependent on some constant $0 < \alpha < 1$, the $i$th out-link is chosen as follows.

- With probability $\alpha$ the destination is chosen uniformly at random from all existing vertices.
- With probability $1-\alpha$ the $i$-th out-link is taken to be the $i$-th out-link of the prototype vertex $w$.

Similarly to this (linear growth) process, Kumar et al. have defined a process, which is referred to as exponential growth process. At time step $t$ $(1+p)^t$ new vertices are added simultaneously. Despite also allowing self loops, the out-links are chosen similarly.

Let $N_{t,r}$ be the expected number of vertices of degree $r$ in the graph that is constructed using the linear growth model. Further, define $P_r = \lim_{t\to\infty} \frac{N_{r,t}}{t}$ to be the asymptotic probability of a vertex to have degree $r$. Kumar et al. have shown that $P_r = \Theta(r^{-\frac{2-\alpha}{1-\alpha}})$ and, thus, the graph obeys a $\beta$-PL degree sequence with $\beta = \frac{2-\alpha}{1-\alpha}$. Similarly, they have proven that the exponential growth model also forms a power-law graph.

Additional to the power-law degree distribution of the copying model, Kumar et al. have shown that the number $K(t,i,j)$ of complete bipartite graphs $K_{i,j}$ at time $t$ is large (i.e., scaling with $t^\varepsilon$ in the copying model) compared to the random graph model $G_{n,p}$ and the $(\alpha,\beta)$ graph model proposed by Aiello, Chung, and Lu [ACL01]. This observation fits very well to the large observed number of complete bipartite graphs in the hyperlink graph of the WWW (see [KPRT99a, KPRT99b]).

**Generalized model** A very general model for $\beta$-PL graphs has been developed by Cooper and Frieze [CF03]. This model includes most of the above generation processes as special cases.

This generalized model also simulates a growth process. Initially, at step $t=0$ the graph consist of a single isolated vertex. At any time $t>0$ either new vertices or edges are inserted. I.e. with probability $1-\alpha$ the procedure NEW inserts a new vertex, and with probability $\alpha$ the procedure OLD adds further edges connecting already existing vertices. The number of edges inserted by procedures NEW and OLD are given by corresponding distributions. Within procedure OLD an initial vertex is chosen from the existing vertices and connected to several terminal vertices. The way how to choose the vertices can be either uniformly at random or according to their degree. More precisely, the process depends on parameters $\alpha$, $\beta$ (not to be mistaken for the exponent of the power-law), $\gamma$, $\delta$, $\mathbf{p}$, and $\mathbf{q}$ that are described below:

- Choice of procedure at step $t$:

| | |
|---|---|
| $\alpha$ | Probability that an OLD vertex generates edges. |
| $1-\alpha$ | Probability that a NEW node is created. |

- Procedure NEW:
  $\mathbf{p} = (p_i)_{i \geq 1}$ Probability that the new node generates $i$ new edges.
  $\beta$ Probability that choices of terminal vertices are made uniformly.
  $1 - \beta$ Probability that choices of terminal vertices are made according to degree.

- Procedure OLD:
  $\mathbf{q} = (q_i)_{i \geq 1}$ Probability that the old node generates $i$ new edges.
  $\delta$ Probability that the initial node is selected uniformly.
  $1 - \delta$ Probability that the initial node is selected according to degree.
  $\gamma$ Probability that choices of terminal vertices are made uniformly.
  $1 - \gamma$ Probability that choices of terminal vertices are made according to degree.

Once again, the above process inserts directed edges. Nevertheless, as also proposed by Cooper and Frieze, it is possible to consider the corresponding undirected graphs. The authors have proven that (for the undirected case) the proportion of vertices of degree $k$ is with high probability asymptotic to $Ck^{-x}$, where $x > 2$ is an explicit function of the parameters of the model.

**Directed graph models** While the above models are often used to analyze undirected power-law graphs (resp., the power-law distributions of either in- or the out-degrees) Bollobás et al. have proposed a directed graph model that enables to achieve power-law distributions for in- and out-degrees, simultaneously. Further, the power-law exponents for these two degree distributions can be chosen independently, where the exponents depend on values of initial attractiveness (see above). Since the focus of this thesis is on undirected graphs, we refer to the original paper for a detailed discussion [BBCR03].

#### 2.2.2.3 Summary on graph models

As a concluding remark, we can state that there are several graph models which try to include the observed growth characteristic of large-scale networks. Based on methods similar to preferential attachment, these models converge to some scale-free power-law degree distribution. However, using only preferential attachment the power-law exponent cannot be adjusted suitable. Nevertheless, based on the described strategies, it is possible to define a class of random graphs in order to model the observed power-law behavior for specific exponents. Once again, these models can only cover some very general aspects of the set of real-word graph class, and thus, cannot replace some detailed analysis of the individual classes of large-scale networks.

### 2.2.3 Analysis of $\beta$-PL degree sequences and construction of subgraphs with bounded average degrees

In this subsection, we derive several results on power-law graphs that are used within the latter proofs. The analysis is split into two parts. Firstly, we analyze the average degree when restricting to specific subsequences of $\beta$-PL graphs. Doing so, we derive an upper bound for the average degree of a degree subsequence of a $\beta$-PL graph that contains all vertices with degree at least 2 (Lemma 2.2). Secondly, we discuss how to generate graphs on degree (sub)sequences in such a way that the average degree of any subgraph of the resulting graphs is bounded by some constant, only dependent of $\beta$ (Lemma 2.4). Within these discussions we use the following definition.

**Definition 2.3** *A sequence $S = [s_i]_{1 \leq i \leq d} \in \mathbb{N}^d$ is a degree sequence if there exists some graph $G$ in such a way that the degrees of $G$ are contained in $[\,1\,..\,d\,]$ and that the number of vertices of $G$ with degree $i$ equals $s_i$, for all $i \in [\,1\,..\,d\,]$.*
*A sequence $S = [s_i]_{1 \leq i \leq d} \in \mathbb{N}^d$ is call a candidate degree sequence if we want to express that the sum of the degrees is even but we have not tested whether a corresponding graph exists.*

Similarly to the usage of graphs within the above definition, we use the terms vertices, edges, and degrees, when arguing on degree sequences.

#### 2.2.3.1 Analysis of degrees of subgraphs

First of all, we investigate some threshold $d$ in such a way that for any $(N, \beta)$-PL graph the sum of degrees of all vertices with degree at least $d$ at most $0.5N$.

**Lemma 2.1** *Let $\beta > 2$ and $N \in \mathbb{N}$ be fixed. The sum of degrees of all vertices with degree at least $d = \left(2\frac{\beta-1}{\beta-2}\right)^{\frac{1}{\beta-2}}$ of a $(N, \beta)$-PL graph is at most $0.5 \cdot N$.*

*Proof:* The sum of degrees $\text{sum}_d$ of the vertices with degree at least $d$ can be bounded at follows

$$\begin{aligned}
\text{sum}_d &= \sum_{i=d}^{N^{\frac{1}{\beta}}} i \lfloor N \cdot i^{-\beta} \rfloor \;\leq\; N \cdot d^{1-\beta} + \int_{i=d}^{N^{\infty}} N i^{1-\beta} di \\
&= N\left(d^{1-\beta} + \frac{1}{\beta-2} d^{2-\beta}\right) \;\leq\; N d^{2-\beta} \frac{\beta-1}{\beta-2} \\
&= N\left(2\frac{\beta-1}{\beta-2}\right)^{\frac{2-\beta}{\beta-2}} \frac{\beta-1}{\beta-2} \;=\; 0.5N
\end{aligned}$$

□

Within the latter **NP**-completeness proofs we use some special degree subsequences of a $(N, \beta)$-PL graph degree sequence. Lemma 2.2 states an upper bound for the average degree of this degree sequences.

**Lemma 2.2** *Let $S$ be a $(N, \beta)$-PL degree sequence with $N = (2\hat{\delta}k)^8 \cdot (2\hat{\delta} + 2k)^\beta$, where $\hat{\delta} = \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} \right\rceil + 2$. Further, let $S'$ be a candidate degree subsequence of $S$ with the following properties:*

- $s'_1 = 0$
- *The number $x$ of missing vertices with degree $i \in [2..\hat{\delta}]$ is at most the number $y$ of missing vertices with degree $j \in [2\hat{\delta}..N^{\frac{1}{\beta}}]$, with respect to $S$ (i.e., $x \leq y$).*

*The average degree of $S'$ is at most $\frac{30}{11}$ of the average degree of any $(N', \beta)$-PL with $N' \geq N$.*

*Proof:* First of all, we consider the degree subsequence $\hat{S}$ of $S$, that contains all vertices of $S$ with degree at least 2. The following calculation shows that the average degree of $\hat{S}$ is at most $\hat{\delta}$.

$$\begin{aligned}
\frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i \lfloor N \cdot i^{-\beta} \rfloor}{\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N \cdot i^{-\beta} \rfloor} &\leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} iN \cdot i^{-\beta}}{\sum_{i=2}^{N^{\frac{1}{\beta}}} (N \cdot i^{-\beta} - 1)} \leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i^{1-\beta}}{\sum_{i=2}^{N^{\frac{1}{\beta}}} i^{-\beta} - \frac{N^{\frac{1}{\beta}}}{N}} \\
&\leq \frac{2^{1-\beta} + \int_2^{\infty} i^{1-\beta} di}{2^{-\beta} + \int_3^{N^{\frac{1}{\beta}}} i^{-\beta} di - N^{\frac{1}{\beta}-1}} \\
&= \frac{2^{1-\beta} + \frac{1}{\beta-2} 2^{2-\beta}}{2^{-\beta} + \frac{1}{\beta-1} \left(3^{1-\beta} - N^{\frac{1-\beta}{\beta}}\right) - N^{\frac{1-\beta}{\beta}}} \\
&= \frac{2^{-\beta} \left(2 + \frac{4}{\beta-2}\right)}{2^{-\beta} + \frac{1}{\beta-1} \left(3^{1-\beta} - \beta N^{\frac{1-\beta}{\beta}}\right)} \leq \frac{2^{-\beta} \left(2 + \frac{4}{\beta-2}\right)}{2^{-\beta}} \\
&= 4 \cdot \frac{1}{\beta - 2} + 2 \leq \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} \right\rceil + 2 = \hat{\delta}
\end{aligned}$$

For the second last inequality we use $3^{1-\beta} \geq \beta N^{\frac{1-\beta}{\beta}}$ or, equivalently, $N^{\frac{1}{\beta}} \geq 3\beta^{\frac{1}{\beta-1}}$. This proposition can easily seen to be true, when using $N^{\frac{1-\beta}{\beta}} \geq 2\hat{\delta} + 2k > 6$ and $3\beta^{\frac{1}{\beta-1}} \leq 6$. Further, we apply $x = \frac{1}{\beta-2} > 0$ to $x \leq 2^x$ (easily seen to be true for all $x > 0$) in order to prove the last inequality.

Based on this upper bound for the average degree of $\hat{S}$, we can remove pairs of vertices $\{v, w\}$ from $\hat{S}$, with $\deg(v) \leq \hat{\delta}$ and $\deg(w) \geq 2\hat{\delta}$, without increasing the average degree of the sequence. Thus, when removing $x$ vertices with degree at most $\hat{\delta}$ and $y$ vertices with degree at least $2\hat{\delta}$, with $y > x$, the average degree of the resulting degree sequence is at most the average degree of $\hat{S}$. Therefore, we can use this bound as an upper bound for the average degree of $S'$ and get.

$$\begin{aligned}
\operatorname{avgdeg}(S') &\leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}{\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N\cdot i^{-\beta}\rfloor} \\
&= \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}{\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N\cdot i^{-\beta}\rfloor}\cdot\frac{\sum_{i=1}^{N'^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor + \alpha_{N',\beta}}{\sum_{i=1}^{N'^{\frac{1}{\beta}}} i\lfloor N'\cdot i^{-\beta}\rfloor + \alpha_{N',\beta}}\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}{\sum_{i=1}^{N'^{\frac{1}{\beta}}} i\lfloor N'\cdot i^{-\beta}\rfloor}\cdot\frac{1+\sum_{i=1}^{N'^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}{\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N\cdot i^{-\beta}\rfloor}\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}{\frac{N'}{N}\sum_{i=1}^{N^{\frac{1}{\beta}}} i\left(\lfloor N\cdot i^{-\beta}\rfloor - 1\right)}\cdot\frac{1+\sum_{i=1}^{N'^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}{\frac{N}{N'}\sum_{i=2}^{N^{\frac{1}{\beta}}} \left(\lfloor N'\cdot i^{-\beta}\rfloor - \frac{N'}{N}\right)}\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \frac{\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}{N-N^{\frac{2}{\beta}}+\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}\cdot\frac{1+N'+\sum_{i=2}^{N'^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}{-N'N^{\frac{1-\beta}{\beta}}+\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}\cdot\operatorname{avgdeg}(\beta,N') \\
&= \left(1-\frac{N-N^{\frac{2}{\beta}}}{N-N^{\frac{2}{\beta}}+\sum_{i=2}^{N^{\frac{1}{\beta}}} i\lfloor N\cdot i^{-\beta}\rfloor}\right)\cdot \\
&\quad\left(1+\frac{1+N'+N'N^{\frac{1-\beta}{\beta}}+\sum_{i=N^{\frac{1}{\beta}}+1}^{N'^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}{-N'N^{\frac{1-\beta}{\beta}}+\sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N'\cdot i^{-\beta}\rfloor}\right)\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \left(1-\frac{1-N^{\frac{2-\beta}{\beta}}}{1-N^{\frac{2-\beta}{\beta}}+2^{1-\beta}+3^{1-\beta}+\int_3^\infty i^{1-\beta}di}\right)\cdot \\
&\quad\left(1+\frac{1+N^{\frac{1-\beta}{\beta}}+\int_{N^{\frac{1}{\beta}}}^\infty i^{-\beta}di}{-N^{\frac{1-\beta}{\beta}}+2^{-\beta}+3^{-\beta}+\int_4^\infty i^{-\beta}di}\right)\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \left(1-\frac{1-N^{\frac{2-\beta}{\beta}}}{1-N^{\frac{2-\beta}{\beta}}+2^{1-\beta}+3^{1-\beta}+\frac{1}{\beta-2}3^{2-\beta}}\right)\cdot \\
&\quad\left(1+\frac{1+2N^{\frac{1-\beta}{\beta}}}{-N^{\frac{1-\beta}{\beta}}+2^{-\beta}+3^{-\beta}+\frac{1}{\beta-1}4^{1-\beta}}\right)\cdot\operatorname{avgdeg}(\beta,N') \\
&\leq \frac{30}{11}\cdot\operatorname{avgdeg}(\beta,N')
\end{aligned}$$

Within the calculation we use the following inequalities:

$$
\begin{aligned}
1.\quad -N^{\frac{1-\beta}{\beta}} + \sum\nolimits_{i=2}^{N^{\frac{1}{\beta}}} i^{-\beta} &\geq -N^{\frac{1-\beta}{\beta}} + 2^{-\beta} + 3^{-\beta} + \frac{1}{2}(4^{-\beta} - 5^{-\beta}) + \int_4^{N^{\frac{1}{\beta}}} i^{-\beta} di \\
&\geq \left(\frac{1}{4}5^{-\beta} - N^{\frac{1-\beta}{\beta}}\right) + 2^{-\beta} + 3^{-\beta} + \int_4^{\infty} i^{-\beta} di + \int_{N^{\frac{1}{\beta}}}^{\infty} i^{-\beta} di \\
&\geq \left(\frac{1}{4}5^{-\beta} - \frac{\beta}{\beta-1} N^{\frac{1-\beta}{\beta}}\right) + 2^{-\beta} + 3^{-\beta} + \int_4^{\infty} i^{-\beta} di \\
&\geq 2^{-\beta} + 3^{-\beta} + \int_4^{\infty} i^{-\beta} di,
\end{aligned}
$$

$$
\begin{aligned}
2.\quad \lfloor N' \cdot i^{-\beta} \rfloor &\geq N' \cdot i^{-\beta} - 1 \\
&= \frac{N'}{N}\left(N \cdot i^{-\beta} - \frac{N}{N'}\right) \\
&\geq \frac{N'}{N}\left(\lfloor N \cdot i^{-\beta} \rfloor - 1\right)
\end{aligned}
$$

$$
\begin{aligned}
3.\quad \lfloor N \cdot i^{-\beta} \rfloor &\geq N \cdot i^{-\beta} - 1 \\
&= \frac{N}{N'}\left(N' \cdot i^{-\beta} - \frac{N'}{N}\right) \\
&\geq \frac{N}{N'}\left(\lfloor N' \cdot i^{-\beta} \rfloor - \frac{N'}{N}\right)
\end{aligned}
$$

$$
4.\quad \sum\nolimits_{i=2}^{N^{\frac{1}{\beta}}} i^{1-\beta} \leq 2^{1-\beta} + 3^{-\beta} + \int_3^{\infty} i^{1-\beta} di
$$

Finally we use that function $f_N$ with

$$
f_N : \beta \mapsto \left(1 - \frac{1 - N^{\frac{2-\beta}{\beta}}}{1 - N^{\frac{2-\beta}{\beta}} + 2^{1-\beta} + 3^{1-\beta} + \frac{1}{\beta-2}3^{2-\beta}}\right) \cdot \left(1 + \frac{1 + 2N^{\frac{1-\beta}{\beta}}}{-N^{\frac{1-\beta}{\beta}} + 2^{-\beta} + 3^{-\beta} + \frac{1}{\beta-1}4^{1-\beta}}\right)
$$

is monotone declining for $\beta > 2$ and that the limit for $\beta = 2$ (i.e, $\lim_{\varepsilon \to 0} f_N(2+\varepsilon)$ ) can be bounded by $\frac{30}{11}$ for the given value of $N$. □

Later in this thesis, we use some slightly weaker form of the above lemma. Namely, we use $\delta$, which is defined to be the maximum average degree of all $\beta$-PL graphs, for some fixed $\beta > 2$, instead of the average degree of a $\beta$-PL graph with $N' \geq N$. Since the average degree increases with larger number of vertices the resulting threshold still holds. Doing so, we probably get some threshold that is slightly larger than required due to the structure within the proof. However, we achieve to state our result independent of the size of the graph.

#### 2.2.3.2 Construction of subgraphs

After analyzing the degree subsequences, we state how to construct graphs for some specific type of candidate degree sequence $S$ (see Lemma 2.3). We describe how to build a graphs $G$ with degree sequence $S$ in such a way that any subgraph of $G$ has a bounded average degree.

**Lemma 2.3** *Let $\delta \in \mathbb{R}^+$, and let $S$ be some candidate degree sequence on $n$ vertices with maximum degree $d$. If $S$ has the following properties*

*1.* $s_1 > \binom{\lfloor \delta+1 \rfloor}{2}$

*2.* $s_i > 0$ *for all* $i \in [\,2 \,..\, \lfloor \frac{\delta}{2} \rfloor\,]$

*3.* $s_i > 0$ *for all* $i \in [\,d - \lfloor \frac{\delta}{2} \rfloor + 1 \,..\, d\,]$

*4.* $\sum\limits_{i=1}^{\lfloor \delta/2 \rfloor} i \cdot s_i \; - \; \binom{\lfloor \delta \rfloor + 1}{2} \; \geq \; \sum\limits_{i=\lfloor \delta+1 \rfloor}^{d} i \cdot s_i$

*then it is possible to build a graph $G = (V, E)$ with degree sequence $S$, such that every subgraph has (induced) average degree at most $\delta$.*

*Proof:* In order to prove the lemma we state an explicit construction of $G$. Let $n$ be the number of vertices in $S$. We start with $n$ isolated vertices each assigned some degree with respect to $S$. Then, we successively add the required number of edges in such a way that in the end every vertex has the degree is has been assigned to.

First of all, we partition the isolated vertices into the following sets:

$A$ The set of vertices with assigned degree at most $\delta/2$.

$B$ The set of all vertices with assigned degree greater than $\delta$

$C$ The set of all remaining vertices (i.e., assigned degree $i$ with $\delta/2 < i \leq \delta$)

The edges are inserted as follows:

1. First of all, the vertices in $B$ are assigned their edges. In order to do so, we add edges from vertices in $A$ to vertices in $B$. Based on properties 2 and 3 we can show that it is possible to satisfy all degrees of the vertices in $A$. We proceed as follows.

   Throughout the whole construction process we define an continously updated order $S_b = [v_i]_{1 \leq i \leq |B|}$ on the vertices in $B$ in such a way that, for every $i \in [\,2 \,..\, |B|\,]$, it always holds $\deg^*(v_{i-1}) \leq \deg^*(v_i)$, where $\deg^*(v)$ is the missing number of edges of $v$ (i.e., the difference of the assigned degree and the actual degree of $v$).

Starting with the vertices with highest degree $x$ in $A$ we assign $x$ edges to the vertices $v_{|B|-x+1}, \ldots, v_{|B|}$. Due to properties 2–4, we can iterate this process until all demands of vertices in $B$ are satisfied in such a way that no vertices in are linking twice and that there remain at least $\binom{\lfloor \delta+1 \rfloor}{2}$ unlinked vertices of degree 1 in set $A$.

Finally, all vertices $v \in A$ with $\deg^*(v) > 0$ are moved to set $C$.

2. Now, we add edges to satisfy all remaining open connections of vertices in $C$. Iteratively, we choose all vertices with maximum value $x$ of $\deg^*$ in $C$. Using these vertices we build as many cliques of size $x+1$ as possible. For all remaining vertices we satisfy one of their links with a vertex of degree one (i.e., we reduce $x$ by one). Iterating this process at most $x-1$ times end up with vertices that have at most one unsatisfied connection. In very step in the above iteration we need at most $x$ vertices of degree one. Thus, using property 1 we know that there exists enough vertices of degree one satisfy these links. Further, since $S$ is a candidate degree sequence the number of the finally remaining vertices with one unsatisfied connection must be even. Therefore, we can inter-connect the remaining vertices pairwise.

After satisfying all links of the degree sequence $S$, we can determine the maximum average degree of all subgraph of the resulting graph $G$. For any subgraph $\tilde{G}$ of $G$, we partition the vertices as follows:

- set $S_1$ contains all vertices with (induced) degree greater than $\delta$;
- set $S_2$ contains all vertices in $V(\tilde{G}) \setminus S_1$ that are connect to vertices in $S_1$
- set $S_3$ contains all vertices in $V(\tilde{G}) \setminus (S_1 \cup S_2)$.

Further, define $n_i = |S_i|$ and $d_i = \sum_{v \in S_i} \deg_{\tilde{G}}(v)$, where $\deg_{\tilde{G}}(v)$ denotes the degree of vertex $v$ in graph $\tilde{G}$. Due to the construction the vertices in $S_1$ (degree greater than $\delta$) are only connected to vertices in $S_2$ (which have degree at most $\delta/2$). Therefore, we know $n_2 \geq \frac{d_1}{\delta/2}$ what is equivalent to $d_1 \leq \frac{\delta}{2} n_2$. Thus, the average degree to $\tilde{G}$ can be bounded as follows:

$$\operatorname{avgdeg}(\tilde{G}) = \frac{d_1 + d_2 + d_3}{n_1 + n_2 + n_3} \leq \frac{\frac{\delta}{2} n_2 + \frac{\delta}{2} n_2 + \delta n_3}{n_2 + n_3} \leq \delta$$

This inequality holds for all subgraphs of $G$ and concludes the proof. □

Using the above lemma can be state the following result on power-law graphs. Given some appropriate degree subsequences of a $\beta$-PL graphs. Lemma 2.4 guarantees the existence of a corresponding graph in such a way that the average degree of all subgraphs is bounded by some constant, which only depends on $\beta$.

**Lemma 2.4** *Let $\beta > 2$, $\delta_{\max} = 4 \cdot 2^{\frac{1}{\beta-2}} + 1$, and let $S$ be a candidate degree sequence with maximum degree $d$, that is a degree subsequence of a $(N, \beta)$-PL. If the following conditions hold*

1. $s_i \geq \lfloor Ni^{\beta} \rfloor$ *for all* $i \in [\,1 \;..\; \lfloor \frac{\delta_{\max}}{2} \rfloor\,]$
2. $s_i \leq \lfloor Ni^{\beta} \rfloor$ *for all* $i \in [\,\delta_{\max} \;..\; d\,]$
3. $0 < s_i$ *for all* $i \in [\,d - \frac{\delta_{\max}}{2} + 1 \;..\; d\,]$

*it is possible to build a graph $G = (V, E)$ with degree sequence $S$, such that every subgraph of $G$ has average degree at most $\delta_{\max}$.*

*Proof:* It is sufficient to consider only values $N \geq (\delta_{\max} + 1)^{\beta}$ in detail. For all other values, the maximum degree of all $(N, \beta)$-PL graphs is at most $\delta_{\max}$. Therefore, this value is a trivial upper bound for any subgraph of $G$. The existence of $G$ can easily be seen from condition 1.

The proof for values $N \geq (\delta_{\max} + 1)^{\beta}$ is done by applying Lemma 2.3, using $\delta = \delta_{\max}$. Due to the power-law nature of the degree sequence and the large value of $N$ and the above conditions the conditions 1–3 of Lemma 2.3 can be easily seen to be true. Using the definitions of $\delta$ and $N$ as states above, the last conditions of the lemma evaluates to:

$$\sum_{i=1}^{\lfloor \delta_{\max}/2 \rfloor} i \cdot \lfloor Ni^{-\beta} \rfloor \;-\; \binom{\lfloor \delta_{\max} \rfloor + 1}{2} \geq \sum_{i=\lfloor \delta_{\max}+1 \rfloor}^{d} i \cdot \lfloor Ni^{-\beta} \rfloor .$$

Due to $\beta > 2$ we get $N > (\delta_{\max} + 1)^2$ and therefore it is sufficient to show

$$\sum_{i=2}^{\lfloor \delta_{\max}/2 \rfloor} Ni^{1-\beta} \geq \sum_{i=\lfloor \delta_{\max}+1 \rfloor}^{d} Ni^{1-\beta}.$$

The latter inequality holds if

$$\int_{2}^{\frac{\delta_{\max}-1}{2}} i^{1-\beta} di \geq \int_{\delta_{\max}-1}^{\infty} i^{1-\beta} di$$

Using the definition $\delta_{\max} = 4 \cdot 2^{\frac{1}{\beta-2}} + 1$, we can verify this inequality as follows:

$$\begin{aligned}
\int_{2}^{\frac{\delta_{\max}-1}{2}} i^{1-\beta} di &= \frac{1}{\beta-2}\left(2^{2-\beta} - 2^{\frac{\beta-1}{\beta-2}(2-\beta)}\right) = \frac{1}{\beta-2}\left(2^{2-\beta} - 2^{1-\beta}\right) \\
&= \frac{1}{\beta-2} 2^{1-\beta} \geq \frac{1}{\beta-2} 2^{3-2\beta} \\
&= \frac{1}{\beta-2}\left(2^{\frac{1}{\beta-2}+2}\right)^{2-\beta} = \int_{\delta_{\max}-1}^{\infty} i^{1-\beta}.
\end{aligned}$$

After satisfying all required conditions, we can apply Lemma 2.3 and build a graph with degree sequence $S$ in such a way that all subgraphs have average degree at most $\delta_{\max}$. □

### 2.2.4 Average-case analysis for power-law graphs

At the end of the discussion of power-law graphs, we briefly want to explain the advantages that could be taken of developing random algorithms on general power-law graphs.

In the past, the introduction of the general random graph model $G_{n,p}$ has initiated research of computational complexity for input chosen randomly from the corresponding sets of graphs. Based on the independent occurrence of edges, it has been possible to develop algorithms that perform well on average. I.e., despite of computational hardness or poor approximation results one some specific input instances, it has been possible to prove good results, on the huge majority of possible inputs. For a wide range of graph problems, e.g., independence-number and coloring [KV02], perfect-matching [DFP93], edge connectivity [YDL94], or restricted bisection [DDSW03] there exists good results for the average-case analysis on random graphs.

Consequently, analogous to the research on the $G_{n,p}$ model, we might get similar good results when assuming random power-law graphs. In this section, we have shown that for some of the models there exist results on the diameter, the cluster coefficient, and the size of neighborhoods (see subsection 2.2.1). Similarly to the results on the robustness and vulnerability of this type of networks (also discussed in 2.2.1), it seams very promising to derive good average-case complexity for other problems w.r.t. nowadays large-scale networks. We even can hope to reapply some of the techniques that have been used, when analyzing the models of general random graphs. Additionally, we can use properties of $\beta$-PL graphs (e.g.,the large number of vertices of small degrees and rare but prominent number of vertices with high degrees). Combining these techniques and observed properties, it should be also possible to prove good results for the average-case analysis on power-law graphs.

## 2.3 The hyperlink graph of the WWW

In this section, we present some network specific properties the hyperlink structure of the WWW. This structure is represented by the hyperlink graph, i.e. the webpages correspond to vertices, while the hyperlinks between two webpages are modeled as directed edges connecting corresponding vertices.

In the previous part of this chapter, we have stated that this huge graph (with billions of vertices), either considered in its directed or undirected version, follows the power-law graph behavior (with power-law exponent $\beta \approx 2.1$ [New03]). Further, due to its sparsity, the proposed small diameter [ABJ99] (which is known not to be true, in the case, see below), and the appearance of highly interlinked clusters (representing webpages concerning some same topic) the hyperlink structure is often classified as a small-world graph (e.g., see [Adm99]).

In the following, first of all, we discuss the structure of the directed hyperlink graph in more detail and show that, in general, it does not belong to the class of small-world graphs (due to its large diameter). Second, we summarize results on the communities and explain some general property of the clusters that is widely used to determine relevant pages within the communities (e.g., heuristics for determining dense subgraphs, see section 5.3).

### 2.3.1 Structure of the hyperlink graph

While the undirected version of the hyperlink graph appears to be a small-world graph, this classification is incorrect for its directed version. Broder et al. have investigated hyperlink graphs that result from two AltaVista crawls each with over 200 million pages and 1.5 billion links [BRM$^+$00, KRR$^+$00b]. The observed macrostructure of the hyperlink graph is illustrated in Figure 2.3 and can be described as follows, often referred to as the bow-tie structure of the WWW.

- 90% of the pages are connected and form a giant component. These vertices can be categorized in four sets of roughly the same size:
  - The set SCC builds a strong connected component.
  - The set IN contains all vertices that can reach the vertices in SCC via directed paths, but are not in SCC themselves.
  - The set OUT contains all vertices that can be reached from vertices in SCC via directed paths, but are not in SCC themselves.
  - The remaining set of vertices are referred to as tendrils (i.e. vertices not in SCC but can be either reached from vertices in IN, or that can reach vertices in OUT) or tubes (vertices on paths from IN to OUT using no vertices in SCC).
- 10% of the pages build small disconnected components

Further, Broder et al. investigated the length of a path between two randomly chosen vertices (using several random BFS runs on the hyperlink graph). First of all, in the directed case, it is easy to see that such paths exist for only approx. 25% of all possible pairs. Based on these path, the following results have been derived. The diameter of SCC is at least 28. The maximum finite existing shortest path is at least 503. However, its length is likely to be close to 900, assuming that no short tubes connect the most distant page from IN to the most distance page in OUT. Finally, using the average connected distance from the BFS runs (either in the the original direction of the edges (referred to as out-links), the inverse direction (referred to as in-links), or either direction (referred to as undirected) the following average distances (assuming that the vertices are connected) have been derived:

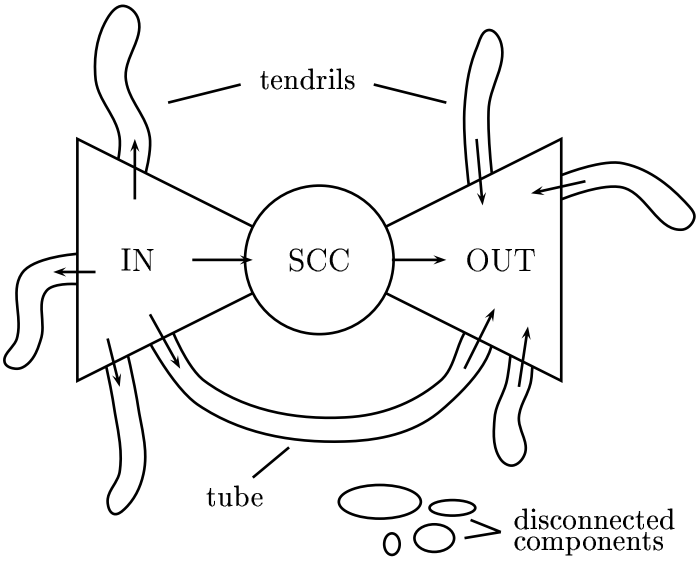

Figure 2.3: The structure of the directed hyperlink graph [BRM+00]

| | in-links | out-links | undirected |
|---|---|---|---|
| avg. distance | 16.12 | 16.18 | 6.83 |

These results are interestingly contradicting the average distance of 19 predicted by Albert, Jeong and Barabasi [ABJ99] on a smaller set of vertices.

All in all, we may conclude that the hyperlink graph is not connected and thus has infinite diameter. Even when restricting to those pairs of vertices that are connected, we observe a large value for the maximum shortest path. Thus, the hyperlink graph is no small-world graph, in general. However, when weaken the condition for the diameter to the average shortest path length (if existent), or even considering the undirected graph, we observe the required small distance between two vertices.

### 2.3.2 Communities in the hyperlink graph

After analyzing the distance of vertices, we proceed to discuss the structure of communities. Communities are characterized by sets of highly interlinked pages (dense subgraphs) that share some common interest (same topic). In the following, different to small-world graphs, where we have analyzed the clustering coefficient, we focus on specific subgraphs, namely dense bipartite graphs. There are several results on the existence of dense bipartite graphs [KPRT99b, Kle99, IMK+03] within the subgraphs that correspond to communities.

Let $C_{i,j}$ be a complete bipartite graph on two vertex sets of size $i$ and $j$. The number of expected $C_{i,j}$'s in a random graph $G_{n,p}$ is $\binom{n}{i}\binom{n}{j}p^{i.j}$. Therefore, using $p$ proportional to $\frac{1}{n}$ (i.e., guaranteeing constant average degree), this number is negligible for $ij > i+j$. However, Kumar et al. observed that the number of $C_{i,j}$ in the hyperlink graph is of relevant size and increases for larger $n$ [KPRT99b]. Similar results hold, when restricting to dense bipartite graphs. Based on the motivation of the copying model for $\beta$-PL graphs (also proposed by Kumar et al., see paragraph 2.2.2.2), it is possible to explain this large number of occurring subgraphs.

Similarly, Kumar et al. have observed that the average distance of two vertices within a community can be measured very well in term of a so called alternating connectivity [KKR$^{+}$99]. When using alternating connectivity, we do not count the length of directed path but require that every second edges is traversed in inverse direction (i.e., within complete directed bipartite graphs all vertices have alternating connectivity of at most two). Similarly, Kumar et al. have observed that, assuming some definition of clusters that is based on the HITS algorithm (see below), the average alternating distance of two vertices within the same cluster is at most twice the distance within the corresponding undirected graph. Thus, the alternating distance is significantly smaller than the distance based on directed paths.

One of the most prominent results that is based on the concept of bipartite subgraphs is the previously mentioned HITS algorithm of Kleinberg [GKR98, Kle99]. This algorithm is used to determine good authorities for one or more query terms, i.e. webpages that contain good information on these terms (where good is some measurement, based on endorsement of creators of webpages). The search algorithm consists of two main steps.

- First of all, using the set of webpages that is returned, when applying the query to a standard index-based search engine, a root set of vertices is created (e.g., choosing the top 200 entries in the answer of the search engine). This root set is enlarged by adding all webpages that either link to the root set, or are linked from pages contained in the root set. The resulting set is referred to as the base set.
- Within the second step of the HITS algorithm, for each page $i$ within the base set, we determine an authority value $x_i$ and a hub value $y_i$. These values provide some final ranking on the vertices. The authority value describes the quality of information of the page, concerning the query terms, while the hub value judges whether the page links to pages that are good authority pages. Both values are determined by using the following iterative weight-propagation procedure (initially all weights are considered to be equal). The new hub weight of a page is determined by the sum of the authority weights the pages points to. Similarly, the new authority weight equals to the sum of the hub weights of the pages it is referred from:

$$x_p = \sum_{q \to p} y_p \qquad\qquad y_p = \sum_{p \to q} x_q$$

> Using the adjacency matrix of the hyperlink graph we can expand this definition and derive $x \leftarrow A^T y$ and $y \leftarrow Ax$. Applying one more expansion step we get
>
> $$x \leftarrow (A^T A)x \qquad\qquad y \leftarrow (AA^T)y.$$
>
> Therefore, both vectors converge to the principle eigenvectors of the matrices $A^T A$ and $AA^T$, respectively (if the initial weights are chosen appropriately, e.g. all positive). It is possible to use the power-iteration technique to approximate the resulting vectors (for more information on eigenvectors and power-iteration see, e.g., Golub and van Loan [GVL89]).

Finally, the outcome of the HITS algorithm is defined to consist of those pages from the base set that have the highest authority weight. Similar to the HITS algorithm, other search technique are also based on similar endorsement policies (i.e., an link to a page increases its overall weight). As an example, Brin and Page evaluate some PageRank that, based on some random walk on the hyperlink graph, determines some overall quality of the page rank independent of the search query [BP98].

### 2.3.3 Summary for the hyperlink graph

In this section we have described some properties of the hyperlink graphs. Besides the bow-tie structure we have discussed that communities may be characterized by dense bipartite subgraphs. These properties, which provide some good description of the underlying structure of the hyperlink graph, are used within several approximations and heuristics (e.g., search engines, community detection, and information bases).

We have chosen the hyperlink graph of the WWW as an example to illustrate that the power-law structure of the degree sequence is only a very abstract way to describe large-scale networks and cannot cover network-typical properties. Therefore, we have always to keep in mind whether we want to state some general result on power-law graphs (see the main results within this thesis) or if we want to get optimized algorithms for some problem for some specific type of this class of graphs (see section 5.3).

## 2.4 Summary

In this chapter, we have given a general discussion on large-scale networks. Doing so, we have described the small-world characteristic that occurs in most of these networks. Besides sparsity and short paths we have seen that this characteristic also indicates the existence of high cluster tendency. According to this observation, we can expect good results, when applying density-based clustering to this class of networks. In order to describe the overall set of networks, we have discussed the power-law degree distribution of the underlying graphs. Based on the extensive study of this characteristic within the literature, we have given an overview of properties and models for this class of graphs. Further, we have seen that the power-law exponent for most of these graphs is chosen from the interval $]2 .. 3]$. Using these results we can abstract the family of large-scale networks using $\beta$-PL graphs, with $\beta > 2$. This abstraction is used within the following chapters, when discussing the complexity of detecting dense subgraphs.

At this point we want to stress the observation that the class of power-law graphs is a very general abstraction of large-scale networks. As demonstrated in the previous section, when discussion the hyper-link graph, we have seen that besides this structural property there exist additional characteristics specific to single subclasses of these networks. Especially when considering approximation results for density-based clustering, we require these additional properties in order to derive good results (see chapter 5). Therefore, $\beta$-PL graphs have to be understood to as an abstraction for the whole family of large-scale networks but not as a model for a specific subtype of these structures.

# Chapter 3

# Graph transformations

In this chapter, we present several graph transformations which are used to derive a reduction from CLIQUE to $\gamma$-DENSE-SUBGRAPH-PROBLEM (denoted by $\gamma$-DSP), see sections 4.3 and 4.4. Within this reduction, we convert, for a wide range of functions $\gamma$, instances $(G, k)$ of the CLIQUE problem to instances $(G', k')$ of $\gamma$-DSP. The overall idea of the reduction is that the input graph $G$ contains a clique of size $k$ if and only if the transformed graph $G'$ contains a subgraph on $k'$ vertices with density (i.e., number of edges) at least $\gamma(k')$. In general, we build $G'$ by applying iteratively four elementary transformations, which are described below. To make the reduction work, we show for each transformation and for sequential combinations that the input contains some specific type of subgraph, characterized by a certain number of vertices and edges, if and only if the resulting graph contains a corresponding subgraph. Having done this, we can reason that the existence of a clique of size $k$ in $G$ is equivalent to the existence of a corresponding $\gamma$-dense subgraph in $G'$.

In the following, we introduce the four kinds of transformations. A principle characteristic of some of these transformations is their locality, i.e., each vertex or edge of the graphs is replaced simultaneously without considering the rest of the graph. In order to generalize this property, we define and discuss a suitable family of graph transformations (referred to as general local graph transformations). Finally, we demonstrate how to combine the introduced transformations to achieve the properties, stated above. The terms transformation, operation, and operator are used interchangeably.

## 3.1 Elementary graph transformations

In this section, each of the four above mentioned transformations is presented and analyzed separately. While the first two of them are used to increase resp. decrease the density of the resulting subgraph, the latter ones are used to assure some minimum degree or some precise number of edges, respectively.

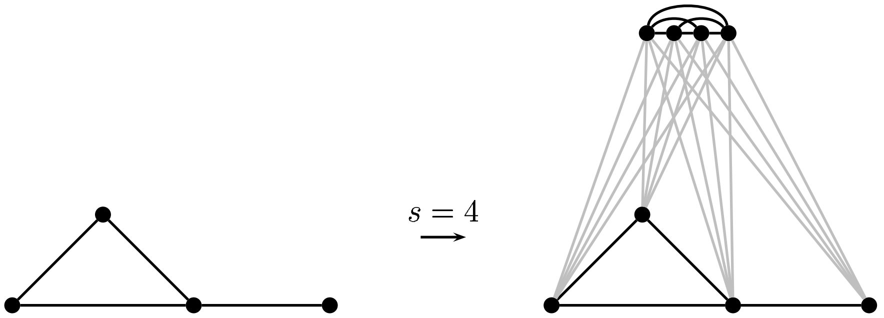

Figure 3.1: The densification transformation $R_s$

### 3.1.1 The Transformation $R_s$

Let $G$ be any undirected graph. The idea of the operator $R$ is to modify the graph in such a way that the density of the graph (and its subgraphs) increases. The relative density (i.e, the fraction of existing edges w.r.t. the possible number of edges on the considered number of vertices) also increases, unless $G$ itself is a clique. Due to this property, $R$ can be interpreted as a *densification transformation.*

#### 3.1.1.1 Definition of $R_s$

The transformation $R_s$ is defined using the following sequence $(G_j)_{0 \leq j \leq s}$ of graphs. Let $G_0 = G$ and $G_j = h(G_{j-1})$, for $j > 0$, where $h$ transforms an input graph $H$ by adding a new vertex that is connected to all other vertices in $H$. Finally, let $R_s(G) = G_s$. For $s = 4$, the transformation is illustrated in Figure 3.1. Obviously, the following property holds (the inductive proof is not stated):

$$G \text{ has a clique of size } k \iff R_s(G) \text{ has a clique of size } k + s \tag{3.1}$$

#### 3.1.1.2 Inversion of $R_s$

After defining the transformation itself, we consider the possibility to inverse the operation. Given a graph $G' = R_s(G)$, we would like to construct a graph $\tilde{G}$ isomorphic to $G$, i.e., $\tilde{G} \cong G$. There are two possible kinds of inversion problems, depending on the supplied parameters:

1. *Both $G'$ and $s$ are given as input*: We can simply remove from $G'$ $s$ vertices that have degree $|V(G')| - 1$.
   If there are more than $s$ vertices having that degree, we can remove any s of them (all such vertices are topologically equivalent, and so are the resulting graphs). Since the original graph $G$ is one of the possible outcomes, any resulting graph is a correct solution.

2. *Only $G'$ is given as input*: Without the parameter $s$, we are, in general, not able to reconstruct a graph isomorphic to $G$. Let $x$ be the number of vertices in $G'$ with degree $|V(G')| - 1$. Obviously the largest possible value of $s$ is equal to $x$. Further, any value in $[\,0\,..\,x\,]$ is an admissible choice for $s$. Therefore, as above, we can construct a set of $x + 1$ graphs that can be transformed into $G'$ by choosing the parameter $s$ appropriately.
   There are two settings where $s$ can be reconstructed exactly and thus, a unique (up to isomorphism) solution can be found. Firstly, if there exists no vertex in $G'$ with degree $|V(G')| - 1$ then $s = 0$ must hold. Secondly, if we know that $G$ contains no vertex with degree $|V(G)| - 1$ then $s$ must equal $x$.

#### 3.1.1.3 Decision problem Clique$_\delta$

In addition to increase the density, $R$ can be used to define a family of special **NP**-complete versions of the Clique problem. In these new problems the sizes of the desired clique is related to the graph size. Such restrictions are well-known in the area of **NP**-complete problems (e.g., Asahiro et al. use a version where the size of the graph is exactly three times the demanded clique size [AHI02]).

The operator $R$ can be used to bound the size of the graph from above. First of all, define Clique$_\delta$ to be the set of all tuples $(G, k)$ such that graph $G$ has a clique of size at least $k$ and $|V(G)| \leq (1+\delta)k$. Clique can in a straight forward manner be reduced to Clique$_\delta$, for any fixed $\delta$, with $0 < \delta \in \mathbb{Q}$. To do so, let the tuple $(G, k)$ be an input to the Clique problem.

- $|V(G)| \leq (1 + \delta)k$: Obviously, $(G, k)$ is an input of the Clique$_\delta$ problem, thus no transformation is required.

- $|V(G)| > (1+\delta)k$: We can apply $R_s$ with $s = \left\lceil \frac{1}{\delta} \right\rceil |V(G)| - \left\lfloor 1 + \frac{1}{\delta} \right\rfloor k$. Due to the size restriction on $|V(G)|$ it follows directly that $s \geq 0$. The transformed graph $G_s$ has $|V(G)| + s \leq \delta \left(s + \left(1 + \frac{1}{\delta}\right) k\right) + s = (1 + \delta)(k + s)$ vertices. Furthermore, proposition (3.1) implies that $G_s$ has a clique of size $k + s$ if and only if $G$ has a clique of size $k$. Thus $(G_s, k + s)$ is the corresponding input to the Clique$_\delta$ problem.

Later in this thesis, we consider the case $\delta = \frac{1}{2}$ and reduce Clique$_{\frac{1}{2}}$ in order to show **NP**-completeness of $\gamma$-DSP.

Similarly, given some clique size, we can bound the size of the graph from below by adding isolated vertices. Doing so, the size of the graph increases. A combination of both techniques can be used to adjust the values in more sophisticated ways, e.g. as required in [AHI02].

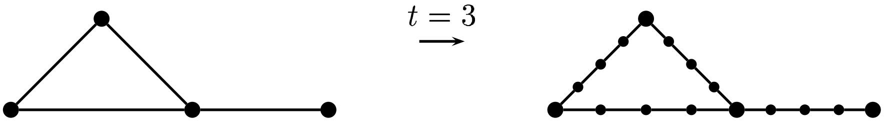

Figure 3.2: The sparsification operator $S_t$.

### 3.1.2 The Transformation $S_t$

In contrast to the previous section, where we described how to increase the density of a graph, we now state a method to decrease its density. This decrease takes place in such a way that the density of all subgraphs of the input is reduced simultaneously. The corresponding transformation $S_t$ is called *sparsification transformation.*

#### 3.1.2.1 Definition of $S_t$

In order to assure that the density of all subgraphs is reduced, we replace all edges by chains. Formally, we can define the transformation as follows. Let $G$ be any undirected graph. Applying $S_t$ to $G$ results in a graph $G_t$ that is a copy of $G$, where every edge $e = \{v_1, v_2\}$ is replaced by a path $p = (v_1, w_{e,1}, \dots, w_{e,t}, v_2)$ of length $t+1$ (number of edges) involving $t$ new vertices $w_{e,1}, \dots, w_{e,t}$. The new vertices are referred to as *inner vertices* and the other ones as *outer vertices.* Inner vertices always have degree 2, while outer ones have the same degree as the corresponding vertex in $G$. According to the construction, $G_t$ has $|V(G)|+t|E(G)|$ vertices and $(t+1)|E(G)|$ edges. For $t = 3$ the transformation is illustrated in Figure 3.2.

Similar to the operator $R$, an occurrence of a clique in $G$ corresponds to the occurrence of some corresponding subgraph in $G_t$, and vice versa.

$$\begin{aligned} G \text{ has a clique of size } k \quad \iff \quad & S_t(G) \text{ has a subgraph on } k + t\binom{k}{2} \text{ vertices} \\ & \text{and } (t+1)\binom{k}{2} \text{ edges} \end{aligned}$$

While the implication from left to right is straightforward, the other direction has to be proven in detail. For the proof we refer the reader to the proof of Lemma 3.2, where we are going to show a more general proposition. Similar constructions have been used in the literature for the complexity analysis of this and similar problems [FK94, FS97, GNY94].

#### 3.1.2.2 Inversion of $S_t$

For the inversion of $S$ we derive results analogous to those in the previous subsection. Let $G_t = S_t(G)$. Once again, only if both $G_t$ and $t$ are supplied, it is possible to uniquely reconstruct a graph isomorphic to $G$. If the value of $t$ is not given, it is, in general, possible to compute several graphs that can be transformed to $G_t$ by applying the transformation $S$ with some appropriate parameter $\tilde{t}$ (see below).

For $t = 0$ we have that $S_t(G) \cong G$. Therefore, in the following, we only consider choices $t \geq 1$. The inversion process is based on the property that all inner vertices have degree two and every edge in the original graph $G$ has been replaced by a chain of length $t+1$. Let $X$ be the set of all vertices of $G_t$ with degree unequal two. Obviously, these vertices must be outer vertices. Consider the subgraph $G_{\bar{X}}$ of $G_t$ induced by the vertices in $\bar{X} = V(G_t) \setminus X$. Its connected components are either circles or chains:

- Each circle $c$ in $G_{\bar{X}}$ has length $i_c \cdot (t+1)$, for some $i_c \in \mathbb{N}$ and $i_c \geq 2$, and it corresponds to an isolated circle in $G$ of length $i_c$.
- Each chain $p$ in $G_{\bar{X}}$ has length $i_p \cdot (t+1) - 2$, with $i_p \in \mathbb{N}$ and $i_p \geq 1$, and corresponds to a chain in $G$ of length $i_p$ connecting two outer vertices in $X$. All outer vertices on such chains can be determined easily. Starting at one end exactly every $t+1$-sth vertex is an outer vertex, with degree two.

Thus, if we know the value of $t$ we can build a graph isomorphic to $G$. Otherwise, if we have no information on $t$, it is possible that there exist different values $\tilde{t}$, for which we can find some graph $\tilde{G}$ such that $S_{\tilde{t}}(\tilde{G}) \cong G_t$. All possible values of $\tilde{t}$ can be determined as follows. Let $Y_{\text{circle}}$ be the set of sizes (i.e., number of vertices) of those connected components of $G_{\bar{X}}$ that form circles, and let $Y_{\text{chain}}$ be the corresponding set of chain lengths. Further we define $y_{\min} = \min\{Y_{\text{circle}} \cup Y_{\text{chain}}\}$. For all admissible values $\tilde{t}$ it must hold

$$\begin{array}{llll}
\tilde{t} \in [\,1 \,..\, y_{\min}\,] & & & \\
\wedge \quad (\,\forall\, y_{\text{circle}} \in Y_{\text{circle}} & \exists\, i \in \mathbb{N},\ i \geq 2\,) & [\, y_{\text{circle}} & = \ i \cdot (\tilde{t}+1) \qquad ] \\
\wedge \quad (\,\forall\, y_{\text{chain}} \in Y_{\text{chain}} & \exists\, i \in \mathbb{N},\ i \geq 1\,) & [\, y_{\text{chain}} & = \ i \cdot (\tilde{t}+1) - 2\,]
\end{array}$$

Note, if $S_t$ was applied to some $G$, using some value $t$ which is greater than two and odd, unique reconstruction is not possible. Obviously, in such a case, $t = 2 \cdot i + 1$, with $i \in \mathbb{N}$ and $i \geq 1$, and furthermore $S_i(S_1(G)) = G_t$. Thus, both $G$ and $S_1(G)$ can be transformed into $G_t$ using operator $S$.

### 3.1.3 The Transformation $N_x$

The transformation $N$, which we introduce in this subsection, can be used to increase the minimum degree of the graph while keeping the structure generated

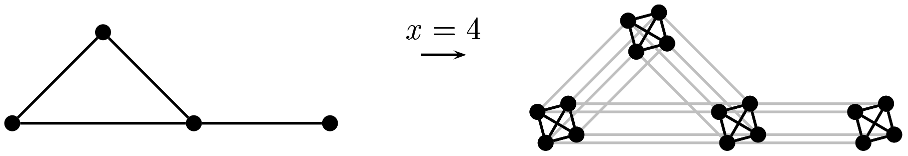

Figure 3.3: The degree-bounding transformation $N_x$.

by other operators, such as $R$ and $S$. This property is used when reducing CLIQUE to $\gamma$-DSP in power-law graphs. The transformation is called *degree-bounding transformation.*

#### 3.1.3.1 Definition of $N_x$

To guarantee that the topology of the transformations $R$ and $S$ are not disturbed, the general idea is to replace vertices instead of edges. Informally speaking, every vertex is replaced by a clique, while the edges are expanded to complete matchings connecting those cliques.

Let $G = (V, E)$ be an undirected graph, and let $x \in \mathbb{N}$, with $x \geq 1$. $N_x(G) = G_x$ is defined formally by the following sets of vertices and edges:

$$\begin{array}{rclcl} V(G_x) & = & \{ \quad (v, i) & \mid & v \in V,\ 1 \leq i \leq x \ \} \\ E(G_x) & = & \{ \ \{(v, i), (w, j)\} & \mid & (v = w) \ or \ (\{v, w\} \in E \ \wedge \ i = j) \ \} \end{array}$$

It is easy to see that $G_x$ has $x \cdot |V(G)|$ vertices and $\binom{x}{2} \cdot |V(G)| + x \cdot |E(G)|$ edges. In Figure 3.3, we outline the transformation for $x = 4$.

In the literature, there exists a wide range of names for this transformation. The notion of the Cartesian Graph Product [Viz63] of $G$ with $K_x$ is the most commonly used. The same operation sometimes is also referred to as sum [Ber58]. Further there exist several generalizations for this family of transformations, e.g., $p$-sum [Ber58] and NEPS (Non-complete Extended P-Sum) [CL70].

Alternatively, we can interpret the operation $N_x$ as build $x$ copies of the input graph and completely connecting corresponding vertices. Therefore, we easily observe that for any subgraph in $G$ there exists an isomorphic subgraph in $N_x(G)$. Thus, unlike for $R$ and $S$, when applying $N$ to some input, it is not possible to set up a similar equivalence between the existence of a clique on the one hand and the existence of some subgraph of appropriate size and density on the other.

However, when $S$ is followed[1] by $N$, we can prove the following result (see Lemma 3.2 for a complete proof).

[1] Let $f$ and $g$ be two transformations. We use $f \circ g$, to express that $g$ is followed by $f$, i.e., $(f \circ g)(x) = f(g(x))$.

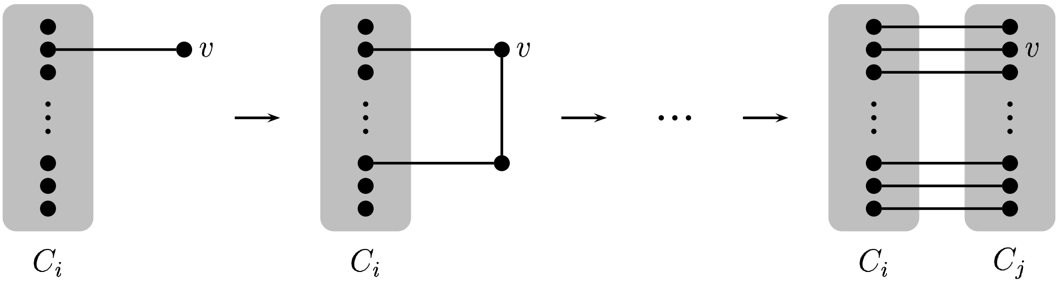

Figure 3.4: Reconstruction of a neighboring clique in $N_x(G)$

$$G \text{ has a clique of size } k \iff \begin{array}{l} (N_x \circ S_t)(G), \text{ with } x, t \geq 1 \text{ has a subgraph on} \\ x \cdot \left(k + t\binom{k}{2}\right) \text{ vertices and} \\ \binom{x}{2} \cdot \left(k + t\binom{k}{2}\right) + x(t+1)\binom{k}{2} \text{ edges} \end{array}$$

#### 3.1.3.2 Inversion of $N_x$

Similar to the inversion problems already discussed, the inversion of $N_x$ is only unique if both $G_x$ and $x$ are provided. The method of reconstruction is to identify the cliques which where introduced when applying $N$ to the input graph $G$. In the following, let $V(G) = \{v_1, \ldots, v_{|V(G)|}\}$, and let $C_1, \ldots, C_{|V(G)|}$ be the corresponding cliques in $G_x = N_x(G)$.

First, assume we have detected some clique $C_i$, with $V(C_i) = \{v_{i,1}, \ldots, v_{i,x}\}$. Choose any two vertices $v_{i,k}, v_{i,l} \in V(C_i)$ and any path $(v_{i,k}, w_1, w_2, v_{i,l})$ of length three using no edge in $C_i$. Due to the construction of $G_x$, both vertices $w_1$ and $w_2$ must belong to some clique $C_j \neq C_i$, and $\{v_i, v_j\} \in E(G)$. Further, for $x \geq 2$, we can observe that for any choice of $c_{v_i,k}, c_{v_i,l}$, and one of their neighbors not in $C_i$, there must exist a unique vertex forming a path of length three. As a result, we can start from a clique $C_i$ and detect all other cliques $C_j$ within the same connected component of $G_x$. This can be done in the following way. Choose any outgoing edge $\{v_{i,k}, v\}$ of $C_i$. If vertex $v$ exists, it must belong to some clique $C_j$ connected to $C_i$ by a complete matching. To detect $C_j$, we locate all paths of length three that start in $v_{i,k}$, proceed to $v$, and end at some vertex in $C_i$. All vertices on these path which are not endpoints induce $C_j$. The principle is outlined in Figure 3.4.

Thus, for every connected component in $G_x$, starting with any of its cliques $C_i$, we can identify all other cliques $C_j$ of the component. If we can detect at least one clique in every connected component in $G_x$, we can construct a graph isomorphic to $G$.

It remains to show how to identify a clique within a connected component of $G_x$. To do so, we iteratively examine all pairs of vertices $\{v, w\}$ and test whether the set of vertices $\{v, w\} \cup ( \ N(v) \cap N(w) \ )$ induces a clique of size $k$. Let $\{v, w\} \in E(C_i)$, for some $C_i$. It is straightforward that $|N(v) \cap N(w)| = x - 2$. Thus, we have obtained a necessary condition whether two vertices can be extended to a required clique. However, this condition is not sufficient, consequently, when we detect two connected vertices $v$ and $w$ with appropriate common neighborhood (i.e., both size and clique condition hold), we need further testing. We proceed as above, and try to partition the vertices into cliques of size $k$. There are two possible outcomes:

1. In the first case, we fail to partition $V(G_x)$ correctly. Consequently we know that the initial clique was not an intended one, and thus we proceed with the next pair of vertices. Note that, in every non-empty connected component of $N_x$, there is at least one clique $C_i$. Thus, we finally detect some pair of vertices such that the second case holds.

2. In the second case, we find a valid partitioning of $V(G_x)$. If the initial pair of vertices $\{v, w\}$ belongs to some clique $C_i$, we have in fact reconstructed all cliques $C_j$ within this connected component in $G_x$.

   It remains to show that the reconstruction is also possible, if $v$ and $w$ belong to two different cliques $C_i$ and $C_j$. Since the partitioning process stopped properly, we know that $v$ and $w$ are connected. Further, we know that $v$ and $w$ belong to the same copy $\tilde{G}$ of $G$ (with respect to the second interpretation of operator $N$, see above). The same is true for their common neighborhood. Consequently the initial clique $\tilde{C}$ is also a subgraph of $\tilde{G}$.

   Because of the way how $G_x$ is constructed, the partitioning process has identified all $x$ copies of $\tilde{C}$ in $G_x$. Let us consider all other neighbors of $\tilde{C}$ in $N_x$. Once again, since the partitioning process stopped properly, we can conclude that these vertices have been partitioned into cliques and that those cliques are connected to $\tilde{C}$ via complete matchings. This property can be extended to all vertices of the corresponding connected component $H_{\tilde{C}}$ in $\tilde{G}$. The vertices of $H_{\tilde{C}}$ are therefore partitioned into cliques of size $x$. The vertices of these cliques can be labeled in such a way that all matchings connect vertices with the same label (see left side side of Figure 3.5). Therefore, $\tilde{G}$ is isomorphic to a graph $N_x(\hat{G})$, for some appropriate graph $\hat{G}$.

   For the reverse construction, we collapse every clique into a vertex and remove parallel edges. Thus, $H_{\tilde{C}}$ results in some graph isomorphic to $\hat{G}$ (see right hand side of Figure 3.5). So does every copy of $H_{\tilde{C}}$ in $G_x$. We finally end up with $x$ copies of $\hat{G}$ where corresponding vertices are completely inter-connected. This graph matches the definition of $N_x(\hat{G})$ and thus is

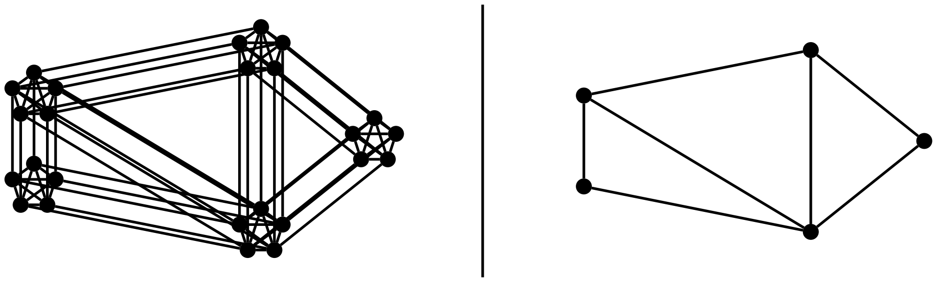

Figure 3.5: Example for elements in the reconstruction process of $N_5$ (left: copy of $G$, right: corresponding reconstruction)

isomorphic to the connected component of $\tilde{G}$ containing the vertices $v$ and $w$.

Combining both cases, we have proven that the inverse construction outputs a graph isomorphic to $G$.

The observation within the final part of the second case can be interpreted as follows. If $G$ is isomorphic to $N_y(G')$, for some graph $G'$, i.e., $G_x = (N_x \circ N_y)(G')$, we cannot distinguish which of the two operations $N_x$ or $N_y$ was applied first. Thus, although we can test whether a given graph is the result of an application of the operation $N$ for some specific input $x$, it is possible that several choices of $x$ are admissible. Thus, if only $G_x = N_x(G)$ is given, a unique reconstruction of $G$ is in general not possible.

## 3.1.4 The Transformation $T^{\alpha}_{r,N(r)}$

The fourth transformation (denoted $T^{\alpha}_{r,N(r)}$) adds to the given graph another graph, whose vertex degrees differ by at most one. This graph is added either disjointly or in such a way that the old and the new vertices are completely connected. Accordingly, $T$ is called the *graph-adjoining operation.*

### 3.1.4.1 Definition of $T^{\alpha}_{r,N(r)}$

Before stating a definition of $T$, we discuss the class of graphs being added. First of all, we define a graph to be *quasi-regular* if its vertex degrees differ by at most one. In the following, we prove that it is possible to build a quasi-regular graph efficiently for all possible tuples $(n, m)$, with $0 \leq m \leq \binom{n}{2}$.

**Lemma 3.1** *For every $n, m \in \mathbb{N}$, with $0 \leq m \leq \binom{n}{2}$, both given in unary (i.e., as inputs $1^n$ and $1^m$), a quasi-regular graph having exactly $n$ vertices and $m$ edges can be computed in time polynomial in the input length.*

*Proof*: Define $d^* = \lceil \frac{2m}{n} \rceil$ and $d_* = \lfloor \frac{2m}{n} \rfloor$. There are two distinct cases: either $d^*$ is even or $d_*$ is even. First, let $d_*$ be even. Compute a $d_*$-regular graph (e.g., by

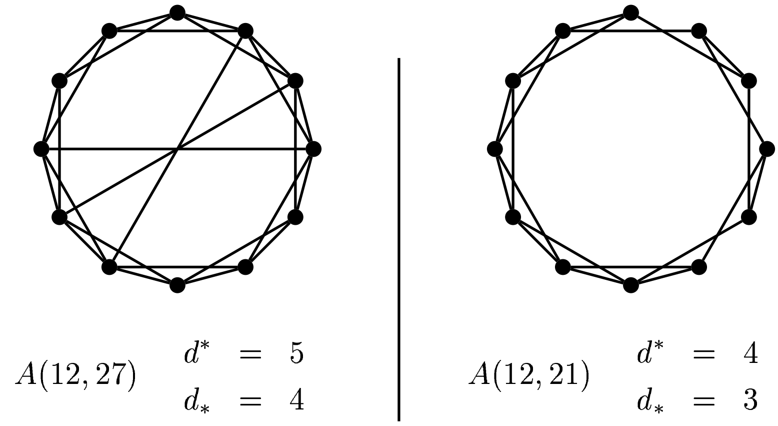

Figure 3.6: Example for construction of quasi-regular graphs $A(r, N(r))$

considering $n$ circular ordered vertices, connecting each vertex with its $d_*/2$ left and its $d_*/2$ right neighbors in the circle) and add some arbitrary matching of size $m - (d_*/2)n$. It is always possible to add such a matching since

$$m - \frac{d_* n}{2} \leq m - \left(\frac{2m}{n} - 1\right)\frac{n}{2} = \frac{n}{2}.$$

If $d^*$ is even then compute a $d^*$-regular graph and remove an existing matching of size $(d^*/2)n - m$. Analogously to the first case, this is possible since

$$\frac{d^* n}{2} - m \leq \left(\frac{2m}{n} + 1\right)\frac{n}{2} - m = \frac{n}{2}.$$

Both cases are outlined in Figure 3.6. Clearly, the graphs can be computed in time polynomial in $n$ and $m$, and hence polynomial in the size of the input. □

Further, for a matter of convenience, we define $A(n, m)$ to be a quasi-regular graph on $n$ vertices and $m$ edges.

Using the above idea, i.e., firstly arranging the vertices in circular order, secondly connecting every vertex to some number of its neighbors, and then thirdly adding resp. removing edges, it is possible to state a unique construction process for all reasonable tuples $(n, m)$. In the following we use $A(n, m)$ to denote the resulting graph for input $(n, m)$. Further, it is possible to do define $A(n, m)$ in such a way that the following properties also hold:

- $A(n, m)$ can be tested for isomorphism with any graph $G$ in time polynomial in $n$, $m$, and the size of $G$.
- Let $H = (V, E)$ be a graph, and let $\bar{H} = (V, V^2 \setminus E)$ be its complement. $A(n, m)$ (resp., $\bar{A}(n, m)$) is disconnected iff $m < n$ (resp., $m > \binom{n}{2} - n$).

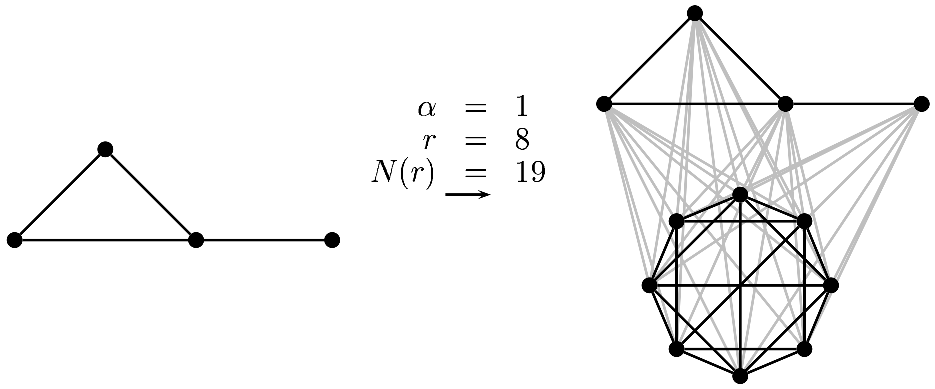

Figure 3.7: The graph-adjoining operator $T^{\alpha}_{r,N(r)}$

Based on the construction of $A(n,m)$ (which is not give in more detail) we state the formal definition of $T^{\alpha}_{r,N(r)}$. Within this definition, we use $r$ and $N(r)$ instead of $n$ and $m$ when building the quasi-regular graph. On the one hand, this avoids misunderstandings concerning the number of vertices and edges in $G$, and on the other hand, this choice of parameters better fits the latter settings.

Let $G$ be any undirected graph, $\alpha \in \{0,1\}$, $r \in \mathbb{N}$, and $0 \leq N(r) \leq \binom{r}{2}$. We define $T^{\alpha}_{r,N(r)}(G) = (V', E')$ as follows. Based on the definitions

$$G_1 = A(r, N(r)),$$
$$E^* = \begin{cases} \emptyset & , \textit{if } \alpha = 0 \\ \{\ \{v,w\} \mid v \in V(G), w \in V(G_1)\ \} & , \textit{if } \alpha = 1 \end{cases}$$

we set

$$V' = V(G)\ \cup\ V(G_1)$$
$$E' = E(G)\ \cup\ E(G_1)\ \cup\ E^*$$

For parameters $\alpha = 1$, $r = 8$, and $N(r) = 19$, this process is illustrated in Figure 3.7.

Unlike the three transformations $R$, $S$, and $N$, the operator $T$ is applied without considering the topology of the input. For that reason, a correlation of the occurrence of cliques in the input graph to the occurrence of some corresponding subgraphs in the output graph cannot be stated in general. However, for special settings of the parameters, when applying all four transformations $R$, $S$, $N$ and $T$ sequentially, we are able to prove that the input contains a clique of size $k$ if and only if the final graph contains a subgraph on $k'$ vertices and some correlated density $\gamma(k')$. The non-trivial proof of this property and the existence of the desired parameter sets constitutes the technical part of the **NP**-completeness proofs in this thesis (see sections 4.3 and 4.4).

#### 3.1.4.2 Inversion of $T^{\alpha}_{r,N(r)}$

To conclude the discussion on $T$, we once again consider the corresponding inversion problem. Given a graph $G' = T^{\alpha}_{r,N(r)}(G)$, a unique reconstruction of $G$ is, in general, only possible if both parameters $r$ and $N(r)$ are supplied. E.g., it is easy to see that for a graph $G' = T^{\alpha}_{r_1,N_1(r_1)}(T^{\alpha}_{r_2,N_2(r_2)}(\hat{G}))$ we can not decide which of the two transformations $T^{\alpha}_{r_1,N_1(r_1)}$ or $T^{\alpha}_{r_2,N_2(r_2)}$ was applied first, if we have no information on $r_1$ and $N(r_1)$. On the contrary, if parameter $\alpha$ is missing, while $r$ and $N(r)$ are given, we can state two processes in such a way that only one succeeds and outputs a unique reconstruction of $G$.

First of all, assume that all parameters are supplied. Dependent on $\alpha$ we can detect an induced subgraph isomorphic to $A(r, N(r)$. Removing this subgraph from $G' = T^{\alpha}_{r,N(r)}(G)$ results in some graph isomorphic to $G$. We consider the following two cases:

1. $\alpha = 0$: We know that $G$ and $A(r, N(r)$ are disconnected. Therefore, we have to find an induced isolated subgraph that is isomorphic to the connected components of $A(r, N(r))$. Based on the properties of $A(r, N(r))$, stated above, we know that $A(r, N(r))$ is disconnected if and only if $N(r) < r$. In such a case, due to quasi-regularity, the connected components consist of isolated vertices or paths of predefined length. It is easy to find the required number of these components. Thus, we can choose a subgraph isomorphic to $A(r, (N(r))$ in time polynomial to the size of $G'$. Otherwise, if $r \geq N(r)$, we know that $A(r, N(r))$ is connected. Due to definition of $A(r, N(r))$, it is possible to check every component on isomorphism to $A(r, N(r))$, in time polynomial in $r$, $N(r)$ and the size of the component.

2. $\alpha = 1$: In this case, we know that, $G$ and $A(r, N(r))$ are completely connected. In the following we consider $\bar{G}'$ instead of $G'$. It is easy to see that the components of $\bar{A}(r, N(r))$ are not connected to the rest of the graph. Therefore, similar to above, we can select a subgraph isomorphic to graph $\bar{A}(r, N(r))$ in time polynomial in the size of $G'$. Removing the corresponding vertices from $G'$ results in a graph isomorphic to $G$.

It is easy to see that only one of the two cases can hold for some triple $(G', r, N(r))$. Thus, parameter $\alpha$ is not required to guarantee a unique reconstruction of the input graph.

## 3.2 General local graph transformation

Within the reduction of CLIQUE to $\gamma$-DSP we use the combined graph transformation $N_x \circ S_t$. This operation replaces all vertices and edges without considering the rest of the graph. The resulting transformation can be generalized using a *general local graph transformation* (denoted by $\text{Trans}_{G_v,G_e}$) that depends on two graphs $G_v$ and $G_e$. These two graphs represent the image of a transformed vertex or edge, respectively. In chapter 4, we use this generalized family of transformations, when analyzing the quality of the derived lower bound for **NP**-completeness of $\gamma$-DSP in $\beta$-PL graphs.

Let $G_v$ and $G_e$ be two undirected graphs, and let $n_v = |V(G_v)|$ and $n_e = |V(G_e)|$. For any undirected input graph $G$, the graph $G' = \text{Trans}_{G_v,G_e}(G)$ is constructed as follows:

> Within the construction, we assume some fixed orderings on the set of vertices of $G_v$ (i.e, $V(G_v) = \{v_1, v_2, \ldots, v_{n_v}\}$) and the set of vertices of $G_e$ (i.e, $V(G_e) = \{w_1, w_2, \ldots, w_{n_e}\}$).
>
> Further, we require a sequence $S = (\,(l'_1, l''_2)\,, \ldots,\, (l'_{n_v}, l''_{n_v})\,)$ of $n_v$ pairs of indices of $V(G_e)$ in such a way that, for all $1 \leq i \leq n_v$, the two vertices $w_{l'_i}, w_{l''_i} \in V(G_e)$ are topologically equivalent w.r.t. $G_e$ ($w_{l'_i}$ and $w_{l''_i}$ may be identical).

- Initially we choose graph $G'$ to be empty.
- For every vertex $\tilde{v} \in V$ we add a copy $G_v^{\tilde{v}} = (V_v^{\tilde{v}}, E_v^{\tilde{v}})$ of graph $G_v$ to $G'$.
- For every edge $\tilde{e} = (\tilde{v}, \tilde{w}) \in E$ we add a copy $G_e^{\tilde{e}} = (V_e^{\tilde{e}}, E_e^{\tilde{e}})$ of graph $G_e$ to $G'$. Further, the two graphs $G_v^{\tilde{v}}$ and $G_v^{\tilde{w}}$, corresponding to vertices $\tilde{v}$ and $\tilde{w}$, are connected to $G_e^{\tilde{e}}$ as follows:

  > Let $V(G_v^{\tilde{v}}) = \{v_1^{\tilde{v}}, \ldots, v_{n_v}^{\tilde{v}}\}$, $V(G_v^{\tilde{w}}) = \{v_1^{\tilde{w}}, \ldots, v_{n_v}^{\tilde{w}}\}$, further let $V(G_e^{\tilde{e}}) = \{w_1^{\tilde{e}}, \ldots, w_{n_e}^{\tilde{e}}\}$. Using sequence $S$ we add the following $2n_v$ edges:
  >
  > $$\{v_1^{\tilde{v}}, w_{l'_1}^{\tilde{e}}\} \cdots \{v_{n_v}^{\tilde{v}}, w_{l'_{n_v}}^{\tilde{e}}\} \text{ and } \{v_1^{\tilde{w}}, w_{l''_1}^{\tilde{e}}\} \cdots \{v_{n_v}^{\tilde{w}}, w_{l''_{n_v}}^{\tilde{e}}\}$$

Figure 3.8 illustrates an example of a general local transformation $\text{Trans}_{G_v,G_e}$ applied to a single edge, with $G_v = K_3$ and $G_e = C_8$. Using to the above definition and a suitable choice of sequence $S$, it is straight forward to choose graphs $G_v$ and $G_e$ in order to get

$$\text{Trans}_{G_v,G_e} \equiv N_x \circ S_t.$$

Therefore, the following argumentation on the average degree of the output graphs of general local graph transformations can be used to analyze the reduction process within the **NP**-completeness proof of $\gamma$-DSP.

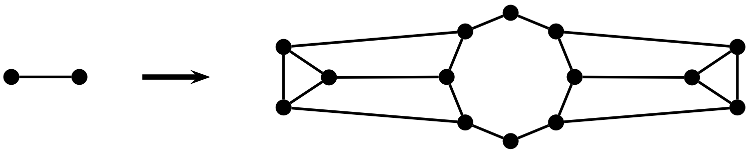

Figure 3.8: General local transformation using $G_v = K_3$ and $G_e = C_8$

### 3.2.1 Average Degree

In the following, we analyze the average degree $\delta'$ of a graph $G' = \mathrm{Trans}_{G_v,G_e}(G)$, dependent on the sizes and average degrees of the graphs $G_v$, $G_e$, and $G$. Within the analysis we use the following abbreviations:

$$n_v = |V(G_v)| \quad n_e = |V(G_e)| \quad \alpha_v = \frac{2|E(G_v)|}{|V(G_v)|} \quad \alpha_e = \frac{2|E(G_e)|}{|V(G_e)|} \quad \delta = \frac{2|E(G)|}{|V(G)|}$$

According to the definition of the general local graph transformation we get:

$$\begin{aligned}
\delta' &= \frac{2|E(G')|}{|V(G')|} = \frac{|V(G)| \cdot 2|E(G_v)| \ + \ |E(G)| \cdot (4n_v + 2|E(G_e)|)}{|V(G)| \cdot n_v \ + \ |E(G)| \cdot n_e} \\
&= \frac{|V(G)| \cdot \alpha_v n_v \ + \ \frac{1}{2}\delta|V(G)| \cdot (4n_v + \alpha_e n_e)}{|V(G)| \cdot n_v \ + \ \frac{1}{2}\delta|V(G)| \cdot n_e} \\
&= \frac{\alpha_v n_v \ + \ \frac{1}{2}\delta \cdot (4n_v + \alpha_e n_e)}{n_v + \frac{1}{2}\delta \cdot n_e} \ = \ \alpha_e + \frac{2n_v(\alpha_v - \alpha_e + 2\delta)}{2n_v + \delta \cdot n_e} \\
&= \alpha_e + \frac{2(\alpha_v - \alpha_e + 2\delta)}{2 + \delta \cdot \frac{n_e}{n_v}}
\end{aligned}$$

This result can be interpreted as follows:

1. The average degree of $G'$ does not depend on the size but on the average degree of $G$. Therefore, if the average degree of the input graph is known to be constant (e.g., scale-free graphs), the average degree of the output is also constant, regardless of the size of the input graph.

2. Increasing the value of $\frac{n_e}{n_v}$, while fixing the values of $\alpha_v$, $\alpha_e$, and $\delta$, we can decrease the value of the second term in the last sum. Therefore, independent of the input graph, we can choose $G_v$ and $G_e$ such a way that the resulting average degree is well approximated by $\alpha_e$ (with small error term).

Within the proof of the **NP**-completeness of the $\gamma$-DSP, we use the second property. I.e., we transform the input graph in such a way that the average degree of any subgraph of the output graph (on some specific number of vertices)

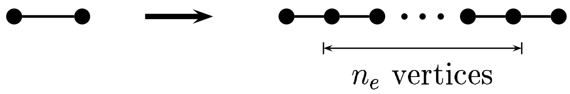

Figure 3.9: Modeling operation $S_t$ by using a general local transformation

is bounded from above by an appropriate chosen threshold (correlated to $\alpha_e$). Further, when considering subgraphs, we prove that the gap to this threshold is minimal, when considering a subgraph that correspond to the image of some densest subgraphs of the input. Using additional transformations, this gap is shown to be zero if and only if the input graph contains a clique of demanded size (see Theorem 4.4).

### 3.2.2 Special choices for $G_v$ and $G_e$

In the following, we consider some special types of graphs $G_v$ and $G_e$ (e.g., required when $\mathrm{Trans}_{G_v,G_e} \equiv N_x \circ S_t$) and discuss the corresponding average degree of graphs $G' = \mathrm{Trans}_{G_v,G_e}(G)$.

- First of all, we restrict to the case that graph $G_v$ consists of a single vertex. Therefore, we get $n_v = 1$ and $\alpha_v = 0$, and consequently, the value of $\delta$ results to:

  $$\delta' = \alpha_e + \frac{4\delta - 2\alpha_e}{n_e \cdot \delta + 2}$$

  In the following we consider two types of graph $G_e$:

  1. Let $G_e$ be a line segment on $n_e$ vertices, and $S$ defined in such a way that the two copies of $G_v$ are connected to different ends of the line segment (see Figure 3.9).

     Based on this choice of $G_v$, $G_e$, and $S$ we get:

     $$\mathrm{Trans}_{G_v,G_e} \equiv N_x \circ S_t \equiv S_t \qquad \text{with } t = n_e \text{ (and } x = 1)$$

     Using $\alpha_e = \frac{2n_e - 2}{n_e} = 2 - \frac{2}{n_e}$, the average degree $\delta'$ of $G'$ evaluates to:

     $$\delta' = \frac{\delta|V| + 2n_e(\frac{1}{2}\delta|V|)}{|V| + n_e(\frac{1}{2}\delta|V|)} = \frac{2\delta + 2n_e\delta}{2 + n_e\delta} = 2 + \frac{2}{n_e} \cdot \frac{\delta - 2}{\delta + \frac{2}{n_e}}$$

     Based on the upper bound $2 + \frac{2}{n_e}$ of $\delta'$, it is easy to see that for any input graph $G$, with $|E(G)| > 0$, we get

     $$|E(G')| = \frac{1}{2}\delta'|V(G'| < |V(G')| + \frac{|V(G')|}{n_e}.$$

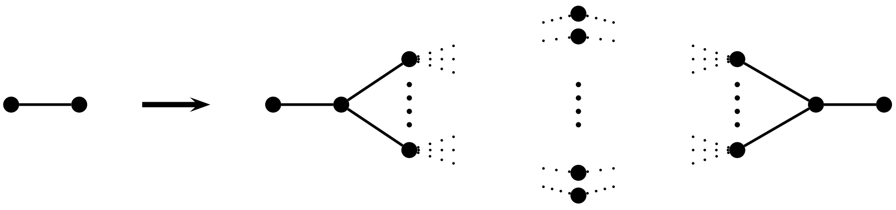

Figure 3.10: Example for tree-based a general local transformation

Further, using

$$|V(G')| \leq \binom{|V(G)|}{2} n_e + |V(G)| \leq |V(G)|^2 \cdot n_e,$$

we observe that, for all $\varepsilon > 0$, there exists some sufficient large value of $n_e$ (polynomial in $|V(G)|$) in such a way that

$$|E(G')| \leq |V(G')| + |V(G)|^2 \;\in\; |V(G')| + o(|V(G')|^{\varepsilon}).$$

This property is used, when proving **NP**-hardness for $\gamma$-DSP in general graphs, with $\gamma(k) \in k + \Omega(k^{\varepsilon})$.

2. Later in this thesis, when restrict to $\beta$-PL input graphs to $\gamma$-DSP, we would prefer to choose graph $G_e$ in such a way that it contains more vertices with higher degree, but still has average low degree.

   E.g., we could define $G_e$ to be the graph that results when taking two copies of a rooted tree and collapsing corresponding leaves. Further, when applying $\text{Trans}_{G_v,G_e}$ to some edge $\{\tilde{v}, \tilde{w}\}$ we add a matching between the two roots of the trees the vertices corresponding to $\tilde{v}$ and $\tilde{w}$, similar to above (see Figure 3.10).

   Let $x$ be the number of leaves in the underlying tree. The average degree of $G_e$ evaluates to $\alpha_e = \frac{2(n_e+x-2)}{n_e}$ and thus the average degree $\delta'$ of $G' = \text{Trans}_{G_v,G_e}(G)$ results to:

$$\begin{aligned}\delta' &= \frac{2\delta|V(G)| + 2(n_e + x - 2)(\frac{1}{2}\delta|V(G)|)}{|V(G)| + n_e(\frac{1}{2}\delta|V(G)|)} \;=\; \frac{4\delta + 2(n_e + x - 2)\delta)}{2 + n_e\delta}\\ &= 2 + \frac{2x\delta - 4}{n_e\delta + 2} \;=\; 2 + \frac{2x}{n_e} \cdot \frac{\delta - \frac{2}{x}}{\delta + \frac{2}{n_e}}\\ &= \alpha_e + 4\frac{\delta - \frac{n_e+x-2}{n_e}}{2 + \delta \cdot n_e} \;<\; \alpha_e + \frac{4}{n_e}\end{aligned}$$

   Once again, we can see that the average degree of the final graph strongly depends on the value of $\alpha_e$. Using other definitions of $G_e$,

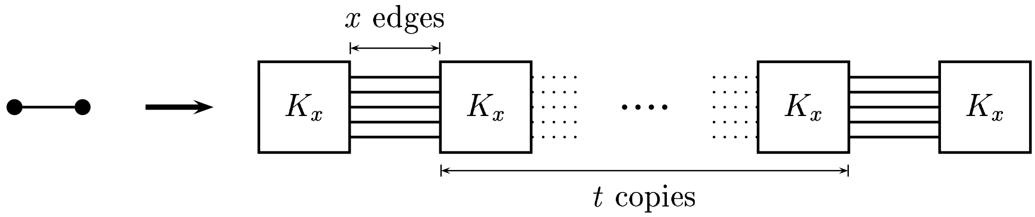

Figure 3.11: General local transformation modeling $N_x \circ S_t$

this idea could be extended to guarantee specific degree distributions for the final graph. However, within the latter discussion, we will see that it is not possible to use arbitrary degree distributions for $G_e$ (e.g., many vertices with degree one) without making the **NP**-completeness proof fail.

- After choosing $G_v$ to consist of an isolated vertex, we show how to define graphs $G_v$ (and $G_e$) in order to guarantee that the minimum degree of $G' = \text{Trans}_{G_v,G_e}(G)$ is only slightly above some given constant, for any input graph $G$. This property is used within the **NP**-completeness proof of $\gamma$-DSP for $\beta$-PL graphs (see section 4.4). Further, the choice of $G_v$ and $G_e$ also guarantees that transformation $\text{Trans}_{G_v,G_e}$ equals $N_x \circ S_t$.

  Let $x - 1$ be the required minimum degree. We define $G_v$ to be a complete graph on $x$ vertices and $G_e$ to correspond to $t$ copies of cliques of size $x$ that are connected chain-like using $t-1$ matchings of size $x$. When transforming an edge $\{v, w\}$ the clique corresponding to $v$ is connected to one end of the chain and $w$ to the other (once again, using matchings of since $x$). This construction is outlined in Figure 3.11 and is equivalent to transformation $N_x \circ S_t$. Once again let $G' = \text{Trans}_{G_v,G_e}(G)$. The average degree $\delta$ of $G$ evaluates to:

$$\begin{aligned}\delta' &= \frac{|V(G)|\binom{x}{2} + \delta|V(G)| \cdot x + \frac{1}{2}\delta|V(G)| \cdot tx \cdot (x+1)}{|V(G)| + \frac{1}{2}\delta|V(G)|tx} \\ &= \frac{x(x-1) + 2\delta x + \delta tx(x+1)}{2 + \delta tx} \\ &= (x+1) + \frac{x}{t} \cdot \frac{x + 2\delta - 3 - \frac{2}{x}}{\delta x + \frac{2}{t}}\end{aligned}$$

  Choosing $t \gg x$ we can approximate $\delta' \approx x + 1$. Further, all vertices have degree at least $x-1$ (or $x$ resp. $x+1$, if $G$ contains no isolated vertices resp. vertices of degree one). Within the **NP**-completeness proof we use input graphs in such a way that the minimum degree is exactly $x + 1$.

## 3.3 Combined transformations $N_x \circ S_t \circ R_s$

In this section, we investigate the resulting graph structure when applying iteratively the three graph transformations $R$, $S$, and $N$ that have been defined in the section 3.1. We prove (see Lemma 3.2) the equivalence of the existence of a clique of size $k$ in the input graph and the existence of a subgraph in the output graph with corresponding size and density. Within the lemma, we do not require the existence of some topological unique subgraph in the output graph but only the existence of a subset of vertices of appropriate size with some minimum number of existing edges between these vertices.

**Lemma 3.2** *Let $G$ be an undirected simple graph, and let $s, t, x \in \mathbb{N}$, with $x \geq 1$. Further, we assume $(\ t = 0\ ) \Rightarrow (\ x = 1\ )$.*

> *$G$ contains a clique of size $k$ if and only if the transformed graph $G' = (N_x \circ S_t \circ R_s)(G)$ contains a subgraph (not necessarily induced) on exactly*
>
> $$x \cdot \left(k + s + t\binom{k+s}{2}\right) \text{ vertices}$$
>
> *and at least*
>
> $$\binom{x}{2} \cdot \left(k + s + t\binom{k+s}{2}\right) + x \cdot (t+1) \cdot \binom{k+s}{2} \text{ edges.}$$

*Proof*: If $t = 0$, and thus $x = 1$, we know $G' = (N_1 \circ S_0 \circ R_s)(G) = R_s(G)$. From Observation 3.1 we derive that $G$ contains a clique of size $k$ if and only if $G'$ contains a subgraph on $k + s$ vertices and at least $\binom{k+s}{2}$ edges. Obviously,

$$\begin{aligned} k + s &= x \cdot \left(k + s + t\binom{k+s}{2}\right) \\ \binom{k+s}{2} &= \binom{x}{2} \cdot \left(k + s + t\binom{k+s}{2}\right) + x \cdot (t+1) \cdot \binom{k+s}{2} \end{aligned}$$

and, consequently, the proposition holds.

For the rest of the proof, we assume $t, x \geq 1$. If $G$ contains a clique $C$ of size $k$, the subgraph of $G'$ that corresponds to the transformation of $C$ has the require number of vertices and edges. It remains to show that $G$ contains a clique of size $k$ if $G'$ contains some subgraph $H$ with the required number of vertices and edges.

First of all, we prove that the existence of $H$ implies the existence of some set of vertices $X$, with $|X| = |V(H)|$, which has the following properties:

1. For each cliques $C_i$, introduced by transformation $N$, either all or none vertices are contained in $X$.

2. For all but at most one chain, introduced by transformation $S_t$, either all or none of the vertices that correspond to the $t$ inner vertices are contained in $S$. If a chain is contained entirely, it holds that the vertices that correspond to the two incident outer vertices are also elements of $X$.

3. Let $G_x$ be the subgraph of $G'$ that is induced by $X$. The number of edges in $G_x$ is at least as high as the number of edges in $H$.

To proof the existence of such a set $X$, we start with the set of vertices $V(H)$ and iteratively reselect vertices until we end up with some set that has the desired properties. Within the proof, we guarantee that the number of induced edges does not decrease, at any time. Further, when speaking of cliques we consider those cliques $C_i$ that have been build when applying operation $N$. Similarly, if it is clear from context, we use the terms edges and degrees instead of induced edges and induced degrees.

We know that, according to transformation $N$, the vertices of $G$ can be partitioned into cliques of size $x$ and that the vertices of each clique can be labeled in such a way that inter-clique edges connect vertices with the same label. Thus, it is always possible to reselect the vertices within every clique in such a way that vertices with smallest label are chosen first. Obviously, the number of edges does not decrease. Throughout the whole process, when removing or adding vertices of a clique, we proceed according the this order on the vertices.

In the following, we show how exchange vertices in order to to select entire cliques. Doing so, we refer to those cliques that correspond to inner vertices of the operation $S$ to inner cliques, and to those of outer vertices to outer cliques. Assume that there exist at least two cliques which are not selected entirely (Since the number of vertices in the subgraph is a multiple of $x$ it never happens that only one clique is partially selected). According to the following rules, applied in the same order as stated, we choose two of these cliques, $A$ and $B$, and exchange vertices. Let $a = |V(A)|$ and $b = |V(B)|$ (only considering selected vertices).

1. Try to choose two inner cliques, where $a \neq b$ (w.l.o.g., we assume $a < b$).

   Due to $t \geq 1$, every selected vertex in $A$ is connected to at most $(a-1)+2 = a+1$ vertices. We remove a vertex in $A$ and add the next unselected vertex in $B$. The new vertex is connected to at least $b \geq a+1$ vertices in $B$. Therefore the number of edges does not decrease. This procedure is iterated until either no vertices of $A$ or all vertices of $B$ are selected.

2. Try to choose two inner cliques ($a = b$), where $A$ is connected to some (inner or outer) clique $C$ with $z$ ($\neq a$) selected vertices.

   If $z < a$ then the vertex with highest label in $A$ has induced degree at

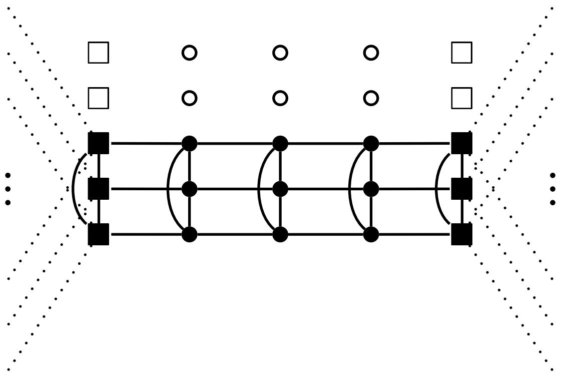

Figure 3.12: A typical situation when exchanging vertices in $(N_x \circ S_t \circ R_s)(G)$

most $a$ and thus exchanging that vertex to the next unselected vertex in $B$ does not decrease the number of edges. Otherwise, if $z > a$, we can remove the top selected vertex from $B$ (degree at most $b+1$) and choose the next unselected vertex in $A$. The new vertex must be connected to a selected one in $C$, due to $z > a$. Once again, the number of induced edges does not decrease. We continue exchanging vertices until one of the cliques is unselected or the other one is selected completely.

3. Consider all induced connected components of $G'$ which contain a partially selected inner clique. Note that within all cliques the same number $z$ of vertices are selected (otherwise case 1 holds). Further (since case 2 does not hold), all inner cliques have two neighboring cliques with the same number of selected vertices. For $t = 3$, $z = 3$, $x = 5$ this situation is outlined in Figure 3.12 (circles resp. squares represent vertices of inner resp. outer cliques; selected vertices are filled).
   We choose one of those components. If it contains an outer clique which has only one induced neighboring clique, we choose that outer clique and an arbitrary one of the inner cliques. We can exchange vertices from the outer clique to the inner clique without decreasing the number of induced edges (similar to case 2). Otherwise, dependent on $z$, we distinguish two sub-cases:

   (a) If $z \geq 2$ choose any two inner cliques $A$ and $B$. It is possible to remove 2 vertices from $A$ (loss of $(z-1)+(z-2)+4 = 2z+1$ edges) and replace them with the next unselected vertex $v \in V(B)$ and one of the next unselected vertices $w \notin V(A)$ in the neighboring cliques of $B$ (gain of $z+z+1 = 2z+1$ edges). Vertex $w$ must exist since both neighboring cliques of $B$ are also partially selected. Now, we can continue exchanging vertices from $A$ to $B$ until the the number of partially selected cliques decreases.

(b) Otherwise, $z = 1$, in every partially selected cliques exactly one vertex is selected. Due to $t \geq 1$ and the choice of this case, we know that the component is isomorphic to a 2-edge-connected subgraph of graph $(S_t \circ R_s)(G)$. Let $v$ be an outer vertex with minimum induced degree within the selected component. Removing $v$ and all its neighbors leads to a loss of $i + 1$ vertices and $2i$ edges. Due to the 2-edge-connectivity of the initially induced graph the remaining (at least $i + 1$) vertices are still connected. We choose any $i + 1$ connected vertices and add in each of the corresponding cliques the next unselected vertex. Doing so, we gain $(i + 1) + i = 2i + 1$ edges and thus equalize the number of lost edges. Similar to the previous cases, the number of partially selected cliques is also reduced.

4. There exists exactly one partially selected inner clique $A$. All other inner cliques are either selected completely or not at all. Choose any partially selected outer clique $B$. Let $a'$ and $b'$ be the minimum induced degrees in $A$ and $B$. If $a' \leq b'$ remove the corresponding vertex of $A$ and choose some unselected vertex $v \in V(B)$. Obviously, $v$ has induced degree at least $b'$. Similar, if $a' > b'$ remove the corresponding vertex from $B$ and choose some unselected vertex $v \in V(A)$. This vertex has induced degree at least $a' - 1 \geq b'$. Therefore, the number of edges is not decreased. We continue exchanging vertices until the number of partially selected cliques decreases.

5. There exists no partially selected inner clique. Note, for any outer clique all its selected vertices have the same induced degree. Choose any two partially selected outer cliques $A$ and $B$. Let $a'$ and $b'$ be the corresponding induced degrees, w.l.o.g. we assume $a' \leq b'$. We can exchange vertices from $A$ to $B$ without decreasing the number of edges. Thus, after the maximum number of possible exchanges, once again the number of partially selected cliques decreases.

For all the above cases, we stated a method that guarantees to decrease the number of partially selected cliques. Thus, when iterating this process, finally, there remain no partially selected cliques, i.e., the first condition from above holds.

In order to further satisfy the second condition, we start to exchange complete cliques. Note that this has no impact on the truth of the first condition. Assume there exist two partially selected chains $A$ and $B$ (compared to the second property). We iteratively exchange inner cliques of $A$, which have only one selected neighboring clique, with unselected cliques of $B$, which have at least one selected neighboring clique. This process does not decrease the number of edges. Finally, the number of partially selected chains is decreased by one. Iterating this process results in at most one partially selected chain.

At this point the proof is nearly finished. So far, having started with some appropriate subgraph $H$ of $G'$, we have constructed a subgraph $H'$ of $G'$ on $x \cdot \bigl(k+s+t\binom{k+s}{2}\bigr)$ vertices and at least induced $\binom{x}{2} \cdot \bigl(k+s+t\binom{k+s}{2}\bigr) + x \cdot (t+1) \cdot \binom{k+s}{2}$ induced edges with the above properties. Using these properties, we can argue that $H'$ is isomorphic to a transformed clique of size $k$, i.e., $H \cong (N_x \circ S_t \circ R_s)(K_k)$.

Since every selected vertex of $H'$ is an element of an entirely selected clique, we know that there are $\bigl(k+s+t\binom{k+s}{2}\bigr)$ such cliques. Every cliques induce $\binom{x}{2} \cdot \bigl(k+s+t\binom{k+s}{2}\bigr)$ edges. Therefore the remaining edges, at least $x(t+1)\binom{k+s}{2}$, must be contained in the matchings connecting these cliques (i.e., the matchings are located within the corresponding chains). Each completely selected chain contributes $x(t+1)$ such edges. Thus, using property 2, there must be $\binom{k+s}{2}$ such chains, which correspond to the same number of "selected" edges in $R_s(G)$.

Now, we can conclude. In order to induce the required number of line segments (inner cliques) we need $x \cdot t\binom{k+s}{2}$ vertices. Thus, there remain at most $x \cdot (k+s)$ vertices for the outer cliques. This corresponds to at most $k+s$ vertices in $R_s(G)$. Using the above observation, these vertices must induce $\binom{k+s}{2}$ edges in $R_s(G)$. The only possibility to induce at least $\binom{k+s}{2}$ edges with at most $k+s$ vertices is to form a clique of size $k+s$. Therefore, the existence of $H$ implies the existence of a clique of size $k+s$ in $R_s(G)$. Using Observation 3.1, this is equivalent to the existence of a clique of size $k$ in $G$. This completes the proof of the second implication of the lemma. □

# Chapter 4

# The complexity of finding dense subgraphs

In the beginning of this thesis (chapter 1), we have discussed the usability of density-based cluster techniques, when abstracting large networks with small-world property. Further, in chapter 2, we have discussed the main characteristics of power-law graphs, a class of graphs that is often used to generalize large-scale networks occurring in a wide range of research areas. In the present chapter, we investigate the computational complexity of finding dense subgraphs in either general or power-law graphs (where, for a fixed number of vertices, density is measured in terms of the number of edges, or, equivalently, the average degree).

While finding subgraphs of arbitrary size with highest average degree can be done in polynomial time, we show that it is **NP**-complete to decide, whether a subgraph on exactly $k$ vertices and at least $\gamma(k)$ edges exists, for any function $\gamma \in k + \Omega(k^{\varepsilon})$ (with $\varepsilon > 0$). Due to the appearance of power-law degree structures on large-scale networks, the above problem is also discussed for this special graph class. We show that, despite of this restriction, the problem remains **NP**-complete for a wide range of functions $\gamma$.

## 4.1 Overview and complexity results

### 4.1.1 Definition of density

First of all, before presenting an overview on density based subgraph problems, we explain our choice to use the average degree as a measure of density.

In the literature, density of a (sub)graph is very often defined in terms of a function on the number of its vertices and edges. Basically, there are three standard definitions of density.

- The density $d_1(G)$ of a graph $G = (V, E)$, often referred to as relative density, is defined to be the fraction of the number of edges in $G$ relative to the maximum number of possible edges on the vertex set of $G$ (e.g., see [AF99, RGW02]):

  $$d_1(G) = \frac{|E|}{\binom{|V|}{2}}$$

  It is easy to see that $d_1(G) \in [\,0 \,..\, 1\,]$, for any graph $G$.

- The density $d_2(G)$ of a graph $G = (V, E)$ is defined to be the average degree of the vertices of $G$ (e.g., see [Gol84, FKP01]):

  $$d_2(G) = \frac{2|E|}{|V|} = \mathrm{avgdeg}(G)$$

  Instead, it is possible to use the ratio of the number of the edges to the number of vertices, since both values only differ by a factor two. We further observe $d_2(G) \in [\,0 \,..\, |V| - 1\,]$.

- The density $d_3(G)$ of a graph $G = (V, E)$ is defined to be its number of edges (e.g., see [SW98, AHI02]):

  $$d_3(G) = |E|$$

  Obviously, $d_3(G) \in [\,0 \,..\, \binom{|V(G)|}{2}\,]$.

If we use the above definitions to compare the density of different graphs, we get different results, depending on the structure of the underlying graphs. As a case in point, we consider definition $d_1$. On the one hand, due to its range $[\,0 \,..\, 1\,]$, it seems to be very intuitive since it enables to compare graphs of different sizes directly. On the other hand, the maximal number of possible edges is quadratic in the number of vertices. Therefore, when using definition $d_2$ for comparing two graphs $G_1$ and $G_2$, with $|V(G_1)| = \frac{1}{2}|V(G_2)|$, we need $|E(G_1)| \approx 4|E(G_2)|$ in order to guarantee that $d_1(G_1) = d_1(G_2)$. Thus we get that smaller graphs are "preferred" when using definition $d_1$. Similarly, when using definition $d_3$, larger graphs are favored.

However, if we assume that the number of edges increases linearly with the number of vertices (e.g., when considering graphs with scale-free characteristic, see discussion in 2.2.2.1), the definition of $d_2$ appears to be most appropriate. In chapter 2 we have observed that scale-invariance (or at least size-independent average degree) often arises when modeling large real-world data. For this reason, throughout this thesis, we use definition $d_2$ for measuring the density of graphs (most times, however, we consider the number of edges, which is an equivalent measure if considering subgraphs of a fixed number of vertices).

### 4.1.2 Dense subgraphs of arbitrary size

One of the most intuitive dense subgraph problems is to find the densest subgraph for some given input graph (denoted by Densest-Subgraph-Problem).

**Problem 4.1** Densest-Subgraph-Problem

*Input:* *undirected graph $G$*
*Output:* *subgraph $G'$ of $G$, which has maximum average degree with respect to all subgraphs of $G$*

Using flow techniques, Goldberg has shown that there is a polynomial-time algorithm for the Densest-Subgraph-Problem [Gol84]. The idea of the algorithm is to introduce two new vertices $s$ and $t$ that are connected to all vertices of the input graph $G$. Using appropriate weights for the new edges, while all edges in $G$ are assigned equal weights, it is possible to state an equivalence between the existence of a min-cut of size at most $\alpha$ and the existence of a subgraph with density at least some value correlated to $\alpha$. Combining several runs of this algorithm (using different edge weights), it is possible to evaluate a subgraph of $G$ with maximal density among all subgraphs of $G$. The fastest algorithm currently known for this problem, which is also based on flow techniques, has been developed by Gallo, Grigoriadis, and Tarjan [GGT89] and runs in time $O(mn \log(n^2/m))$. Modifying the above definition, it is also possible to consider edge-weighted graphs. These variants can be solved similarly.

Obviously, by applying a single min-cut computation, the corresponding decision problem Variable-Density-Subgraph-Problem can be decided in polynomial time. It is defined as follows.

**Decision-Problem 4.2** Variable-Density-Subgraph-Problem

*Input:* *$(G, d)$, where $G$ is an undirected graph, and $d \in \mathbb{Q}$*
*Question:* *Does $G$ contain a subgraph with average degree at least $d$?*

### 4.1.3 Dense subgraphs with required size

In the previous subsection, the dense subgraphs could be of arbitrary size. However, if we intend to construct a density-based clustering of a graph we usually want to partition the vertex set into subsets of comparable size and density. When applying the above methods iteratively to some input graph, there is no guarantee on the sizes and densities of the resulting clusters. Hence, this method, which detects some densest subgraph, cannot be applied to clustering, in general. As a consequence, we have to refine the approach by adding size and density restrictions for the desired subgraphs. Unfortunately, as we can see in the following, most of the corresponding decision problems are complete for **NP** and thus corresponding solutions are not computable in polynomial time (unless **P** equals **NP**).

The most general form of including size and density is stated in the GENERAL-DENSE-$k$-SUBGRAPH-PROBLEM:

**Decision-Problem 4.3** GENERAL-DENSE-$k$-SUBGRAPH-PROBLEM

*Input:* $(G, k, d)$*, where* $G$ *is some undirected graph,* $k \in \mathbb{N}$*, and* $d \in \mathbb{Q}$

*Question: Does* $G$ *contain a subgraph on* $k$ *vertices with average degree at least* $d$*?*

The membership of the problem in this complexity class **NP** can easily been verified. Further, we can use the query $(G, k, \binom{k}{2})$ to reduce an instance $(G, k)$ of CLIQUE, which is known to be **NP**-complete [Kar72]. Even when restricting to the DENSEST-$k$-SUBGRAPH-PROBLEM [SW98, AKK99, FKP01, AHI02], where we ask for the densest subgraph with exactly $k$ vertices, the problem is **NP**-hard (e.g., see [RRT94]).

#### 4.1.3.1 Fixed-parameter dense subgraph problem

Within the above problems, **NP**-completeness has been derived due to the possibly large number of demanded edges. However, in case of building abstractions for sparse input graphs, there is no need to require the existence of all possible edges (i.e., maximum density) within the subgraphs. Thus, in contrast to the above *variable* dense subgraph problems, where we can specify arbitrarily large average degree for the demanded subgraph, we restrict to *fixed-parameter* problems. The overall idea of this class of problems is to define the minimal required average degree of the desired subgraph (on $k$ vertices) in terms of a fixed function $\gamma$ that may depend on $k$. Especially, if we have detailed information on the overall graph structure (e.g., when operation on large-scale networks) it is possible to define dense subgraphs by correlating their sizes and densities, appropriately (e.g. some multiple of the average degree of the input graph). In the literature the number of edges is often used instead of the average degree in order to describe the density [FKP01, AHI02]. For the above problem the computational complexity is equivalent for both definitions of density, since the average degree and the number of edges differ by the factor $\frac{1}{2}k$, which can be hidden within function $\gamma$. In order to guarantee consistency with the literature we define the fixed-parameter $\gamma$-DENSE-$k$-SUBGRAPH-PROBLEM (denoted by $\gamma$-DSP) in terms of edges of the subgraph.

**Decision-Problem 4.4** $\gamma$-Dense-$k$-Subgraph-Problem ($\gamma$-DSP)

*Input:* $(G, k)$, *where* $G$ *is some undirected graph, and* $k \in N$
*Question:* *Does* $G$ *contains a subgraph on* $k$ *vertices with at least* $\gamma(k)$ *edges?*

Similar to the above definition of $\gamma$-DSP, it is possible to use hyper-graphs instead of undirected graphs. Nehme and Yu [NY97] investigated the complexity of the constrained maximum value sub-hypergraph problem, which contains the dense $k$-subgraph problem as a special case. They obtained bounds on the number of (hyper-)edges a (hyper-)graph may have in such a way that the problem is still polynomial-time solvable (namely, $n - s + \alpha \log n$ edges, where $n$ is the number of vertices, $s$ is the number of connected components, and $\alpha$ is any constant). Similarly, fixed parameter-restrictions to simple input graphs were also considered in [AITT00, AHI02]. These scenarios have no consequences for complexity of the above problem since the restrictions affect the graph outside of possible dense subgraphs, while we are interested in the existence of dense subgraphs of fixed quality inside any arbitrary graph (or graphs with different properties, e.g., power-low graphs).

The complexity of the general $\gamma$-DSP depends on the choice of function $\gamma$. Obviously, we may restrict to functions $\gamma$ with $0 \leq \gamma(k) \leq \binom{k}{2}$, since all other values of $\gamma(k)$ do not correspond to valid choices for the number of edges of a graph on $k$ vertices. It is easy to see that the problem is trivially solvable in polynomial time for $\gamma(k) = 0$ (i.e., we do not demand the existence of any edge and thus any subgraphs on $k$ vertices is a valid solution). Further, if we choose $\gamma(k) = \binom{k}{2}$ the problem is equivalent to deciding on the existence of a clique on $k$ vertices an thus $\gamma$-DSP is known to be **NP**-complete. The containment in two different (besides $\mathbf{P} = \mathbf{NP}$) complexity classes for the two extreme choices of $\gamma$ poses the question if there exists dichotomy (i.e. there exists some threshold in such a way that $\gamma$-DSP is solvable in polynomial for all choices of $\gamma$ blow this threshold, and is contained in the class of **NP**-complete problems, otherwise). The analysis of this question is a main theoretical focus in this thesis and is presented within the following sections. The overall result enables to state membership in **P** resp. **NP**-c for a wide range of functions. Nevertheless, a small gap remains open and thus dichotomy can only be assumed but not proven, in general.

In the literature, there are several results on the complexity of $\gamma$-DSP for specific functions $\gamma$. Asahiro, Hassin, and Iwama [AHI02] studied the $k$-$f(k)$ dense subgraph problem, $(k, f(k))$-DSP for short, which asks whether there is a $k$-vertex subgraph of a given graph $G$ which has at least $f(k)$ edges. Therefore, besides the notation, this problem is equivalent to $\gamma$-DSP. The authors have proved that the problem remains **NP**-complete for $f(k) = \Theta(k^{1+\varepsilon})$ for all $0 < \varepsilon < 1$ and is polynomial-time solvable for $f(k) = k$. Feige and Seltser [FS97] even proved

| | | $\mathcal{P}$ | $\mathcal{NP}$-c |
|---|---|---|---|
| Feige, Seltser | (1997) | — | $\gamma(k) = k + k^{\varepsilon}$ |
| Asahiro et al. | (2002) | $\gamma(k) = k$ | $\gamma(k) = \Theta(k^{1+\varepsilon})$ |
| H et al. | (2003) | $\gamma(k) = k + O(1)$ | $\gamma(k) = k + \Omega(k^{\varepsilon})$ |

Table 4.1: Overview on the complexity results for $\gamma$-DSP

the special result that $(k, f(k))$-DSP is **NP**-complete if $f(k) = k + k^{\varepsilon}$ for any $0 < \varepsilon < 2$. In this thesis, we show how to improve these bounds. Further, while the previous results only considered special choices of function $\gamma$, we prove upper and lower bounds for the containment in **P** or **NP**-c, respectively, and thus provide an almost complete classification of $\gamma$-DSP with respect to algorithmic difficulty depending on the choices of $\gamma$ [HKMT02, HKMT03]. A short comprehensive overview of the results is stated in Table 4.1.

The detailed proofs of our results are presented in the succeeding sections. First of all, in section 4.2, we show that $\gamma$-DSP is polynomial-time solvable for functions $\gamma \in k + O(1)$, i.e. asking whether a graph contains some subgraph on $k$ vertices and at least $k + c$ edges, for some $c > 0$. Moreover, by analyzing the polynomial-time algorithm, we easily observe that the problem can be solved in subexponential time $2^{n^{o(1)}}$ for $\gamma =\in k + k^{o(1)}$. Further, in section 4.3 we prove that $\gamma$-DSP is **NP**-complete for $\gamma \in k + \Omega(k^{\varepsilon})$ for $0 < \varepsilon < 2$. These results, for general graphs, are subsumed in Theorem 4.1.

**Theorem 4.1** *Let $\gamma : \mathbb{N} \to \mathbb{N}$ be a function that is computable in polynomial time, with $0 \leq \gamma(k) \leq \binom{k}{2}$.*

1. *If $\gamma \in k + O(1)$, then $\gamma$-*DSP *is solvable in polynomial time.*
2. *If $\gamma \in k + k^{o(1)}$, then $\gamma$-*DSP *is solvable in subexponential time.*
3. *If $\gamma \in k + \Omega(k^{\varepsilon})$ for some $\varepsilon > 0$, then $\gamma$-*DSP *is* **NP***-complete.*

All in all, we establish a rather sharp boundary between polynomial time solvable and **NP**-complete cases. As a more intuitive formulation of the problem, i.e., when considering constant average degree, we obtain that detecting a $k$-vertex subgraph of average degree at least two (which is nearly the case of any connected graph) can be done in polynomial time whereas finding a $k$-vertex subgraph of slightly-higher average degree, at least $2 + \varepsilon$, for some $\varepsilon > 0$, is already **NP**-complete. Thus, density-based clustering is inherently hard as a general methodology.

**$\gamma$-DSP on power-law graphs** In chapter 2 we described the family of $\beta$-PL graphs that builds an abstract graph representation for may natural large-scale networks. Obviously, the graphs used within the **NP**-completeness proof of $\gamma$-DSP do not belong to the class of $\beta$-PL graphs. Therefore, the computational complexity for $\gamma$-DSP may differ, when restricting to power-law graphs, and thus presumably when applying this technique to large real-world date. While the bound for polynomial tractability also holds when restricting to a subclass of input problems this is not true for **NP**-completeness, in general. Therefore, in section 4.4, we investigate the **NP**-completeness for $\gamma$-DSP on $\beta$-PL graphs. Unlike to the general problem we can only prove **NP**-completeness for functions $\gamma(k) \geq \frac{15}{11}\delta k$, where $\delta$ is the maximum average degree for all $\beta$-PL graphs for some constant value of $\beta$. Referring to this restricted version as $\gamma$-DSP-$\beta$-PL we can summarize these results in Theorem 4.2.

**Theorem 4.2** *Let $\gamma : \mathbb{N} \to \mathbb{N}$ be a function that is computable in polynomial time, with $0 \leq \gamma(k) \leq \binom{k}{2}$, and let $\beta > 2$.*

1. *If $\gamma \in k + O(1)$, then $\gamma$-DSP-$\beta$-PL is solvable in polynomial time.*
2. *If $\gamma \in k + k^{o(1)}$, then $\gamma$-DSP-$\beta$-PL is solvable in subexponential time.*
3. *If $\gamma(k) \geq \frac{15}{11}\delta k$, then $\gamma$-DSP-$\beta$-PL is **NP**-complete, where $\delta$ is the maximum average degree of all $\beta$-PL graphs.*

Different to $\gamma$-DSP for general graphs, there remains a larger gap between the derived bounds for polynomial time solvability and **NP**-completeness. In section 4.5 we discuss the quality of the lower bound, when using the proposed reduction technique. Doing so, we observe that where as it seems possible to slightly decrease the factor of $\frac{15}{11}$ (by applying further refinements), it is not likely that the applied reduction technique can be used to prove significantly better lower bounds. In order to do so we require stronger results on the class of $\beta$-PL graphs. Nevertheless, despite of laking exact proofs for the computational complexity of the remaining gap, we give evidence that for almost all function $\gamma$ within the remaining gap the problem $\gamma$-DSP restricted to $\beta$-PL may be assumed to be **NP**-complete, similar to the result on general graphs.

**Structure of the proofs** The proof of the polynomial-time cases is mainly based on dynamic programming over collections of minimal subgraphs having certain properties. For instance, for the above-mentioned polynomial-time result for $(k, f(k))$-DSP with $f(k) = k$ [AHI02], we simply need to find shortest cycles in a graph, which is easy. For functions $f(k) = k + c$ with $c > 0$, the search for similar minimal subgraphs is not obvious to solve and is the main difficulty to overcome in order to obtain polynomial-time algorithms.

In the **NP**-hardness proofs we extend techniques used in [FK94, GNY94, FS97, AHI02], that are well suited for $\Theta$-behavior of functions but do not suffice for

$\Omega$-behavior. The completeness is proved by reduction from a special version of CLIQUE using elementary graph transformations. These transformations can be divided in *densification* (by enriching graphs with vertices and edges so that only cliques consisting of at least half of the vertices are relevant [AHI02]) and *sparsification* (either by vertex replacement, e.g., replacing each vertex with a cycle on $r$ vertices [FS97], or by edge replacement, e.g., replacing each edge with a path of length $s$ [FK94, GNY94, FS97]). Whereas densification increases average degrees, sparsification is intended to lower average degrees of transformed subgraphs. However, as both kinds of transformation are rather coarse in its effects (for instance, it is easy to calculate that vertex replacement with $r > 0$ vertices together with edge replacement with path length $s > 1$ maps cliques to very sparse graphs having average degree less than $2 + \varepsilon$, for every constant $\varepsilon > 0$), we need additional transformations of intermediate density effects to adjust fine-grained behavior. This is realized by *graph adjoining* (i.e., adding graphs having prescribed numbers of vertices and edges). The main issue remaining for getting results for $\Omega$-behavior is to unify reductions for several growth classes by a non-trivial choice of the parameters involved in transformations.[1] When restricting to the input class of power-law graphs, we require some further graph (having subgraphs of sufficiently bounded average degree) in order to guarantee the desired overall degree distribution.

## 4.2 Finding $(k + O(1))$-dense subgraphs in polynomial time

In this section, we show how to solve $\gamma$-DSP for $\gamma \in k + O(1)$ in time polynomial in the number of vertices. In other words, we prove that searching a subgraph on $k$ vertices with at least $k + c$ edges, with $c$ constant, is a polynomial-time problem [HKMT02, HKMT03]. We will formalize this problem as EXCESS-$c$ SUBGRAPH.

For a graph $G$, let the *excess* of $G$, denoted by $\mathfrak{e}(G)$, be defined as the difference of the number of edges and the number of vertices, i.e., $\mathfrak{e}(G) = |E(G)| - |V(G)|$. A (sub)graph $G$ with $\mathfrak{e}(G) \geq c$ is said to be an *excess-c (sub)graph.*

[1] Basically, having Turán's theorem [Tur41] in mind, one could ask whether it is possible, at least in the case of dense graphs, to deduce intractability results using inapproximability of MAXIMUM CLIQUE due to Håstad [Hås99]: there is no polynomial-time algorithm finding cliques of size at least $n^{\frac{1}{2}+\varepsilon}$ (where $n$ is the size of the maximum clique) unless $\mathbf{P} = \mathbf{NP}$. Assume we would have a polynomial-time algorithm for $\gamma$-DSP with, e.g., $\gamma(k) = \beta\binom{k}{2}$ and $0 < \beta < 1$, are we now able to decide whether there is a clique of size $k^{\frac{1}{2}+\varepsilon}$? Turán's theorem [Tur41] says that there is a clique of size $k$ in a graph with $n$ vertices and $m$ edges, if $m > \frac{1}{2}n^2\frac{k-1}{k-2}$. Unfortunately this implies that we can only assure that in a graph with $n$ vertices and at least $\beta\binom{n}{2}$ edges, there is a clique of size at most $\frac{3-2\beta}{1-\beta}$, which is constant and makes the argument fail. This objection remains valid with the recent improvements of inapproximability results for MAXIMUM CLIQUE as in [Kho01].

**Problem 4.5** EXCESS-$c$ SUBGRAPH

| | |
|---|---|
| *Input:* | *$(G,k)$, where $G$ is some undirected graph, and $k \in N$* |
| *Output:* | *Does $G$ contain an excess-$c$ subgraph with (exactly) $k$ vertices?* |

We will show how to find excess-$c$ subgraphs in polynomial time. The general solution is based on the case of a connected graph as it is considered in the following lemmata.

**Lemma 4.1** *Let $c \geq 0$ be any integer. Given a connected graph $G$ on $n$ vertices, an excess-$c$ subgraph of minimum size can be computed in time $O(n^{2c+2})$.*

*Proof*: Let $G$ be any connected graph with $\mathfrak{e}(G) \geq c$. Obviously, there exists a subgraph $G_c$ of minimum size with excess $c$. For the degree-sum of $G_c$ we obtain

$$\sum_{v \in V(G_c)} \deg_{G_c}(v) = 2|E(G_c)| = 2(|V(G_c)| + c).$$

Since $G_c$ is minimal w.r.t. the number of vertices, there exists no vertex with degree less than two. Therefore, the number of vertices with degree greater than two in $G_c$ is at most

$$\sum_{v \in V(G_c)} (\deg_{G_c}(v) - 2) = 2(|V(G_c)| + c) - 2|V(G_c)| = 2c.$$

Let $S$ be the set of all vertices in $G_c$ with degree greater than two. If there is a path connecting vertices $u, v \in S$ using only vertices from $V(G_c) \setminus S$ ($u$ and $v$ are not necessarily distinct), then there can be no shorter path connecting $u$ and $v$ containing vertices from $V(G) \setminus V(G_c)$. Otherwise $G_c$ would not be minimal w.r.t. the number of vertices. In the following we will describe how to find such a subgraph $G_c$, if it exists.

We examine all sets $S' \subseteq S$ of size at most $2c$, i.e., the elements of $S'$ are those vertices where paths can cross. For each such set we can iteratively construct a candidate $H(S')$ for $G_c$. In each step we include a path of minimum length among all paths connecting any two vertices in $S'$. We may restrict ourselves to those paths that do not intersect or join common edges, since those cases are covered by other appropriate choices of S'. This process is repeated until either excess $c$ is reached or no further connecting path exists. In the latter case, the set $S'$ does not constitute a valid candidate for $G_c$. Otherwise $H(S')$ is kept as a possible choice for $G_c$. After considering all possibilities for $S'$, the graph $G_c$ can be chosen as a vertex-minimal subgraph from all candidates that were found. Note that $G_c$ is not unique w.r.t. exchanging equal length paths.

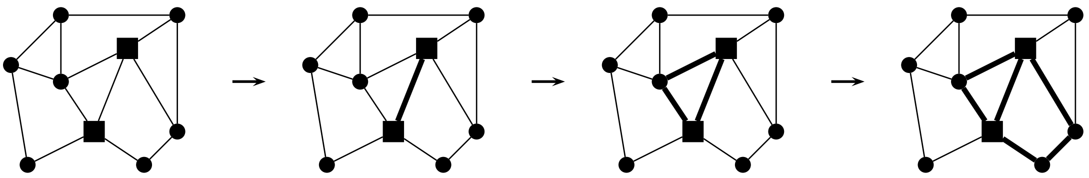

Figure 4.1: Breadth-first search for an excess-1 subgraph. For excess $c = 1$ we have at most $2c = 2$ starting vertices. Three shortest paths (thick lines) connecting two possible starting vertices (boxes) are located and added in order of increasing length.

Since $|S'| \leq 2c$, there are

$$\sum_{i=1}^{2c} \binom{|S|}{i} = O(n^{2c})$$

possible choices for $S'$. For the verification of a chosen set $S'$ that consists of $i$ vertices, we have to find iteratively $i + c$ shortest non-crossing paths, e.g., by using $i + c \leq 3c$ parallel breadth-first-search runs (see A.1), which takes overall time $O(3c|E(G)|) = O(n^2)$.

Hence an excess-$c$ subgraph of minimum size can be determined by testing all possible choices of $S'$ in total time $O(n^{2c+2})$. Figure 4.1 shows an example for $c = 1$. Note that for $c = 0$ we only have to find a shortest cycle (e.g., by breadth-first search), which can be done in time $O(n^2)$. □

Unfortunately, the algorithm of Lemma 4.1 cannot directly be used for the general case of possibly non-connected graphs. For these graphs a solution may contain vertices from different connected components. Therefore, our algorithm is based on solving the subproblem of maximizing the excess for a given number of vertices within a connected graph.

**Lemma 4.2** *Let $c \geq 0$ be any integer and $G$ be a graph with $n$ vertices. We define $\mathfrak{e}_i$ to be the maximum excess of a subgraph of $G$ on (exactly) $i$ vertices. Calculating $\min\{\mathfrak{e}_i, c\}$ for all values of $i \in \{0, 1, \ldots, n\}$ can be done in time $O(n^{2c+2})$.*

*Proof*: We first observe that $\mathfrak{e}_0 = 0$. Also, since $G$ is connected, $\mathfrak{e}_i \geq -1$ for all $i \in [\,1 \,..\, n\,]$. Furthermore, due to the connectivity of $G$ any subgraph can iteratively be extended without decreasing the excess. Thus, if there exists a subgraph on $i > 0$ vertices having excess $\mathfrak{e}_i$, the value $\mathfrak{e}_i$ is a lower bound for the maximum excess of subgraphs with more vertices. Therefore, it is sufficient to know the minimum number of vertices necessary to achieve excess $c$ (as done in Lemma 4.1).

The maximum excess we are interested in is bounded from above by $c$. We get the minimum number of vertices needed for all possible values of $\mathfrak{e} \in [\,0 \,..\, c\,]$ by

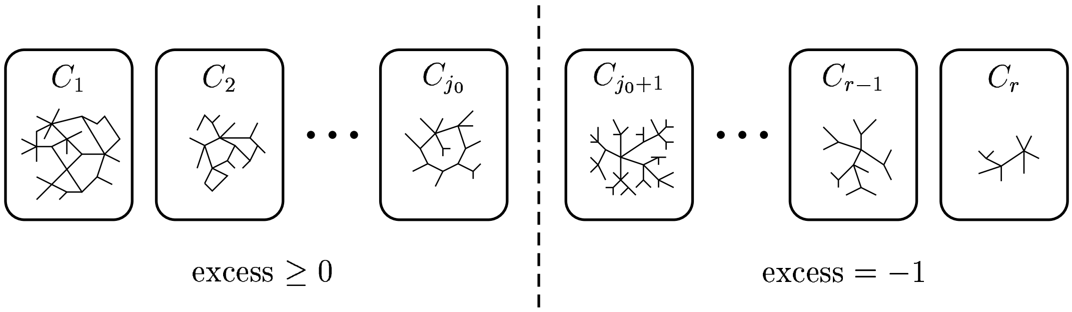

Figure 4.2: Ordering and classification of connected components. The components $C_j$ with excess $\mathfrak{e}(C_j) = -1$ are trees and therefore easier to handle.

performing $c+1$ iterations of the algorithm described in Lemma 4.1. Using these results we can easily calculate the desired value $\min\{\mathfrak{e}_i, c\}$, for each $i \in [\,0\,..\,n\,]$. This requires total time $O(n^{2c+2})$. □

Before we proceed to the main theorem, we have to discuss a further property. Let $(G, k)$ be the instance of the Excess-$c$ Subgraph problem, i.e., we have to find a subgraph of $G$ on $k$ vertices with at least $k + c$ edges. In time linear in $|V(G)| + |E(G)|$ we can (as a preprocessing step) partition $G$ into its connected components and calculate their excess. Let $C_1, \ldots, C_r$ be the list of the components, sorted non-increasingly by their excess. Note that $\mathfrak{e}(C_j) \geq -1$ since all components are connected. Let $j_0$ denote the maximum index of the components with non-negative excess and $k_0$ the total number of all vertices of those components (see Figure 4.2).

**Lemma 4.3** *Let $G$ be an undirected graph that consists of connected components $C_1, \ldots, C_r$. Further let $j_0$ and $k_0$ be defined as stated above.*

1. *If $k > k_0$ then there exists a maximum excess subgraph comprising all vertices from the non-negative excess components $C_1, \ldots, C_{j_0}$.*
2. *If $k \leq k_0$ then there always exists a subgraph of size $k$ having maximum excess within $G$ and consisting only of vertices from components with non-negative excess.*

*Proof:* Let $G'$ be an induced subgraph of $G$. Assume that $G'$ contains vertices of a component $C_i$ with negative excess while there exists a component $C_j$ with positive excess that is not contained entirely.

If at least one vertex in $C_j$ is selected, there exists another so far not selected vertex $v$ in $C_j$ that is adjacent to some selected vertex. Since $C_i$ must be a tree, there must exist a selected vertex $u \in C_i$ that is a leaf in the selection, i.e., it is incident to at most one edge in $G'$. By exchanging $u$ and $v$, no excess is lost.

Otherwise, no vertex in $C_j$ is selected. Once again we exchange leaves from $C_i$ with connected vertices from $C_j$. There are two possibilities. Firstly, if $C_j$ is selected entirely, we cannot lose excess because $\mathfrak{e}(C_j) \geq 0$. Secondly, if all vertices of $C_i$ were exchanged, once again we cannot lose excess since $\mathfrak{e}(C_i) = -1$ and all chosen vertices in $C_j$ are connected (and thus the induced graph has excess at least $-1$).

This process can be iterated until there all vertices in the components with negative excess are unselected or all components with positive excess are contained entirely. □

With these results we are able to state the main theorem of this section.

**Theorem 4.3** *Let $c$ be any integer. For any input $(G, k)$,* EXCESS-$c$ SUBGRAPH *can be decided in time $O(|V(G)|^{2|c|+4})$.*

*Proof*: Let $c$ be any fixed integer. Let $(G, k)$ be a problem instance. The problem can be divided into two cases.

**Case $k \geq k_0$.** In this case, the problem can be solved straightforward. Because of Lemma 4.3, there exists a maximum-excess subgraph on $k$ vertices that contains all components with non-negative excess entirely. Therefore, all remaining vertices must be chosen from the components with negative excess. Those components are trees (each having excess $-1$ by definition) and thus the selected vertices within these components induce a forest. Since we want to maximize the excess, we have to minimize the number of trees. Therefore, as long as possible, we choose complete components ordered by non-increasing size (i.e., largest trees first). From the next component we choose a subtree of sufficient size, to get exactly the desired number of vertices. This procedure determines the minimum number of trees to choose. Finally, the maximum excess of a subgraph of $G$ on $k$ vertices can be evaluated by adding up the excess of all used components. Obviously, in this case the time bound of the theorem holds.

**Case $k < k_0$.** Here, we may restrict our choice to those components with non-negative excess. We show that it is sufficient to calculate separately for each such component the minimum size of subgraphs for all values of excess within the fixed range $\{0, 1, \ldots, c+1\}$. The original problem can be decided by combining these solutions. For each component $C_j$ we create an array $A_j$. At index $i \in \{0, 1, \ldots, |V(C_j)|\}$ we store the maximum excess for any (induced) subgraph of component $C_j$ on $i$ vertices. As we will see later values larger than $c+1$ are of no interest. In these cases the lower bound $c+1$ will be used instead. Due to Lemma 4.2 array $A_j$ can be calculated in time $D|V(C_j)|^{2(|c|+1)+2}$ for some $D > 0$.

Hence, the total time to calculate the values for $A_j$ for all components is

$$\sum_{j=1}^{r} D|V(C_j)|^{2(|c|+1)+2} \leq D\left(\sum_{j=1}^{r} |V(C_j)|\right)^{2|c|+4} = O(|V(G)|^{2|c|+4}).$$

Based on the results of the calculation we can distinguish two different cases.

- If there exists a component that contains an excess-$(c+1)$ subgraph on $k_1 \leq k$ vertices, we can choose this subgraph and add a sufficient number $k-k_1$ of vertices such that the excess decreases by at most one. This can be achieved by appending remaining vertices of the component, adding entire so far unused components (with excess $\mathfrak{e} \geq 0$) and adding at most one incomplete component (a connected subgraph with excess $\mathfrak{e} \geq -1$).

- Otherwise, for the second case, all excess-$(c+1)$ subgraphs of any component have more than $k$ vertices. Assuming that we already calculated the values of the arrays $A_j$, we can compute the maximum excess of a $k$-vertex subgraph from $G$ by considering suitable subsets of the components. Therefore we have to decide how many vertices of each component have to be selected.[2]

  From each component $C_j$ at most $\min(|V(C_j)|, k)$ vertices can be selected. Remember that the corresponding maximum excess is stored in $A_j$. We iterate over all components and within the components over all possible subgraph-sizes and store the currently best sub-result in array $X$. This can be done in such a way that after each iteration $X[i]$ contains the maximum possible (bounded by $c$) excess for an $i$-vertex subgraph of the so far processed components. Finally $X[k]$ contains the value of the maximum excess for any subgraph on $k$ vertices. Thus, it can be decided whether there exists an excess-$c$ subgraph of size $k$. Figure 4.3 shows an algorithm (based on dynamic programming) for this calculation in pseudo-code.

  Since the total size of all components is bounded by $n$ (first and second loop) and $k \leq n$ (third loop), the calculation time of the algorithm is in $O(n^2)$.

It is easily seen that the total calculation time is on $O(|V(G)|^{2|c|+4})$ □

So far, we only considered the EXCESS-$c$ SUBGRAPH problem for constant values $c$. If we are interested in a $k$-vertex subgraph with excess function $f \in O(1)$, the same method can be applied. From $f \in O(1)$ we know that $f(k)$ is bounded from above by a constant $c'$. Obviously, the time complexity for our algorithm is

[2] Note that this problem is a variant of SUBSET SUM, using a set of integer-intervals $\{\ \{0,1,\ldots,|V(C_j)|\}\ \big|\ 0 \leq j \leq j_0\ \}$. Despite SUBSET SUM is **NP**-complete, this problem can be solved in polynomial time, because of the present unary representation of $n$.

```
initialize array $X[\,0\,..\,k\,] := [0, -1, \ldots, -1]$
initialize array $Y[\,0\,..\,k\,] := [0, -1, \ldots, -1]$
for all $j \in \{1, \ldots, j_0\}$ do
  for all $i \in \{1, \ldots, \min(|V(C_j)|, k)\}$ do
    for all $l \in \{0, \ldots, k - i\}$ do
      if $Y[l + i] < X[l] + A_j[i]$ then
        $Y[l + i] := X[l] + A_j[i]$
  copy array $Y$ to $X$
```

Figure 4.3: Algorithm for excess-aggregation of connected components.

$O(n^{2c'+4})$, if $f(k)$ can be computed in the same time. This problem corresponds to finding a $(k+O(1))$-dense subgraph. Applying some modifications the method can also be used to find such a subgraph instead of only deciding its existence.

**Corollary 4.1** *For polynomial-time computable functions $\gamma \in k + O(1)$, $\gamma$-DSP is is solvable in polynomial time. Moreover, finding a $\gamma$-cluster on $k$ vertices is also solvable in polynomial time.*

Looking carefully at our algorithm, we observe that the algorithm runs in time $O((f(k) + 1)^2 n^{2f(k)+4})$ for a given excess function $f$. From this, we easily obtain a subexponential time-complexity of the algorithm for excess function $f = k^{o(1)}$.

**Corollary 4.2** *Let $\gamma : \mathbb{N} \to \mathbb{N}$ be a polynomial-time computable function.*

1. *For $\gamma \in k + O(k^{o(1)})$, finding $\gamma$-clusters can be done in time $2^{n^{o(1)}}$.*
2. *If $\gamma \in k+\Theta(k^{o(1)})$ and $\gamma$-DSP is **NP**-complete, then **NP** $\subseteq$ **DTIME**$\left(2^{n^{o(1)}}\right)$.*

The second statement of the corollary makes the **NP**-completeness of $\gamma$-DSP for $\gamma \in k + \Theta(k^{o(1)})$ unlikely to hold since subexponential-time simulations of **NP** problems are neither known nor expected. On the other hand, also a polynomial-time algorithm is not known for this problem. More interestingly, e.g., Excess-$\lfloor \log k \rfloor$ Subgraph could be a natural example of a problem which is neither **NP**-complete (unless **NP** is in quasi-polynomial time) nor solvable in polynomial time.

## 4.3 NP-completeness for $\gamma$-DSP with $\gamma \in \Omega(k + k^{\varepsilon})$

In this section we prove one of the main theorems of this thesis. We show that $\gamma$-DSP is complete in **NP** for all functions $\gamma \in k + \Omega(k^{\varepsilon})$ [HKMT02, HKMT03].

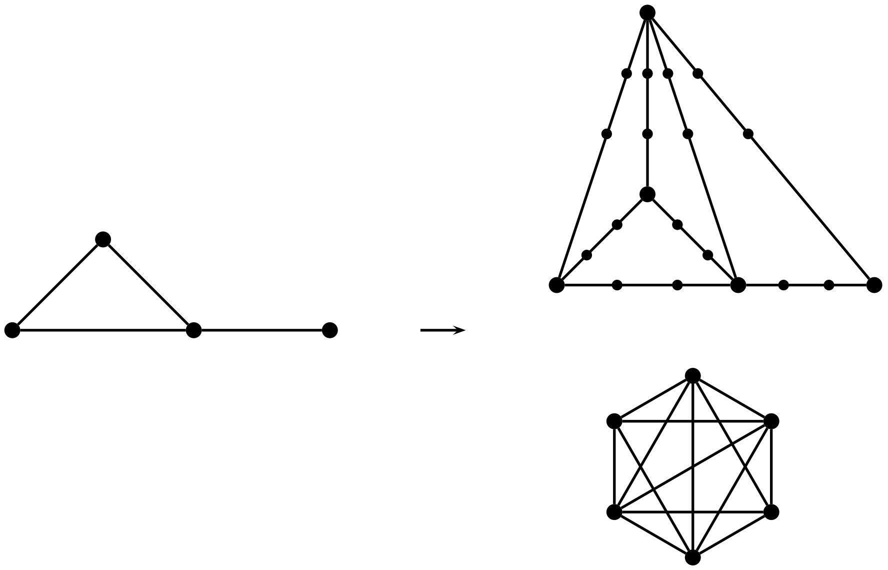

Figure 4.4: Example for the graph transformation $T^{0}_{6,14} \circ S_2 \circ R_1$

**Theorem 4.4** *Let $\gamma : \mathbb{N} \to \mathbb{N}$ be a polynomial-time computable function. Further we require $\gamma \in k + \Omega(k^{\varepsilon})$, for some fixed rational $\varepsilon > 0$, and $f(k) \leq \binom{k}{2}$. The problem $\gamma$-DSP, i.e., given a tuple $(G, k)$ deciding whether graph $G$ contains some subgraph on exactly $k$ vertices with at least $\gamma(k)$ edges, is* **NP***-complete.*

*Proof*: Let $f$ be a polynomial-time computable function with $f \in k + \Omega(k^{\varepsilon})$, for some arbitrary but fixed rational $\varepsilon > 0$ and $f(k) \leq \binom{k}{2}$.

Obviously, given some tuple $(G, k)$, deciding whether graph $G$ contains a subgraph on $k$ vertices with at least $f(k)$ edges is in **NP**. We proof **NP**-hardness by reduction of the **NP**-complete problem Clique$_{\frac{1}{2}}$ (see paragraph 3.1.1.3).

Let $(G, k)$ be any instance of the problem Clique$_{\frac{1}{2}}$. We state a graph transformation in such a way that graph $G$ contains a clique of size $k$ if and only if the transformed graph contains a subgraph on $k'$ vertices and at least $f(k')$ edges, for some appropriate choice of $k'$. The transformation process is based on the graph operations $R$, $S$, and $T$, presented in section 3.1. The choice of the parameters for these transformations will be divided into several cases, dependent on the value of $f(k')$. However, the transformation takes place according to some general construction which is described first. In the latter, we will consider the different cases in detail.

For all cases, we transform $G$ to graph $G' = (T^{\alpha}_{r,N(r)} \circ S_t \circ R_s)(G)$ for some fixed parameters $\alpha$, $r$, $N(r)$, $t$, and $s$. An example of such a transformation is illustrated in Figure 4.4. In the following, we describe how to choose the parameters to receive the desired result, in general.

First of all, we define $k'$ to be the number of vertices that corresponds to the size of the vertex set that results when applying the transformation $S_t \circ R_s$ to a clique of size $k$ plus the number of vertices of the quasi-regular graph added by operation $T^{\alpha}_{r,N(r)}$, i.e.,

$$k' = (k+s) + t\binom{k+s}{2} + r.$$

Now, we describe how to choose $N(r)$. Assume, $G$ contains some clique of size $k$. Obviously, the vertices of the transformed clique plus the vertices of the quasi-regular graph would induce $(t+1)\binom{k+s}{2} + \alpha \cdot r \cdot \left((k+s) + t\binom{k+s}{2}\right) + N(r)$ edges. We define $N(r)$ in such a way that this number of edges equals $f(k')$. Thus, we derive

$$N(r) = f(k') - (t+1)\binom{k+s}{2} - \alpha \cdot r \cdot \left((k+s) + t\binom{k+s}{2}\right).$$

In order to satisfy all requirements in the definition of operator $T$ (see subsection 3.1.4), we will prove that $0 \leq N(r) \leq \binom{r}{2}$ for each of the occurring parameter sets.

Now, to proof the desired property of $G'$, as stated in the theorem, we distinguish the two cases dependent on whether $G$ contains a clique of size $k$ or not.

For the first case, assume $G$ contains a clique of size $k$ as a subgraph. According to the definition of $k'$ and $N(r)$, we know that $G'$ contains some subgraph on $k'$ vertices and $f(k')$ edges. The existence of $G'$, i.e., a valid choice of the parameters of the transformations, is shown in the second case.

For the second case, assume $G$ does not contain a clique of size $k$. In order to prove the theorem, we must show that each subgraph of $G'$ on $k'$ vertices has less than $f(k')$ edges. To do so, we determine a subgraph with maximum number of edges among all subgraphs of $G'$ on $k'$ vertices. In particular, we will guarantee that there exists such a subgraph $H^*$ that contains all vertices of the quasi-regular graph $A(r, N(r))$. Using Lemma 3.2 and the definition of $N(r)$ this enables to complete the proof. First of all, consider the graph $\hat{H}$ that results when removing all vertices of $A(r, N(r))$ from $H^*$. Obviously, $\hat{H}$ must be some densest subgraph in $(S_t \circ R_s)(G)$ on $(k+s) + t\binom{k+s}{2}$ vertices. Since $G$ contains no clique of size $k$ we know (according to Lemma 3.2) that $\hat{H}$ has less than $(t+1)\binom{k+s}{2}$ edges. Now, using definition of N(r), it follows that $H^*$, and thus any subgraph of $G'$ on $k'$ vertices, has less than $f(k')$ edges.

It remains to show that graph $H^*$ always exists. To do so, let $H$ be any induced subgraph of $G'$ on $k'$ vertices that has maximum number of edges among all those subgraphs. Further, let $\tilde{G}$ be the induced subgraph on $G'$ that contains no vertices of $A(r, N(r))$, i.e., $\tilde{G} \cong (S_t \circ R_s)(G)$, and let $l = k' - r = k + s + t\binom{k+s}{2}$. The vertices of $H$ can be separated into vertices belonging to $\tilde{G}$ or $A(r, N(r)$,

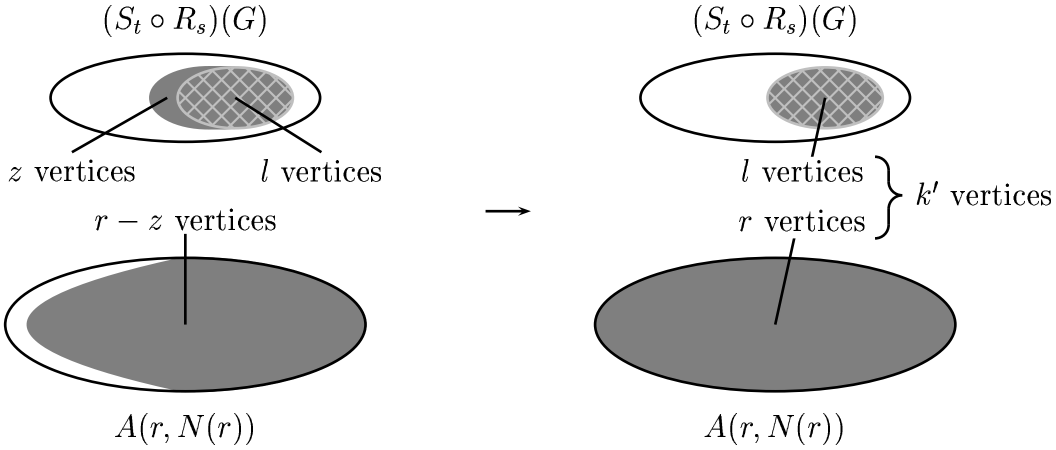

Figure 4.5: Choosing $H^*$ within **NP**-completeness proof for $\gamma$-DSP

respectively. Formally, there exists some $z$, with $0 \le z \le r$ (more precisely, $z \le \min\{r, |V(\tilde{G})| - l\}$), such that $l+z$ vertices of $H$ belong to $V(\tilde{G})$ and $r-z$ vertices of $H$ belong to $V(A(r,N(r)))$. This choice of subgraph is illustrated in Figure 4.5 (left hand side).

Now, let $H^*$ be the induced subgraph which results from replacing an appropriate choice of $z$ vertices in $V(H) \cap V(\tilde{G})$ with the remaining $z$ vertices of $A(r,N(r))$ that are not in $V(H)$ (see Figure 4.5, right hand side). We are done, if we can show that $|E(H^*)| \ge |E(H)|$ (see below). Note that, if $t \ge 1$, the vertices in $H$ can be iteratively removed in such an order that always a vertex with degree at most 2 is removed (simply by removing first all the inner vertices around an outer vertex and then the outer vertex which now has degree 0).

The above discussion concludes the proof of the theorem. In order to make it work, we have to choose all the parameters in such a way that the following two conditions (assumed above) can be satisfied:

- *Constructibility:* We have to guarantee that the graph $G'$ can be computed in polynomial time. Obviously, the operations $R_s$ and $S_t$ are polynomial computable if parameters $s$ and $t$ can be computed in polynomial time. In the latter, $r$ will depend polynomially on $k$ which is logarithmic in the size of the graph. Thus, a unary description of $r$ can be computed in polynomial time. Using Lemma 3.1 we know that the graph $A(r,N(r))$ exists and can be computed in polynomial time, if we further assure that $0 \le N(r) \le \binom{r}{2}$. Usually, in the latter cases, $N(r) \ge 0$ is seen easily and it is often proved together with the next condition.

- *Exchangeability:* This condition refers to the claim $|E(H^*)| \ge |E(H)|$ used above. Note that the claim is trivial for $z=0$. For $z \ge 1$ we consider the edge balance of transforming $H$ into $H^*$. In the case of $\alpha = 0$, which is the majority of our cases, we will argue as follows. On the one hand, we

| Case | upper bound | $s$ | $t$ | $r$ | $\alpha$ |
|---|---|---|---|---|---|
| I | $(1+D)k'$ | 0 | $(k'-r-k)\binom{k}{2}^{-1}$ | $\approx \lceil (4D^4k^2)^{(\frac{7}{6})^j} \rceil$ | 0 |
| II | $(1+D)k'^{\frac{3}{2}}$ | 0 | 1 | $k' - \binom{k}{2} - k$ | 0 |
| III | $\binom{k'}{2} - k'^{\frac{9}{8}}$ | 0 | 0 | $k' - k$ | 0 |
| IV | $\binom{k'}{2} - \frac{k'}{3}$ | $\lceil \frac{1}{3}k'^{\frac{1}{4}} \rceil - k$ | 1 | $k' - k - s - \binom{k+s}{2}$ | 1 |
| V | $\binom{k'}{2}$ | 0 | 0 | $k' - k$ | 1 |

Table 4.2: A rough summary of parameter settings. For a function value $f(k')$ choose the case with the least number in such a way the value is less than or equal to the corresponding upper bound.

remove at most $\Delta \cdot z$ edges from $\tilde{G}$, for some $\Delta$. On the other hand we add at least $\frac{1}{2}\lfloor\frac{2N(r)}{r}\rfloor z$ edges in $A(r, N(r))$. Thus, it is sufficient to satisfy that $\frac{1}{2}\lfloor\frac{2N(r)}{r}\rfloor z \geq \Delta \cdot z$, or, if $2\Delta \in \mathbb{N}$, equivalently, $\frac{N(r)}{r} \geq \Delta$. In the cases with $\alpha = 1$ we will employ more refined arguments.

After the outline of the general proof structure, we now state the precise choice of the remaining parameters. From the theorem, we know $f \in k + \Omega(k^\varepsilon)$, for some fixed $\varepsilon > 0$. Thus, there exists two natural numbers $k_0, D > 1$ in such a way that $k + D^{-1}k^\varepsilon \leq f(k)$, for all $k \geq k_0$. Obviously, we may suppose that $\varepsilon < \frac{1}{8}$ and $D \geq 5$.

Dependent on the value of $f(k')$ we choose different parameters. We define five cases that represent a partitioning of the corresponding interval for $f(k')$ between $k' + D^{-1}k'^\varepsilon$ and $\binom{k'}{2}$. To distinguish between these cases we choose the value of the parameter $k'$ to

$$k' = \left\lceil (D^6k^2)^{\frac{1}{\varepsilon}} \right\rceil .$$

Clearly, $k'$ is computable in time polynomial in the length of $k$. Now, depending on the function value $f(k')$ we choose the parameters $s$, $t$, $r$, and $\alpha$ in such a way, that the property $k' = k + s + t\binom{k+s}{2} + r$, assumed above, is guaranteed. A summary of the cases and the corresponding parameter settings is listed in Table 4.2. The exact choice of $r$ in Case I does not fit into the table and is fully stated, when considering this case.

In the following, to finally complete the proof, we show the two properties *constructibility* and *exchangeability*, for each of the five cases, separately.

**Case I:** $k' + D^{-1}k'^\varepsilon \leq f(k') \leq k' + Dk'$.

We split this case in several subcases. We consider, depending on $j$, with

$0 \leq j < \log_{\frac{7}{6}} \frac{1}{\varepsilon}$, the ranges $k' + D^{-1}k'^{\left(\frac{7}{6}\right)^{j}\varepsilon} \leq f(k') \leq k' + Dk'^{\left(\frac{7}{6}\right)^{j+1}\varepsilon}$. Clearly, we can combine those subcases to cover the complete range from $k' + D^{-1}k'^{\varepsilon}$ to $k' + Dk'$ as required for Case I. For each value of $j$, we apply $R_s$, $S_t$, and $T^{\alpha}_{r,N(r)}$ with the following parameters:

$$\boxed{\begin{array}{ccc} s = 0, & t = (k' - r - k)/\binom{k}{2}, & \alpha = 0, \\[1ex] \multicolumn{3}{c}{r = \left\lceil (4D^4k^2)^{\left(\frac{7}{6}\right)^{j}} \right\rceil + \left[ \left( k' - \left\lceil (4D^4k^2)^{\left(\frac{7}{6}\right)^{j}} \right\rceil - k \right) \mod \binom{k}{2} \right]} \end{array}}$$

The modular term in the definition of $r$ guarantees that $t \in \mathbb{N}$ . Trivially, we have $k + s + t\binom{k+s}{2} + r = k + t\binom{k}{2} + r = k'$. For $k$ large enough (and thus $k'$ and $r$, as well) constructibility and exchangeability can be satisfied as can be seen by the following calculations.

- *Constructibility:*

$$\begin{aligned} N(r) &= f(k') - (t+1)\binom{k}{2} \leq k' + Dk'^{\left(\frac{7}{6}\right)^{j+1}\varepsilon} - (t+1)\binom{k}{2} \\ &= r + Dk'^{\left(\frac{7}{6}\right)^{j+1}\varepsilon} - \binom{k}{2} + k \leq r + D\left(2D^6k^2\right)^{\frac{7}{6}\left(\frac{7}{6}\right)^{j}} \\ &\leq r + \left(3D^8k^3\right)^{\left(\frac{7}{6}\right)^{j}} \leq \left((2D^4k^2)^{\left(\frac{7}{6}\right)^{j}}\right)^2 \\ &\leq \left(\frac{1}{2}r\right)^2 \leq \binom{r}{2} \end{aligned}$$

- *Exchangeability:* Since $t \geq 1$ for $k > 0$, we can choose $\Delta = 2$ and we obtain the following:

$$\begin{aligned} N(r) &= f(k') - (t+1)\binom{k}{2} \geq k' + D^{-1}k'^{\left(\frac{7}{6}\right)^{j}\varepsilon} - (t+1)\binom{k}{2} \\ &\geq r + D^{-1}\left((D^6k^2)^{\frac{1}{\varepsilon}}\right)^{\left(\frac{7}{6}\right)^{j}\varepsilon} - \binom{k}{2} + k \\ &\geq r + (D^5k^2)^{\left(\frac{7}{6}\right)^{j}} - k^2 \geq 2r \end{aligned}$$

**Case II:** $k' + Dk' = (1 + D)k' < f(k') \leq (1 + D)k'^{\frac{3}{2}}$

$$\boxed{s = 0, \qquad t = 1, \qquad \alpha = 0, \qquad r = k' - \tbinom{k}{2} - k}$$

Clearly, we have $k + s + t\binom{k+s}{2} + r = \binom{k+1}{2} + r = k'$. For $k$ large enough (and thus $k'$ and $r$, as well), constructibility and exchangeability can be satisfied as can be seen by the following calculations.

- *Constructibility:*

$$\begin{aligned} N(r) &= f(k') - 2\binom{k}{2} \;\leq\; f(k') \;\leq\; (1+D)\left(r + \binom{k+1}{2}\right)^{\frac{3}{2}} \\ &\leq \frac{3}{2}D(2r)^{\frac{3}{2}} \;\leq\; \frac{9}{2}Dr^{\frac{3}{2}} \;\leq\; \frac{1}{4}r^2 \leq \binom{r}{2} \end{aligned}$$

- *Exchangeability:* Since $t = 1$ we can choose $\Delta = 2$ and we have the following:

$$\begin{aligned} N(r) &= f(k') - 2\binom{k}{2} \;\geq\; (1+D)\left(r + k + \binom{k}{2}\right) - 2\binom{k}{2} \\ &\geq (1+D)r + (D-1)\binom{k}{2} \;\geq\; 2r. \end{aligned}$$

**Case III:** $(1+D)k'^{\frac{3}{2}} < f(k') \leq \binom{k'}{2} - k'^{\frac{9}{8}}$

$$\boxed{s = 0, \qquad t = 0, \qquad \alpha = 0, \qquad r = k' - k}$$

Obviously, $k + s + t\binom{k+s}{2} + r = k + r = k'$. Note that since $\varepsilon < \frac{1}{8}$ we have $k^2 \leq k'^{\frac{1}{8}} \leq r$. For $k$ large enough (and thus $k', r$ as well), constructibility and exchangeability can be satisfied as can be seen by the following calculations.

- *Constructibility:*

$$\begin{aligned} N(r) &= f(k') - \binom{k}{2} \;\leq\; \binom{k+r}{2} - k'^{\frac{9}{8}} - \binom{k}{2} \\ &\leq \binom{r}{2} + k'(k - k'^{\frac{1}{8}}) \;\leq\; \binom{r}{2} \end{aligned}$$

- *Exchangeability:* Since $G$ has at most $\frac{3k}{2}$ vertices, we can set $\Delta = \frac{3k}{2} - 1$ (observe that $2\Delta \in \mathbb{N}$) and we obtain the following:

$$\begin{aligned} N(r) &= f(k') - \binom{k}{2} \;\geq\; (1+D)(k+r)^{\frac{3}{2}} - \binom{k}{2} \\ &\geq r\left((1+D)(k+r)^{\frac{1}{2}} - 1\right) \;\geq\; r(k^{\frac{1}{\varepsilon}} - 1) \;\geq\; \frac{3k}{2}r \end{aligned}$$

**Case IV:** $\binom{k'}{2} - k'^{\frac{9}{8}} < f(k') \le \binom{k'}{2} - \frac{k'}{3}$

$$\boxed{s = \left\lceil \frac{1}{3}k'^{\frac{1}{4}} \right\rceil - k, \qquad t = 1, \qquad \alpha = 1, \qquad r = k' - k - s - \binom{k+s}{2}}$$

Clearly, $s \ge 0$ and we have $k + s + t\binom{k+s}{2} + r = \binom{k+s+1}{2} + r = k'$. Moreover, it is easily seen that $r \ge k'^{\frac{3}{4}}$, since $r = k' - \binom{k+s+1}{2} \ge k' - \frac{1}{2}(k+s+1)^2 \ge k' - \frac{2}{3}k'^{\frac{1}{2}} \ge k'^{\frac{3}{4}}$. Furthermore, for $k$ large enough (and thus $k', r$ as well), constructibility and exchangeability can be satisfied as can be seen by the following calculations.

- *Constructibility:*

$$\begin{aligned}
N(r) &= f(k') - 2\binom{k+2}{2} - r\left(k + s + \binom{k+s}{2}\right)\\
&\le \binom{k'}{2} - \frac{k'}{3} - r\binom{k+s+1}{2} - 2\binom{k+s}{2}\\
&\le \binom{r + \binom{k+s+1}{2}}{2} - r\binom{k+s+1}{2} - \frac{k'}{3}\\
&\le \binom{r}{2} + \binom{\binom{k+s+1}{2}}{2} - \frac{k'}{3} \;\le\; \binom{r}{2} + \frac{1}{4}(k+s+1)^4 - \frac{k'}{3}\\
&\le \binom{r}{2} + \frac{1}{4}\left(\left\lceil \frac{1}{3}k'^{\frac{1}{4}} \right\rceil + 1\right)^4 - \frac{k'}{3} \;\le\; \binom{r}{2} + \frac{1}{4}\left(\frac{2}{3}k'^{\frac{1}{4}}\right)^4 - \frac{k'}{3}\\
&\le \binom{r}{2} + \left(\frac{4}{81} - \frac{1}{3}\right)k' \;\le\; \binom{r}{2}
\end{aligned}$$

- *Exchangeability:* Since $t = 1$, we can iteratively remove vertices in such a way that every removed vertex has maximum degree two (w.r.t. graph $(S_t \circ R_s)(G)$ ), when being removed. Thus, a vertex removed from $H$ was incident with at most $r - z + 2$ edges. Assume that every vertex in $A(r, N(r))$ has degree at least $(r-1) - \left(\binom{k+s+1}{2} - 2\right)$ within $A(r, N(r))$. Then a new vertex in $H$ that is chosen from $A(r, N(r))$ is incident with at least $(r - z) - \left(\binom{k+s+1}{2} - 2\right) + \binom{k+s+1}{2} = r - z + 2$ new edges. Therefore, we can exchange vertices consecutively in such a way that all vertices from $A(r, N(r))$ are chosen. The minimum degree of a vertex in $A(r, N(r))$ is $\lfloor \frac{2N(r)}{r} \rfloor$. Thus, we have to prove $\left\lfloor \frac{2N(r)}{r} \right\rfloor \ge (r-1) - \left(\binom{k+s+1}{2} - 2\right)$ what is equivalent to $2N(r) \ge 2\binom{r}{2} - \binom{k+s+1}{2}r + 2r$ since $k$, $r$, and $s$ are natural numbers. The inequality can be seen as follows:

$$\begin{aligned}
2N(r) &= 2\left(f(k') - 2\binom{k+2}{2} - r\left(k+s+\binom{k+s}{2}\right)\right) \\
&\geq 2\left(\binom{k'}{2} - k'^{\frac{9}{8}} - r\binom{k+s+1}{2} - 2\binom{k+2}{2}\right) \\
&= 2\left(\binom{r+\binom{k+s+1}{2}}{2} - \binom{k+s+1}{2}r - k'^{\frac{9}{8}} - 2\binom{k+s}{2}\right) \\
&= 2\binom{r}{2} + 2\binom{\binom{k+s+1}{2}}{2} - 2k'^{\frac{9}{8}} - 4\binom{k+s}{2}
\end{aligned}$$

Finally, we obtain the desired statement by the following calculations:

$$\begin{aligned}
&2\binom{\binom{k+s+1}{2}}{2} + \binom{k+s+1}{2}r - 2r - 2k'^{\frac{9}{8}} - 4\binom{k+s}{2} \\
&\quad\geq \frac{1}{4}(k+s)^4 + (k+s)^2\left(\frac{1}{2}r - 2\right) - 2r - 2k'^{\frac{9}{8}} \\
&\quad\geq \frac{1}{9}k'^{\frac{1}{2}}\left(\frac{1}{2}k'^{\frac{3}{4}} - 2\right) - 4k'^{\frac{9}{8}} \\
&\quad\geq \frac{1}{18}k'^{\frac{5}{4}} - 5k'^{\frac{9}{8}} \quad\geq\quad 0
\end{aligned}$$

**Case V:** $\binom{k'}{2} - \frac{k'}{3} < f(k') \leq \binom{k'}{2}$

$$\boxed{s = 0, \qquad t = 0, \qquad \alpha = 1, \qquad r = k' - k.}$$

Clearly, we have $k + s + t\binom{k+s}{2} + r = k + r = k'$. For $k$ large enough (and thus $k', r$ as well), constructibility and exchangeability can be satisfied as can be seen by the following arguments.

- *Constructibility:*

$$N(r) \;=\; f(k') - \binom{k}{2} - r \cdot k \;\leq\; \binom{k+r}{2} - \binom{k}{2} - r \cdot k \;=\; \binom{r}{2}$$

Further, since not proven within exchangeability:

$$\begin{aligned}
N(r) &= f(k') - \binom{k}{2} - r \cdot k \;\geq\; \binom{k'}{2} - \frac{k'}{3} - \binom{k}{2} - (k'-k) \cdot k \\
&= k'\left(\frac{k'-1}{2} - \frac{1}{3} - k\right) + k^2 - \binom{k}{2} \;\geq\; 0
\end{aligned}$$

- *Exchangeability:* Let $B$ be the densest $k$-vertex subgraph of $H$. Assume that $G$ has no clique of size $k$ (which in fact, is the only interesting case to consider). Hence, $B$ is not a clique. Since $B$ is the densest subgraph, each vertex of $H$ which does not belong to $B$ is adjacent to at most $k-1$ vertices of $B$. Thus, on the one hand, removing all vertices in $H \setminus B$ yields a loss of at most $z(r-z) + \binom{z}{2} + z(k-1)$ edges. On the other hand, since $A(r, N(r))$ misses at most $\frac{k'}{3} \leq \frac{r}{2}$ edges to be complete, each vertex of the quasi-regular graph $A(r, N(r))$ is adjacent to at least $r-2$ vertices, thus not connected to at most one vertex other than itself. Consequently, choosing all $z$ so far unselected vertices of $A(r, N(r))$ adds at least $(r-z)z + \binom{z}{2} - z + zk = z(r-z) + \binom{z}{2} + z(k-1)$ edges. Thus, an exchange of vertices is possible without decreasing the number of induced edges.

□

## 4.4 NP-completeness for $\gamma$-DSP in $\beta$-PL with $\gamma(k) \geq \frac{15}{11}\delta \cdot k$

In this section, we investigate the problem $\gamma$-DSP when restricting to power-law input graphs. More precisely, we determine a range of functions $\gamma$ where the problem is **NP**-complete.

From previous sections, we know that for arbitrary input graphs the problem is **NP**-complete, if $\gamma \in k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$ (see Theorem 4.4). Further we know that the general problem can be solved in polynomial time, if $\gamma \in k + O(1)$ (Corollary 4.1). Obviously, the bound for polynomial tractability of the problems also holds when restricting to power-law graphs, whereas this is not true when considering **NP**-completeness. Nevertheless, we can prove **NP**-hardness (resp. **NP**-completeness, since containment of $\gamma$-DSP in **NP** once again is easily seen) for a wide range of functions $\gamma$ (see Theorem 4.5). The complexity of $\gamma$-DSP for the remaining gap is discussed in 4.5.

**Theorem 4.5** *Let $\beta > 2$ be some fixed rational, and let $\delta$ be the maximum average degree of all $\beta$-PL graphs. Further let $\gamma : \mathbb{N} \to \mathbb{N}$ be a polynomial-time computable function, with $\gamma(k) \geq \frac{15}{11}\delta \cdot k)$.*

*The problem $\gamma$-DSP on $\beta$-PL graphs, i.e., given a tuple $(G, k)$ deciding whether $\beta$-PL graph $G$ contains some subgraph on exactly $k$ vertices with at least $\gamma(k)$ edges, is* **NP***-complete.*

For the proof we use similar techniques as we have done in section 4.3, when proving **NP**-completeness of $\gamma$-DSP for general input graphs. Once again, we use reduction from $\textsc{Clique}_{\frac{1}{2}}$. Within the proof, we assume that for every instances $(G, k) \in \textsc{Clique}_{\frac{1}{2}}$

- $G$ has minimum degree $k$, and
- $k \geq \max\{\, 4^{\beta} + 2, 2\hat{\delta}\,\}$, with $\hat{\delta} = \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} \right\rceil + 2$.

Obviously, this property has no impact on the **NP** completeness of the problem. For every input $(G, k)$, we construct a new graph $G' \in \beta$-PL in such a way that $G'$ has a subgraph on $k'$ vertices with at least $\gamma(k')$ edges if and only if graph $G$ has a clique of size $k$. Different to the proof of Theorem 4.4, we also use transformation $N$ and some further techniques.

### 4.4.1 Outline of the proof

In the **NP**-completeness proof for general graphs, it has been possible to build graphs with arbitrary degree sequence. Now, in order to guarantee that the final degree sequence obeys a power-law, we have to perform a more careful construction. Nevertheless, we can reuse some of the construction techniques of the previous proof, when building the graph $G'$:

- Firstly, we transform the input graph to some graph $G_1$ and add an additional graph $G_2$ (either disjointly or completely connected). Within the proof, once again, we show that this is done in such a way that there exists a densest subgraph on $k'$ vertices that includes all vertices of $G_2$.
- Secondly, different to the construction for general graphs, we add (disjointly) a third graph $G_3$ in order to guarantee that the final graph $G'$ is a $\beta$-PL graph. We will choose $G_3$ in such a way that any densest subgraph on $k'$ vertices of $G'$ does not contain vertices of $G_3$.

This overall idea is is illustrated in Figure 4.6 (the location of some densest subgraph in $G'$ on $k'$ vertices is highlighted).

Once again, we fix the value of $k'$ and use $\gamma(k')$ to distinguish five cases for the construction of the final graph $G'$. We choose $k'$ to be number of vertices of a $(N, \beta)$-PL-graph with degree at least two, where we leave out some appropriate number of vertices, dependent on the values of $k$ and $\beta$.

$$k' = \sum_{i=2}^{N^{\frac{1}{\beta}}} \lfloor N \cdot i^{-\beta} \rfloor \; - \; \sum_{i=\hat{\delta}+k}^{2\hat{\delta}+\frac{3}{2}k} \hat{\delta}(\tfrac{3}{2}k + \hat{\delta}) \; - \; \binom{\frac{3}{2}k+\hat{\delta}}{2}(4^{\beta}+2)\hat{\delta}$$

$$\text{with} \;\; \hat{\delta} = \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} \right\rceil + 2 \quad \text{and} \quad N = (2\hat{\delta}k)^{8} \cdot (2\hat{\delta} + 2k)^{\beta}$$

This non-trivial value of $k'$ is chosen in order to assures that, within all different cases, we can prove the above stated property that the final $\beta$-PL graph contains a subgraph on $k'$ vertices and at least $\gamma(k')$ edges, if and only if $G$ contains a clique of size $k$.

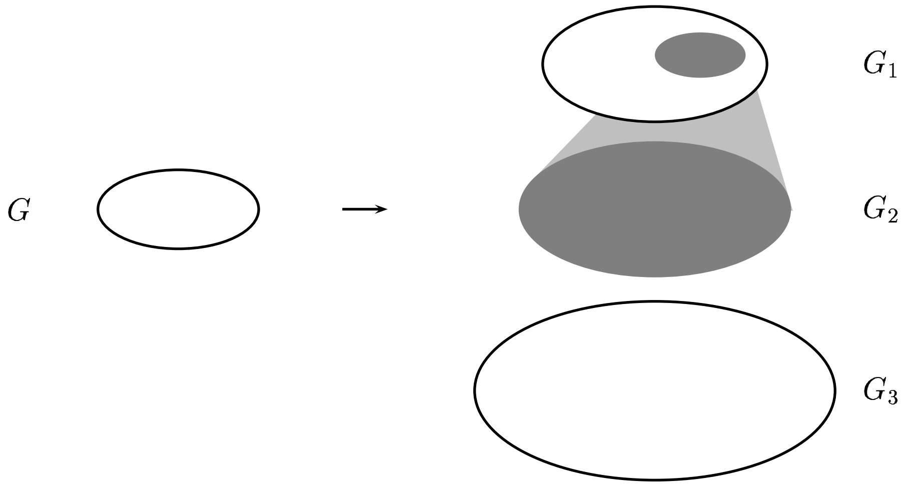

Figure 4.6: Selection of vertices for case I

Based on the choice of $k'$, for each of the five cases, as listed in Table 4.3, we can defined the exact ranges, which dependent on $\gamma(k')$. Further, the graphs $G_1$, $G_2$ and $G_3$ are defined as follows:

- Within all cases we define $G_1 = (N_x \circ S_t \circ R_s)(G)$, using appropriately chosen parameters $s$, $t$, and $x$.
- $G_2$ and $G_3$ are built as follows:
  - Case II – V: $G_2$ is constructed by applying operator $T$ to graph $G_1$, where graph $G_2$ is defined to be the added quasi regular graph. Further, graph $G_3$ is built according to Lemma 2.4.
  - Case I: The construction of $G_2$ and $G_3$ is more difficult and cannot be stated in terms of so far shown techniques (for a detailed description, see below).

Within the remaining part of the proof, each of the five cases is considered separately. Firstly, we show how to choose the parameters in order to build the final graph $G'$. Secondly, we prove that the above condition on the densest subgraph of $G'$ holds.

### 4.4.2 Case I: $\frac{15}{11}\delta \cdot k' \leq \gamma(k') < \hat{\delta}k'$

Before defining the detailed construction process of $G'$, we state a brief motivation. The essential idea of Case I is to build $G_1$ and $G_2$ in such a way that these graphs comprise almost all vertices of the final $\beta$-PL degree sequence that have degree of at least some appropriate threshold. Doing so, we can bound the

| | | | | | |
|---|---|---|---|---|---|
| Case I | $\frac{15}{11}\delta \cdot k'$ | $\leq$ | $\gamma(k')$ | $<$ | $\hat{\delta} \cdot k'$ |
| Case II | $\hat{\delta} \cdot k'$ | $\leq$ | $\gamma(k')$ | $<$ | $\hat{\delta} \cdot k'^{\frac{3}{2}}$ |
| Case III | $\hat{\delta} \cdot k'^{\frac{3}{2}}$ | $\leq$ | $\gamma(k')$ | $<$ | $\binom{k'}{2} - k'^{\frac{9}{8}}$ |
| Case IV | $\binom{k'}{2} - k'^{\frac{9}{8}}$ | $\leq$ | $\gamma(k')$ | $<$ | $\binom{k'}{2} - \frac{k'}{3}$ |
| Case V | $\binom{k'}{2} - \frac{k'}{3} \leq \gamma(k')$ | $\leq$ | $\gamma(k')$ | $\leq$ | $\binom{k'}{2}$ |

Table 4.3: Parameter settings for the **NP**-completeness proof of $\gamma$-DSP for $\beta$-PL

average degree of all subgraphs of $G_3$, and thus, we can guarantee the required condition for the densest subgraph in $G'$.

First of all, we determine the value $N$ to build the final $(N, \beta)$-PL degree sequence. In parallel, we construct a subset $X$ of the corresponding vertices in such a way that $|X| = k'$ and that the sum of degrees is $2\gamma(k')$, i.e., the vertices in $X$ induce exactly $\gamma(k')$ edges, when considering a graph on the vertices of $X$ (with corresponding degrees).

Based on the resulting set $X$ and the value of $N$, we construct the final graph $G'$. As stated above, we define $G_1 = (N_x \circ S_t \circ R_s)(G)$, with appropriately chosen parameters $s$, $t$, and $x$. In order to build $G_2$, we inter-connect the vertices in $X$ (regarding their assigned degrees) after removing a subset of vertices that corresponds to the vertices of a graph $(N_x \circ S_t \circ R_s)(K_k)$. Finally, we determine the number of vertices (with corresponding degrees) that miss to build the desired $(N, \beta)$-PL degree sequence. These vertices are also inter-connected, constituting graph $G_3$.

In the remaining discussion of Case I, we show how to choose $N$, $X$, as well as the remaining parameters, and how to construct the two graphs $G_2$ and $G_3$. Further, we prove the desired property that the densest subgraph on $k'$ vertices of $\tilde{G} = G_1 \uplus G_2$ contains all vertices of $V(G_2)$. Finally, we conclude that $G' = \tilde{G} \uplus G_3$ contains a subgraph on $k'$ vertices and at least $\gamma(k')$ edges if and only if $G$ contains a clique of size $k$.

**Parameter Selection** In the following, we state the process of how to choose $N$ and the set $X$. First of all we construct some starting set $X$ that has the required size $k'$ but where the sum of degrees is too small. Then we will refine the choice of $X$ in order to adjust the sum of degrees to the desired value of $2\gamma(k')$.

The construction of set $X$ is based on the following property that holds for the whole construction process. Let $x'$ be the minimal degree within $X$ (initially $x' = 2$). At any time, it must be possible to:

- build the graph $(N_{x'-1} \circ S_{\lfloor 4^\beta+2 \rfloor} \circ R_{\hat\delta})(G)$, assuming $x'-1 \leq \hat\delta$, by using only non-selected vertices of the actual $(N,\beta)$-PL degree sequence.
- remove vertices from $X$ that correspond to a graph $(N_{x'-1} \circ S_{\lfloor 4^\beta+2 \rfloor} \circ R_{\hat\delta})(K_k)$.

In order to guarantee the existence of the required sets of selected and unselected vertices, we use the following sufficient properties.

1. There exist
   at least $\hat\delta \cdot (4^\beta+2)\binom{\frac{3}{2}k+\hat\delta}{2}$ non-selected vertices, and
   at least $(x'-1) \cdot (4^\beta+2)\binom{k+\hat\delta}{2}$ selected vertices
   with degree $x'$.
2. There exist
   at least $\hat\delta \cdot (\frac{3}{2}k+\hat\delta)^2$ non-selected vertices , and
   at least $\hat\delta \cdot (k+\hat\delta)$ selected vertices
   for all degrees in $[\,\hat\delta + k \,..\, 2\hat\delta + \frac{3}{2}k\,]$

Now, we can state how to define the initial choices of $N$ and $X$ initially. We define

$$N = (2\hat\delta k)^8 \cdot (2\hat\delta + 2k)^\beta \qquad \text{with } \hat\delta = \left\lceil 4 \cdot 2^{\frac{1}{\beta-1}} \right\rceil + 2.$$

and $X$ as the set comprising all vertices of the $(N,\beta)$-PL degree sequence and with degree at least two, where we leave out the following sets of vertices:

- $\hat\delta(4^\beta+2)\binom{\frac{3}{2}k+\hat\delta}{2}$ vertices of degree 2, and
- $\hat\delta(\frac{3}{2}k+\hat\delta)^2$ vertices for each degree in $[\,\hat\delta + k \,..\, 2\hat\delta + \frac{3}{2}k\,]$

Obviously, $X$ has cardinality $k'$ and matches the required properties. Further, we can see that $X$ contains no vertices of degree 1, and that there miss more vertices with degree at least $2\delta$ than vertices with degree in $[\,2 \,..\, \hat\delta\,]$, w.r.t. the $(N,\beta)$-PL degree sequence. The last condition is true since

$$\hat\delta \cdot (4^\beta+2) \cdot \binom{\frac{3}{2}k+\hat\delta}{2} \quad \leq \quad \frac{3}{2}k \cdot \hat\delta \left(\frac{3}{2}k+\hat\delta\right)^2$$

holds for $k \geq \max\{4^\beta+2, 2\hat\delta\}$.Therefore, the degree sequence corresponding to set $X$ satisfies all conditions of Lemma 2.2. Applying this lemma, we derive that the average degree of the vertices in $X$ is less than $\frac{30}{11}$ of the average degree of any $\beta$-PL degree sequence, or, equivalently that the number of edges that correspond to the sum of degrees in $X$ is at most $\gamma(k')$, since $\gamma(k') \geq \frac{15}{11}\delta \cdot k'$.

In the following, using these initial definitions of $N$ and $X$, we alter the value $N$ and the choice of $X$ in such a way that the sum of the degrees corresponds to exactly $\gamma(k')$ edges, while the number of selected vertices remains constant. More precisely, we iteratively exchange a vertex $v \in X$ with a vertex $w \notin X$, where $\deg(w) = \deg(v) + 1$ (or equal-sized sets of vertices with equivalent degree balance). Doing so, we increase the sum of degree in $X$ step by step and finally end up with the desired sum of degrees. In order to guarantee that the required properties of $X$ hold, we proceeded as follows:

- Due to the first property, we exchange the last $(x'-1) \cdot (4^\beta + 2)\binom{k+\hat{\delta}}{2}$ selected vertices of degree $x'$ at once. Since the sum of degrees must increase by exactly one, we also exchange an appropriate number of vertices with degree $x' + 2$ with unselected vertices of degree $x' + 1$.

- To assure the second property, we can leave out the necessary number of vertices throughout the whole process.

- Further, satisfying these two conditions, we always try to exchange vertices with degree $x'$. If this is not possible, due to the above restrictions, we use vertices with highest possible degree. At some some point, it may happen that no further exchanges are possible, while the sum of selected degrees is still less than $2\gamma(k')$. In such a case, we increase $N$ by $(x'+1)^\beta$. Doing so, due to
$$\lfloor (N + (x'+1)^\beta)(x'+1)^{-\beta} \rfloor = \lfloor N(x'+1)^{-\beta} \rfloor + 1$$
the number of vertices with degree $x' + 1$ is increased by one. For other degrees greater than $x'$ the number of vertices can also increase by one. Obviously, the number of all these additional vertices is bounded by $N^{\frac{1}{\beta}}$ (maximum degree).

This process of exchanging vertices and increasing $N$, respectively, can be iterated until finally the sum of degrees equals $2\gamma(k')$. We will use the corresponding value of $N$ and the set $X$ to construct the graph $G'$. Since $\gamma(k') < \frac{1}{2}\hat{\delta}k'$, it holds that $x' < \hat{\delta}$.

The structure of the resulting set $X$ is high-lighted in Figure 4.7 (at the left boundary of the filled region, it is illustrated that there may miss several vertices of degree $x'$ and $x' + 1$).

Before stating the construction of $G'$, we argue that the sum of degrees of the unselected vertices with degree greater than $x'$ can be bounded by $0.5N$. Obviously, there are at most

- $2(x'-1)(4^\beta + 2)\binom{k+\hat{\delta}}{2} + \hat{\delta}(4^\beta + 2)\binom{\frac{3}{2}k+\hat{\delta}}{2}$ unselected vertices of degree $x' + 1$,

- $\hat{\delta}(\frac{3}{2}k + \hat{\delta})$ unselected vertices for each degree in $[\,\hat{\delta} + k \;..\; 2\hat{\delta} + \frac{3}{2}k\,]$, and

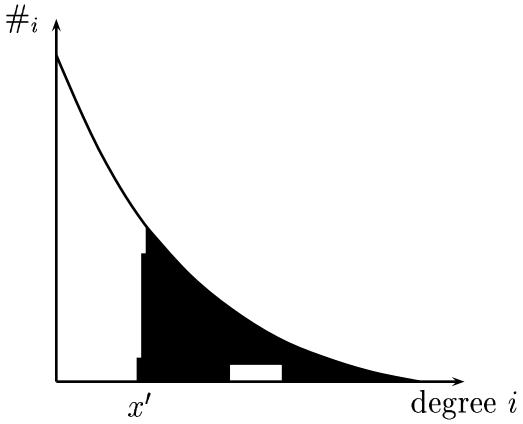

Figure 4.7: Exemplary selection of vertices for Case I of the **NP**-completeness proof of $\gamma$-DSP in $\beta$-PL

- a set of at most $N^{\frac{1}{\beta}}$ additional unselected vertices with degree at least $x'+1$, where the sum of degrees is bounded by $\sum_{i=x'+1}^{N^{\frac{1}{\beta}}} i$.

Based on the following three inequalities (assuming $k$ and thus $N$ large enough)

$$\begin{aligned} 2(x'-1)(4^\beta+2)\binom{k+\hat{\delta}}{2} + \hat{\delta}(4^\beta+2)\binom{\frac{3}{2}k+\hat{\delta}}{2} &\leq 3\hat{\delta}(4^\beta+2)2k^2\hat{\delta}^2 \leq \frac{1}{100\hat{\delta}}N \\ \hat{\delta}(\frac{3}{2}k+\hat{\delta}) &\leq 2k\hat{\delta}^2 \\ \sum_{i=x'+1}^{N^{\frac{1}{\beta}}} i &\leq \frac{1}{2}(N^{\frac{1}{\beta}}+1)N^{\frac{1}{\beta}} \end{aligned}$$

we can bound the sum of degrees of the unselected vertices by

$$\hat{\delta} \cdot \frac{1}{100\hat{\delta}}N \;+\; 2k\hat{\delta}^2 \cdot (2\hat{\delta} + \frac{3}{2}k) \;+\; \frac{1}{2}(N^{\frac{1}{\beta}}+1)N^{\frac{1}{\beta}} \;\leq\; 0.5N.$$

**Construction of G'** Using the final choices of $N$ and $X$, we can describe how to build graph $G' = G_1 \uplus G_2 \uplus G_3$. The three graphs are define as follows.

- The graph $G_1 = (N_x \circ S_t \circ R_s)(G)$ is constructed with the following parameters:

  $$\boxed{s = \hat{\delta} \qquad t = \lfloor 4^\beta + 2 \rfloor \qquad x = x' - 1}$$

  From above we know that $2 \leq x' \leq \hat{\delta}$ and thus $1 \leq x < \hat{\delta}$. Further, we know that $G$ has minimum degree at least $k$ and maximum degree at most

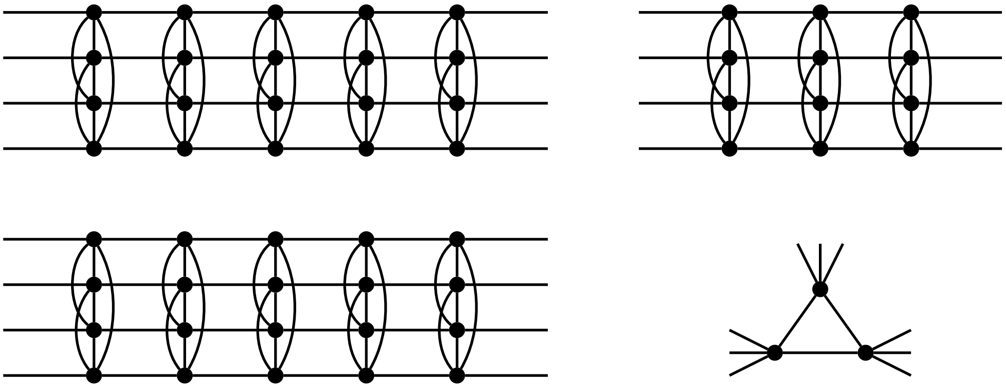

Figure 4.8: Construction of $G'$ for $\beta$-PL (Part 1)

$\frac{3}{2}k - 1$. Due to the properties of $X$, it is easy to see that we can build $G_1$ by using vertices of the $(N, \beta)$-PL degree sequence that are not contained in $X$.
Using lemma 3.2, we know that $G_1$ has a subgraph with $x\left(k+s+t\binom{k+s}{2}\right)$ vertices and at least $\binom{x}{2}\left(k+s+t\binom{k+s}{2}\right) + x(t+1)\binom{k+s}{2}$ edges if and only if $G$ contains a clique of size $k$.

- Now, we describe how to build $G_2$. This rather complex construction is required in the latter proof on exchanging vertices within $G'$. The set of vertices $V(G_2)$ corresponds to the set $X$ after removing a set of vertices that are required to build a graph $(N_x \circ S_t \circ R_s)(K_k)$ (i.e., $x \cdot (\hat{\delta} + k)$ vertices with degree $(s+k-1)+(x-1)$ and $x \cdot t\binom{s+k}{2}$ vertices with degree $x+1 = x'$). Due to the choice of $X$ and $N$, it is always possible to remove that set of vertices.
  In the following, we state how to inter-connect the remaining vertices regarding their assigned degrees. The resulting graph is assigned to $G_2$.

  1. All vertices of degree $x'$ are grouped to sets of $x = x' - 1$ vertices. We use these sets to build graphs isomorphic to graphs that result when applying transformation $N_x \circ S_t$ to a single edge ($x$ and $t$ are chosen as in construction of $G_1$). At each end of these chains, every vertex remains with one so far unlinked connection. The remaining sets are connected to build some additional chain (with length less than $t$). Further, all vertices (at most $i < x$) that have not been assigned to some set, are inter-connected as dense as possible (i.e., we build some graph $K_i$, where every vertex has $x' - i$ additional unlinked connections). For $55 = 20 + 20 + 12 + 3$ vertices, $t = 5$, and $x = 4$ this process is illustrated in Figure 4.8.
  2. In order to link the vertices of degree $x'+1$ we build some quasi-regular graph $A(n', \frac{1}{2}(x'+1)n')$, where $n'$ equals to the number of vertices with

degree $x' + 1$ (if the sum of degrees is odd, we leave out one of the vertices and link its connections similar to the open connections of vertices with degree $x'$). Due to the large number of vertices with degree $x' + 1$, the number of required edges is less than $\binom{n'}{2}$. This guarantees the existence of the graph.

3. The remaining vertices (degree $i \geq x' + 2$)) are connected as follows. For every vertex, we guarantee that at least $x'+2$ of its outgoing edges connect to other vertices with degree at least $x' + 2$. The remaining $i - (x' + 2)$ edges can lead to arbitrary vertices within $X$, i.e. also vertices with degree $x'$ chosen as follows:

   First, we satisfy all remaining open connections of the vertices with degree $x'$ (and $x'+1$). In order to do so, we show that there are enough suitable connections within the vertices with degree at least $x + 2$. It suffices to show:

$$\frac{\lfloor N \cdot x'^{-\beta} \rfloor}{(\lfloor 4^{\beta} + 2 \rfloor)x'} + 4x' + \binom{x'}{2} \leq \sum_{i=x'+2}^{N^{\frac{1}{\beta}}} (i - (x' + 2)) \left(\lfloor N \cdot i^{-\beta} \rfloor - 1\right) - \sum_{i=\hat{\delta}+k}^{2\hat{\delta}+\frac{3}{2}k} i \cdot \hat{\delta}(\frac{3}{2}k + \hat{\delta})$$

which can easily seen to be true since

$$\begin{aligned}
\frac{\lfloor N \cdot x'^{-\beta} \rfloor}{(\lfloor 4^{\beta} + 2 \rfloor)x'} + 4x' + \binom{x'}{2} &\leq N \cdot (4x')^{-\beta} + 4\hat{\delta}^2 \\
&\leq N \cdot (x' + 3)^{-\beta} + 4 \cdot N(x' + 6)^{-\beta} \\
&\leq \sum_{i=x'+3}^{x'+6} (i - (x' + 2)) \left(\lfloor N \cdot i^{-\beta} \rfloor - 1\right) \\
&\leq \sum_{i=x'+3}^{N^{\frac{1}{\beta}}} (i - (x' + 2)) \left(\lfloor N \cdot i^{-\beta} \rfloor - 1\right) - \sum_{i=\hat{\delta}+k}^{2\hat{\delta}+\frac{3}{2}k} i \cdot \hat{\delta}(\frac{3}{2}k + \hat{\delta})
\end{aligned}$$

   Since every vertex with degree $x'$ has only one open connection (up to a small number of exceptions) we can guarantee that not two vertices are linked twice.

   After satisfying the demand of vertices with degree $x'$ (and $x' + 1$) we continue with the vertices with degree greater than $x' + 2$. We group the vertices according to the number of missing links. Starting with those vertices $S$ with highest number $d$ of missing links we build regular graphs $A(n', m')$, for appropriate values of $n'$ and $m'$. If the number of vertices is too small or the sum of degrees is odd, we additionally use vertices of the next level(s). However, in the end we will be able to satisfy all links, due to the large number of vertices in $X$ and the property that the sum of degrees is even.

- Finally, we construct graph $G_3$. Based on the choice of $N$, $\beta$, $G_1$, and $G_2$ we can derive the number of vertices (with corresponding degrees) that miss to build the overall $(N, \beta)$-PL degree sequence. First of all, we will link all vertices with degree greater than $x'$ the vertices with degree 1. Using the above calculation, we know that this can be done with at most $0.5N$ vertices of degree 1. For the remaining vertices with degree $i$ we build chains corresponding to the outcome of a transformation $(N_i \circ S_{t'})$, with $t'$ as large as possible. At each end of the chain there remain $i$ unsatisfied connections. Further, for every degree $i$ there remain at most $i-1$ unmatched vertices corresponding to $i(i-1)$ unsatisfied connections. All in all, for all degrees $i \in [\,2 \,..\, \hat{\delta}\,]$, there remain at most $\sum_{i=2}^{\hat{\delta}} i(i+1) < 2\hat{\delta}^3$ open connections which are satisfied with vertices of degree 1. Since the total number of vertices with degree 1 equals $N$, the number of these vertices is large enough to satisfy all required corresponding connections. The remaining vertices of degree 1 are connected pairwise. Once again, due to the choice of the even sum of degrees, there remain no unmatched connections.

Using these three graphs we can construct graph $G' = G_1 \uplus G_2 \uplus G_3$ that obviously has a $(N, \beta)$-PL degree sequence.

**Densest subgraph on $k'$ vertices** In the following part of the proof we show that $G'$ contains a subgraph of $k'$ vertices with at least $\gamma(k')$ edges if and only if $G$ contains a clique of size $k$.

To do so, let $Y$ be a set of $k'$ vertices of $G'$, and $Y_1$, $Y_2$, and $Y_3$ be the corresponding subsets of vertices in the graphs $G_1$, $G_2$, and $G_3$. We state a process to exchange vertices in such a way that finally all vertices of $G_2$ (i.e., $Y_2 = V(G_2)$) and no vertices of $G_3$ (i.e., $Y_3 = \emptyset$) are selected without decreasing the induced number of edge. We consider two cases:

- $|Y_3| \leq |V(G_2)| - |Y_2|$: First of all, we replace all vertices in $Y_3$ with the unselected vertices in $V(G_2)$ (note that the average induced degree in $Y_3$ is less than minimum degree in $G_2$). To select the remaining unselected vertices in $V(G_2)$ we exchange vertices from $Y_1$. We can choose the vertices from $Y_1$ in such a way that we start with the inner cliques of chains in $G_1$. Before removing an outer clique, we will remove all inner cliques connected to it. Doing so, we can observe that the loss in the sum of induced degrees in $Y_1$ is less than the increase in $G_2$ (due to the lengths of chains and the rules of connections in $G_2$). All in all we can see that the number of induced edges does not decrease.

- $|Y_3| > |V(G_2)| - |Y_2|$: We replace $|V(G_2)| - |Y_2|$ vertices of $|Y_3|$ vertices with the remaining unselected vertices in $V(G_2)$. As mentioned above the sum of induced degrees cannot decrease. The remaining vertices in $Y_3$ are replaced with unselected vertices in $V(G_1)$. W.l.o.g., we can assume that the vertices in $Y_1$ are initially chosen in such a way that at most one of the cliques introduced by operator $N_x$ is partially selected (see Lemma 3.2). Further, due to $s = \hat{\delta}$, we know that $G_1$ is connected. Therefore, it is possible to exchange the remaining vertices of $Y_3$ with unselected vertices in $G_1$ in such a way that all selected vertices in $G_1$ belong to completely selected cliques. Once again, we can conclude that the number of overall induced edges does not decrease.

Finally we know that all vertices of $V(G_2)$ and $x\left(k+s+t\binom{k+s}{2}\right)$ vertices of $V(G_1)$ are selected. The number of induced edges is at least the number of induced edges of the initial set $Y$ (this also holds if $Y$ originally induced some densest subgraph on $k'$ vertices). Obviously, the number of edges in the final subgraph is equal to the number of edges in $G_2$ plus the number of induced edges in $G_1$. Therefore, in order to induce $\gamma(k')$ in $G'$ the selected vertices in $G_1$ must induce at least $\binom{x}{2}\left(k+s+t\binom{k+s}{2}\right) + x(t+1)\binom{k+s}{2}$ edges.

All in all, the existence of a subgraph of $G'$ on exactly $k'$ vertices and at least $\gamma(k')$ edges is equivalent to the existence of a subgraph of $G_1$ on exactly $k+s+t\binom{k+s}{2}$ vertices and at least $\binom{x}{2}\left(k+s+t\binom{k+s}{2}\right) + x(t+1)\binom{k+s}{2}$ edges. According to Lemma 3.2 the last proposition is equivalent to the existence of a clique of size $k$ in $G$. Thus, for Case I, we have proven that $G'$ contains a subgraph on $k'$ vertices with at least $\gamma(k')$ edges if and only if $G$ contains a clique of size $k$.

### 4.4.3 Cases II–V: $\hat{\delta}k' \leq \gamma(k') < \binom{k'}{2}$

Within Cases II-V, the construction of $G'$ and the structure of the proof follows some common pattern. Therefore, we describe the general proof technique before consider each case on its own. We define $G' = \tilde{G} \uplus G_3$, using the following definitions:

- $G_1 = (N_x \circ S_t \circ R_s)(G)$. To assure the constructibility of $G_1$, we have to guarantee $x \geq 1$, $t \geq 0$, and $s \geq 0$, for each case.
- $\tilde{G} = T^{\alpha}_{r,N(r)}(G_1)$, with the following parameters:

$$\begin{aligned} r &= k' - x\left(k+s+t\binom{k+s}{2}\right) \\ N(r) &= \gamma(k') - \binom{x}{2}\left(k+s+t\binom{k+s}{2}\right) - x(t+1)\binom{k+s}{2} \\ &\quad - \alpha \cdot r \cdot x\left(k+s+t\binom{k+s}{2}\right) \end{aligned}$$

Graph $G_2$ that has defined in the overall description of the proof, is considered to be the quasi-regular graph $A(r, N(r)$ added by transformation $T$.

In order to guarantee the existence of $\tilde{G}$, similar to the **NP**-completeness proof of $\gamma$-DSP on general graphs, we have to prove the property *Constructibility* of $A(r, N(r))$, i.e., $0 \leq N(r) \leq \binom{r}{2}$, where the first inequality is often shown combined with some second property (*Exchangeability*, see below).

- For the construction of $G_3$ we use $\beta$ and $N = (2\hat{\delta}k)^8 \cdot (2\hat{\delta} + 2k)^{\beta}$. Based on the graph $\tilde{G}$ we can determine the number of missing vertices (and corresponding degrees) to build the desired $(N, \beta)$-PL graph. Using Lemma 2.4, we can inter-connect these vertices and build $G_3$ in such a way that any subgraph has average degree at most $\hat{\delta} - 1$.

For each case, besides Constructibility (defined above), we show that $G'$ contains a subgraph on $k'$ vertices with at least $\gamma(k')$ edges if and only if $G$ contains a clique of size $k$.

Similar to Case I, let $Y \subseteq V(G')$ be a set of vertices that induced some densest subgraph of $G'$ on $k'$ vertices. Further, let $Y_1$, $Y_2$, and $Y_3$ be the corresponding sets of vertices in $G_1$, $G_2$, and $G_3$. Once again, we exchange vertices in such a way that all vertices in $G_2$ and no vertices in $G_3$ are selected, without decreasing the number edges, by using similar arguments as in Case I. Let $G_Y$, $G_{Y_1}$, and $G_{Y_1}$ be the graphs induced by the finally resulting sets $Y$, $Y_1$, and $Y_2$. Obviously, the number of edges $|E(G_Y)|$ is equivalent to the maximum number of induced edges on all subgraphs of $G'$ on $k'$ vertices and it holds:

$$\begin{aligned} |E(G_Y)| &= |E(G_{Y_1})| + |E(G_{Y_2})| + \alpha \cdot r \cdot x \left(k + s + t\binom{k+s}{2}\right) \\ &= \gamma(k') - \binom{x}{2}\left(k + s + t\binom{k+s}{2}\right) - x(t+1)\binom{k+s}{2} + |E(G_{Y_2})|. \end{aligned}$$

Thus the number of edges in $G_Y$ is at least $\gamma(k')$ if

$$|E(G_{Y_1})| \geq \binom{x}{2}\left(k + s + t\binom{k+s}{2}\right) + x(t+1)\binom{k+s}{2}.$$

Using $|Y_1| = k' - r = x\left(k + s + t\binom{k+s}{2}\right)$, Lemma 3.2, and assuming that it is possible to exchange the vertices as desired, we can conclude that $G'$ contains a subgraph on $k'$ vertices with at least $\gamma(k')$ edges iff $G$ contains a clique of size $k$. Therefore, besides Constructibility of $G_1$, it remains to show how to exchange the vertices in order to assure the properties assumed above.

Similar to Case I we exchange all vertices of $Y_3$ with unselected vertices in $V(G_2)$. If all vertices in $V(G_2)$ get selected, we can continue to exchange the remaining vertices in $Y_3$ with unselected vertices in $V(G_1)$. Otherwise, if $V(G_2)$ is not selected entirely, we choose the remaining set of required vertices from $Y_1$. Finally, all vertices in $V(G_2)$ and $k' - |V(G_2)|$ vertices in $G_1$ are selected.

In order to prove the theorem we have to show that the number of induced edges does not decrease. This can be done as follows:

- *Exchangeability 1*: For the exchange of vertices from $Y_3$ to vertices in $V(G_1)$ and $V(G_2)$, we show that the average induced degree of the new vertices finally is at least $\hat{\delta} - 1$. According to Lemma 2.4 we know that $\hat{\delta} - 1$ is an upper bound for the average induced degree of $Y_3$.

- *Exchangeability 2*: For the exchange of vertices from $Y_1$ to vertices in $V(G_2)$, we proceed similar to the proof for general input graphs (see proof of Theorem 4.4). Assume that we exchange $z > 0$ vertices. For the case $\alpha = 0$ we will remove at most $\Delta z$ edges from $G_1$, for some $\Delta$. Therefore, after adding vertices to $Y_2$, the sum of the induced degrees must increase by at least $2\Delta z$. Thus, it is sufficient to satisfy that the minimum (induced) degree of the (exchanged) vertices in $G_2$ is large enough i.e. $\lfloor \frac{2N(r)}{r} \rfloor z \geq 2\Delta z$ or, if $2\Delta \in \mathbb{N}$, equivalently, $\frac{N(r)}{r} \geq \Delta$. In the case, if $\alpha = 1$, we will employ more refined arguments stated in the corresponding cases.

In the remaining part of the proof, we show that the properties Constructibility and Exchangeability 1 and 2 hold for all four cases.

**Case II:** $\hat{\delta} k' \leq \gamma(k') < \hat{\delta} k'^{\frac{3}{2}}$

| | |
|---|---|
| $s = 0$ | $t = 1$ |
| $x = \hat{\delta} - 1$ | $\alpha = 0$ |
| $r = k' - x\left(k + \binom{k}{2}\right)$ | $N(r) = \gamma(k') - \binom{x}{2}\left(k + \binom{k}{2}\right) - 2x\binom{k}{2}$ |

- *Exchangeability 1*: W.l.o.g., we can assume that the vertices in $Y_1$ (both, before and after exchanging vertices) are chosen similar to the final graph within the proof of Lemma 3.2 (i.e., almost all cliques are chosen completely and have at least one selected neighboring clique). Therefore we know that, in the end, all new vertices have induced degree at least $(x - 1) + 1 = \hat{\delta} - 1$. All vertices in $V(G_2)$ have degree at least $\lfloor 2\hat{\delta} \rfloor > \delta - 1$ (see Exchangeability 2).

- *Exchangeability 2*: Due to $t = 1$ we can set $\Delta = (x - 1) + 2 = \hat{\delta}$. Since $2\Delta \in \mathbb{N}$, it remains to prove $N(r) \geq \hat{\delta} r$.

$$
\begin{aligned}
N(r) &= \gamma(k') - \binom{x}{2}\left(k + \binom{k}{2}\right) - 2x\binom{k}{2} \\
&\geq \hat{\delta}k' - \binom{x}{2}\left(k + \binom{k}{2}\right) - 2x\binom{k}{2} \\
&= \hat{\delta}\left(r + x\left(k + \binom{k}{2}\right)\right) - \binom{x}{2}\left(k + \binom{k}{2}\right) - 2x\binom{k}{2} \\
&= \hat{\delta}r + (\hat{\delta} - 2)x\left(k + \binom{k}{2}\right) - \frac{(x-1)}{2}x\left(k + \binom{k}{2}\right) + \\
&\qquad 2\,x\left(k + \binom{k}{2}\right) - 2x\binom{k}{2} \\
&= \hat{\delta}r + \frac{3}{2}(\hat{\delta} - 2)x\binom{k+1}{2} + 2xk \\
&\geq \hat{\delta}r
\end{aligned}
$$

- *Constructibility*: We have to prove $0 \leq N(r) \leq \binom{k}{2}$. While the lower bound follows directly from Exchangeability 2, the upper bound holds for large enough values of $k$ as the follows:

$$
\begin{aligned}
N(r) &= \gamma(k') - \binom{x}{2}\left(k + \binom{k}{2}\right) - 2x\binom{k}{2} \\
&\leq \hat{\delta}k'^{\frac{3}{2}} = \hat{\delta}((\hat{\delta}k)^8)^{\frac{3}{2}} \\
&= \hat{\delta}^{13}k^{12} = \frac{1}{\hat{\delta}k^2}\hat{\delta}^{14}k^{14} = \frac{1}{\hat{\delta}k^2}((\hat{\delta}k)^7)^2 \\
&\leq \binom{(\hat{\delta}k)^8 - x\left(k + \binom{k}{2}\right)}{2} \leq \binom{r}{2}
\end{aligned}
$$

**Case III:** $\hat{\delta}k'^{\frac{3}{2}} \leq \gamma(k') < \binom{k'}{2} - k'^{\frac{9}{8}}$

| | |
|---|---|
| $s = \hat{\delta}$ | $t = 0$ |
| $x = 1$ | $\alpha = 0$ |
| $r = k' - k - s$ | $N(r) = \gamma(k') - \binom{k+s}{2}$ |

- *Exchangeability 1*: Due to the choice of the parameters we have $G_1 = R_s(G)$. Therefore, the exchange of vertices is possible in such a way that all new vertices in $Y_1$ have degree at least $s - 1 = \hat{\delta} - 1$. Once again, it follows from Exchangeability 2 that the degree of all vertices in $G_2$ is greater than $\hat{\delta} - 1$.

- *Exchangeability 2*: Since $|V(G)| \leq \frac{3}{2}k$ (using definition of $\text{Clique}_{\frac{1}{2}}$) we can set $\Delta = \frac{3}{2}k + s = \frac{3}{2}k + \hat{\delta}$ (once again $2\Delta \in \mathbb{N}$). Thus, we have to show $N(r) \geq (\frac{3}{2}k + \hat{\delta})r$. Based on $k' \geq (\hat{\delta}k)^8$ and $r = k' - (k+s)$ we get $r \geq \binom{k+s}{2}$ and $(k+r)^{\frac{1}{2}} \geq k^4$. Using these two inequalities if follows:

$$\begin{aligned} N(r) &= \gamma(r) - \binom{k+s}{2} \geq \hat{\delta} \cdot k'^{\frac{3}{2}} - \binom{k+s}{2} \\ &\geq (\hat{\delta}(k+r)^{\frac{1}{2}} - 1) \cdot r \\ &\geq (\hat{\delta}k^4 - 1) \cdot r \geq (\frac{3}{2}k + \hat{\delta})r \end{aligned}$$

- *Constructibility*: Once again, we show $0 \leq N(r) \leq \binom{r}{2}$. The lower bound follows from Exchangeability 2. Using the below calculation we can prove the upper bound:

$$\begin{aligned} N(r) &= \gamma(r) - \binom{k+s}{2} &&\leq \binom{r+k+s}{2} - k'^{\frac{9}{8}} - \binom{k+s}{2} \\ &= \binom{r}{2} + r \cdot (k+s) - k'^{\frac{9}{8}} &&\leq \binom{r}{2} + k' \cdot \left((k+s) - k'^{\frac{1}{8}}\right) \\ &= \binom{r}{2} + k' \cdot \left(k + \hat{\delta} - \hat{\delta}k\right) &&\leq \binom{r}{2} \end{aligned}$$

**Case IV:** $\binom{k'}{2} - k'^{\frac{9}{8}} \leq \gamma(k') < \binom{k'}{2} - \frac{k'}{3}$

| | |
|---|---|
| $s = \lceil \frac{1}{3}k'^{\frac{1}{4}} \rceil - k$ | $t = 1$ |
| $x = 1$ | $\alpha = 1$ |
| $r = k' - (k+s) - \binom{k+s}{2}$ | $N(r) = \gamma(k') - \left((k+s) + \binom{k+s}{2}\right) r - 2\binom{k+s}{2}$ |
| | $= \gamma(k') - \binom{k+s+1}{2} r - 2\binom{k+s}{2}$ |

Due to $x = 1$ the operator $N_x$ does not modify the graph. All other parameters are similar to Case IV in the proof for general graph. Therfore, Exchangeability 2 and Constructibility can be proven using equivalent calculations. It remains to prove Exchangeability 1.

- *Exchangeability 1:* Due to $\alpha = 1$ and the large values for $r$ and $s$ the induced degrees of the exchanged vertices are obviously greater than $\hat{\delta} - 1$.

**Case V:** $\binom{k'}{2} - \frac{k'}{3} \leq \gamma(k') \leq \binom{k'}{2}$

| | |
|---|---|
| $s = 0$ | $t = 0$ |
| $x = 1$ | $\alpha = 1$ |
| $r = k' - k$ | $N(r) = \gamma(k') - kr - \binom{k}{2}$ |

Once again, $x = 1$, while all other parameters are equivalent to Case V for the proof for general graphs. Thus, once again, we can restrict to prove Exchangeability 1.

- *Exchangeability:* From Constructibility we know that $N(r) \geq (\hat{\delta} - 1)r$. Therefore, the induced degree of every new vertex in $Y_2$ is at least $\hat{\delta} - 1$. When we have to exchange vertices from $Y_3$ to $V(G_1)$, we know that every new vertex has an induced edge to all vertices in $V(G_2)$. Consequently, the induced degree is at least $r = k' - k \geq \hat{\delta} - 1$.

With the proof of Exchangeability 1/2 and Constructibility for Case II – V we have completed the proof of Theorem 4.5.

## 4.5 Discussion of the upper and lower bounds for the complexity of $\gamma$-DSP on $\beta$-PL graphs

Within the preceeding sections (sections 4.2 to 4.4), we have derived bounds for function $\gamma$ guaranteeing that the decision problem $\gamma$-DSP is either decidable in polynomial time or **NP**-complete. While, for general input graphs, the lower bound for **NP**-completeness is very close to the upper bound for polynomial-time tractability, this is not the case when restricting to $\beta$-PL input graphs. In this section, we discuss the remaining gap for this restricted problem:

1. We argue that the upper bound $k+O(1)$ for polynomial time solvability and the lower bound of $k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$, for **NP**-completeness of $\gamma$-DSP on general graphs are also good corresponding bounds for the restricted problem.
2. We analyze the reduction technique that has been used to derive the lower bound for completeness in **NP**. We show that using this technique, we cannot improve the lower bound, unless further properties of the underlying power-law graphs are known.

We propose that, similar to the general case of $\gamma$-DSP, the restricted problem is **NP**-complete for most of the remaining functions $\gamma$. As already mentioned above, the main difficulties in order to improve the corresponding bounds strongly depend on the following unknown properties for $\beta$-PL graphs.

- There exists no construction for graphs on $\beta$-PL degree subsequences (matching requirements similar to those of graph $G_3$ within Cases II–V of the **NP**-completeness proof) with small upper bound for the average degree of subgraphs. An approach to this question has been initiated in Lemma 2.4, where we have been able to shown a possible construction for maximum average degree $\delta_{\max} = 4 \cdot 2^{\frac{1}{\beta-2}} + 1$. Despite not relying on this bound within our proof, we propose the existence of better bounds which lead to some stricter lower-bound for **NP**-completeness.

- It is still open whether there exists a threshold $\delta_{\min} > 2$, only dependent on $\beta$, in such a way that every $(N, \beta)$-PL graph has a subgraph (trivial solution) on exactly $k$ vertices with average degree at least $\delta_{\min}$, for all $k \in [\, \delta_{\min} + 1 \,..\, N \,]$.

In either of these two cases, we possibly could improve our results for the so far best known bounds for the membership of $\gamma$-DSP in **P** resp. **NP**-c.

### 4.5.1 General discussion of the bounds based on $\gamma$-DSP on general graphs

In this subsection, we give evidence that the derived bounds for the complexity of $\gamma$-DSP on general graphs also hold for $\beta$-PL input graphs.

The polynomial-time algorithm for general input graphs (see section 4.2), is based on testing all appropriate sets (i.e., sets of cardinality at most two times the desired excess) on vertices with degree greater two. When considering $\beta$-PL graphs, we have detailed information on the degree distribution and thus may improve the corresponding upper bound for polynomial solvability. Nevertheless, since the fraction of vertices with degree greater two (in $\beta$-PL graphs) is at least $\frac{3-\beta}{2}$, the algorithm still has exponential runtime for values of excess larger than any constant (i.e., $\gamma \in \omega(1)$). Therefore, this algorithm does not improve the derived upper bound for containment in **P**.

We even do not believe that it is possible to significantly increase the upper bound unless either there always exists some dense enough subgraph (trivial solution)[3] or $k$ is restricted to values $k > \frac{1}{2}|V|$ (in such a case, due to the large number of vertices with degree one, it is possible to test on the existence of a desired subgraph in polynomial time). This assumption is based on the following discussion arguing on the existence of a small lower-bound for **NP**-completeness of the problem, that is consistent with the bound for polynomial tractability of the problem.

Any graph $G$ on $k \in o(n^{\frac{1}{\beta+1}})$ vertices can be embedded in a $\beta$-PL graph $G'$ on $n$ vertices (i.e., it is possible to construct $G'$ in such a way that $G$ is isomorph to

[3] We are aware of the result of Asahiro et al. [AHI02] which shows that, given a graph on $n$ vertices with average degree $d$, there always exists a subgraph of $k$ vertices with at least $d\frac{k(k-1)}{n(n-1)}$ edges. Nevertheless, this result does not improve the upper bound for polynomial solvability of $\gamma$-DSP, for small values of $k \in o(n)$ and $d \in O(1)$. However, if there would be theoretical evidence that properties like the observed scale-invariance could also be applied to very small subgraph of general $\beta$-PL graphs, we could apply the result of Asahiro et al. in order to improve the upper bound for containment in **P**.

one of the connected components). This proposition holds since for every possible value of degree in $G$ there exist at least $k$ vertices with corresponding degree in $G'$. Further, all remaining vertices in $G'$ can be inter-connected according to their assigned degrees. Assuming, that it is possible to connect the remaining vertices in such a way that any subgraph on at most $k$ vertices has small average degree (bounded by some specific value less than $\frac{2\gamma(k)}{k}$), we could conclude that any subgraph on $k$ vertices with at least $\gamma(k)$ edges, if existing, must be a subgraph of the component that is isomorphic to $G$. As an example how to use the bound derived in Lemma 2.4 we refer to Cases II–V of the **NP**-completeness proof for $\gamma$-DSP on $\beta$-PL graphs (section 4.4).

As described earlier, the lower-bound for containment in **NP**-c strongly depends on the construction of $\beta$-PL graphs. We assume that using the large number of vertices with degree one, and some refined construction, it is possible to build power-law graphs in such a way that the above condition on subgraphs of size $k$ can be guaranteed. Further, if any subset of vertices has only few induced edges, its neighborhood must be large and therefore the graph has good expander properties [Alo86]. Based on the small observed average diameter of $\beta$-PL graphs, despite of missing some explicit rule of construction, we assume the existence of instances of $\beta$-PL graphs with this property. Nevertheless, at the current state of research, it is open how to construct $\beta$-PL graphs with the desired property. Solving this problem that is based in the research area of general power-law graphs, would improve the lower bound as suggested.

Based on these arguments and on the bounds derived for $\gamma$-DSP on general graphs, we assume that it is not possible to significantly improve the upper bound $\gamma \in k + O(1)$ for polynomial time tractability of $\gamma$-DSP, if restricting the problem to $\beta$-PL graphs (unless there exists some trivial solution). Similarly, the lower-bound $\gamma \in k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$, for **NP**-completeness is also likely to hold for this restricted version of the problem.

### 4.5.2 Discussion of the proposed reduction technique for NP-completeness of $\gamma$-DSP on $\beta$-PL graphs

In the preceeding subsection, we have argued that $\gamma$-DSP on $\beta$-PL graphs is **NP**-complete for $\gamma \in k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$. However, a theoretical precise proof of containment in this class of problems has only been shown for $\gamma(k) \geq \frac{15}{11}\delta \cdot k$ (see section 4.4). In the following, we analyze whether it is possible to improve the derived bound using the proposed reduction technique.

First of all, we determine some necessary properties for the parameters that are used within the reduction process. Then, we examine the possible topologies of the underlying graphs. Finally, with respect to these results, we discuss the quality of the lower bound proven in Theorem 4.5. Within this discussion, we argue that using the proposed reduction technique, it is not possible to derive significantly better bounds. Thus, in order to get better results it is necessary to use other techniques, e.g. based on improved bounds on the average degrees of subgraphs of power-law graphs (similar to the discussion in the preceeding subsection).

#### 4.5.2.1 Requirements of the reduction technique

The general idea of the reduction process is to construct a new graph $G'$ in such a way that the input graph $G$ contains a clique of size $k$ if and only if $G'$ contains a subgraph on $k'$ vertices and at least $\gamma(k')$ edges. The graph $G'$ is defined on the union of the three graphs $G_1$, $G_2$, and $G_3$, where $G_1$ represents the topology of the input graph, $G_2$ is used to adjust the number of edges in some densest subgraph of required size, and $G_3$ provides the remaining number of vertices to achieve the $\beta$-PL structure of the final graph. In the following, we only consider the case $G' = G_1 \uplus G_2 \uplus G_3$ for the construction of $G'$. The type of construction, wherein the vertices of $G_1$ and $G_2$ are connected completely, has no impact on the lower bound for **NP**-completeness of $\gamma$-DSP (due to the large average degree of some densest subgraph), and is thus not discussed.

Within the reduction, graph $G_1$ is defined to be the outcome of the transformation $N_x \circ S_t \circ R_s$ applied to the input graph $G$. While operator $R$ is only used to adjust the degrees within $G$, the transformations $N$ and $S$ are required to simultaneously decrease the average degree of any subgraph of $G$ without perturbing the overall topology. In section 3.2 we have seen that transformation $N_x \circ S_t$ can be described in terms of some general local graph transformations $\mathrm{Trans}_{G_v,G_e}$. Therefore, in the remaining part of the analysis, we use this transformation (instead of in $N_x \circ S_t$) in order to derive results for the proposed proof technique that are as general as possible. Throughout the discussion we use the following abbreviations:

$$z = k' - |V(G_2)| \qquad n_v = |V(G_v)| \qquad \alpha = \frac{2|E(G_v)|}{|V(G_v)|}$$

$$n_e = |V(G_e)| \qquad \beta = \frac{2|E(G_e)|}{|V(G_e)|}$$

**Properties based on construction rules of $G_1$:** First of all, we discuss the construction of $G_1$. A key observation of the general proof is that $G$ contains a clique of size $k$ if and only if $G_1$ contains a subgraph on $z$ vertices with $y = \gamma(k) - |E(G_2)|$ edges. Within the construction process $G_v$ and $G_e$ are chosen in such a way that $\mathrm{Trans}_{G_v,G_e}(K_k)$ is a graph on $z$ vertices with exactly $y$ edges. Further, the choice of these two graphs guarantees that any subgraph $G_1$ on that number of vertices has less edges, if $G$ contains no clique of size $k$. In the following, we derive some necessary properties for $G_v$ and $G_e$ with respect to this second condition.

Assume that $G$ does not contain a clique of size $k$. Nevertheless, we can assume that $G$ contains a subgraph $H$ on $k$ vertices that misses exactly one edge to form a clique, and further contains an additional edge $e = \{v_1, v_2\}$ with $v_1 \notin V(H)$ and $v_2 \in V(H)$. Obviously, the induced subgraph $H'$ of $G'$ corresponding to $H$ plus the copy of $G_e$ that corresponds to $e$, is a subgraph on $z$ vertices and $y - n_v$ edges. Due to the requirements of the proof it must not be possible to exchange vertices in $V(H')$ with vertices in $V(G') \setminus V(H')$ in such a way that the number of induced edges increases by at least $n_v$. In the following, we state some properties of $G_e$ and $G_v$ that would imply the existence of such a subgraph of $G'$ of size $z$ with at least $y$ edges:

- $\alpha \geq 2(\beta + 2\frac{n_v}{n_e})$ — For every copy of $G_e$ that is a subgraph of $H'$ the average degree of the vertices is $\beta + \frac{2n_v}{n_e}$. Therefore, we can remove $n_v$ vertices from the corresponding vertex sets in $H$ in such a way that at most $(\beta + \frac{2n_v}{n_e})n_v$ edges are lost. After removing these vertices, we select all $n_v$ vertices of $G_v^{v_1}$ (corresponding to the so far not selected vertex $v_1$ incident to edge $e$). Doing so, we gain at least $\frac{1}{2}\alpha n_v + n_v \geq (\beta + \frac{2n_v}{n_e})n_v + n_v$ edges. Thus, the number of induced edges is at least $|E(H)| + n_v \geq y$.

- *sparse and weakly-connected subgraphs in $G_e$* — In the above argumentation, we have only considered the average degree of $G_e$. Thus, it holds for any choice of $G_e$. We can reduce the restriction $\alpha \geq 2(\beta + 2\frac{n_v}{n_e})$ to $\alpha \geq \alpha_{\text{min}}$ (for some $\alpha_{\text{min}} < 2(\beta + 2\frac{n_v}{n_e})$), if we know that $V(G_e)$ contains a subset of $n_v$ vertices which is incident to at most $\frac{1}{2}\alpha_{\text{min}} n_v$ edges. Similarly, this observation holds if the set consists of $\frac{n_v}{i}$ vertices and is incident to at most $\frac{1}{2}\alpha_{\text{min}} \frac{n_v}{i}$ edges, with $i \in [\,1 \,..\, \binom{c}{2} - 1\,]$.
This discussion affects the existence of vertices with degree one in $G_e$ (including the edges to neighboring copies of $G_v$). Thus, the existence of each vertex with degree one in $G_e$ would result in least $\binom{k}{2} - 1$ vertices with degree one in $H'$. If the total number of these vertices with degree one is at least $n_v$ we could exchange these vertices, in the above described manner.

- *dense subgraphs in $G_e$ or $G_v$* — Let us assume that $G_e$ (resp., $G_v$) contains dense subgraphs $G_{\text{dense}}$ in such a way that we can exchange (keeping the number of vertices constant) several copies of $G_e$ in $H'$ (or corresponding dense and weakly connected subgraphs, respectively) with multiple copies of $G_{\text{dense}}$ not in $H'$ and increase the total number of induced edges by at least $n_v$. Similar to the previous item, the resulting induced subgraph of $G'$ would induce at least $y$ edges.

We can apply these observations to the definition of $G_1 = \text{Trans}_{G_e,G_v}(G)$ within the **NP**-completeness proof as follows. From section 3.2 we know that the average degree of appropriate subgraphs in $G_1$ mainly depends on $\text{avgdeg}(G_e)$, which itself depends on the values $\gamma(k')$ and $k'$. Further, for any specific choice of $\beta$, we know that $\alpha$ must not be too large. Similarly, we know that there must be neither too sparse nor too dense subgraphs in $G_e$ (resp., $G_v$).

Based on these observations, we can summarize that the degree distributions of $G_v$ and $G_e$ must be chosen carefully. Strictly speaking, especially if $\beta$ is small (what happens for the lower-bounds of **NP**-completeness of $\gamma$-DSP), the degrees within the corresponding degree sequences should not vary too much from the average degrees $\alpha$ resp. $\beta$. This observation backs our choice to choose transformation $N_x \circ S_t$, since this guarantees that all vertices (in $G_e$ and $G_v$) have degree close to $\beta \approx x + 1$, what corresponds to a valid choice w.r.t. the above discussion.

**Properties based on construction rules of $G_2$:** After considering the properties that result from the structure of graph $G_1$, we proceed to analyze properties of the graph $\tilde{G} = G_1 \uplus G_2$. Within the **NP**-completeness proof we require that a densest subgraph of $\tilde{G}$ on $k'$ vertices contains all vertices of $G_2$ and $z = k' - |V(G_2)|$ vertices of $G_1$. Let $H$ be a subgraph of $\tilde{G}$ that contains all vertices of $G_2$ and $z$ vertices of $G_1$ in such way that the subgraph in $G_1$ contains the maximal number of possible edges. In order to guarantee the correctness of the **NP**-completeness proof, it must not be possible to exchange vertices of $H$ in such a way that the number of edges increases.

Let $\rho$ be the average degree of $G_2$. In the following, we state properties of $G_v$, $G_e$ and $G_2$ which would imply that it is possible to modify the vertex set of $H$ in order to increase the number of induced edges.

- $\rho \leq \frac{\alpha}{2}$ *resp.* $\rho \leq \frac{\beta}{2}$ — Similarly to the discussion of the properties of $G_1$, we can remove $n_v$ vertices of $G_2$ (assuming $|V(G_2)| \geq n_v$) and loose at most $\rho n_v = \frac{\alpha}{2} n_v$ edges. Instead of these vertices, we choose an arbitrary so far unselected copy of $G_v$ and gain at least $\frac{\alpha}{2} n_v$ edges. The same argumentation applies when considering copies of $G_e$.

- *sparse and weakly connected subgraphs in* $G_2$ — Once again, the previous observation holds for all graphs $G_2$. However, it is sufficient that there exists a sparse subgraph in $G_2$ of appropriate size that is weakly connected to the rest of $G_2$. Replacing that subgraph with some sufficiently dense subgraph in $G_1$ increases the number of induced vertices.
  Therefore, similar to $G_1$ (see the discussion on the construction of $G_1$) the number of vertices with degree one within $G_2$ must not be too large. If their number is at least $n_v + n_e$ we can exchange these vertices with vertices in $G_1$ and gain edges (assuming arbitrary graphs $G_e$ and $G_v$, with $G_e$ connected or $\operatorname{avgdeg}(G_e) \geq 2$).

Similarly to the discussion on graph $G_1$, we obtain that graph $G_2$ must not contain too sparse and weakly connected subgraphs. Further, in order to prove exchangeability (see **NP**-completeness proofs for details), the value of $\delta$ has to be sufficiently large w.r.t. $\beta$. Therefore, when constructing $G_2$ we either have chosen quasi regular graphs with sufficiently large average degree or have constructed the graph in such a way, that the number of edges in any subgraphs in $G_1$ and $G_2$ of equal size differ by at least some number of edges (e.g., by choosing higher degrees or shorter chains; see construction of $G_2$ in Case I of the **NP**-completeness proof).

**Properties based on construction rules of $G_3$:** Within the **NP**-completeness proof for $\gamma$-DSP on general graphs, we have not had to satisfy any condition on the final degree sequence (corresponding to an empty graph $G_3$). However, if we want to guarantee some specific resulting degree sequence of $G'$ (e.g., $\beta$-PL graphs) we have to add an additional graph $G_3$. Further, we must guarantee that there exists some densest subgraph of $G'$ on $k'$ vertices that uses only vertices of graph $\tilde{G}$. Thus, similarly to the discussion on $G_1$ and $G_2$, it must hold that there exists no sufficiently dense subgraph in $G_3$. Otherwise we could exchange vertices in order to increase the number of induced edges. This observation implies some low average degree of $G_3$, which can be bounded in terms of $\alpha$, $\beta$, and $\delta$.

A first and straight forward approach of choosing $G_3$ is applied when proving Theorem 4.5. Within the construction of $G_3$ all vertices with degree above some threshold (e.g., the minimum degree in $\tilde{G}$) are either connected to vertices with sufficiently small degree, or inter-connected appropriately (e.g., using long chains). In order to improve the lower bound of **NP**-completeness it would be possible to choose some different construction for $G_3$ that allows to include more vertices of high degrees, while keeping the average degree of all subgraphs sufficiently small. This modification might enable to decrease the values of $\beta$ and $\delta$ and thus allow to perform the reduction of $\text{CLIQUE}_{\frac{1}{2}}$ to $\gamma$-DSP for some smaller choice of function $\gamma$.

#### 4.5.2.2 Discussion for $\beta$-PL input graphs

In the following, we restrict our discussion to $\beta$-PL input graphs (with $\beta > 2$). We derive additional properties for the parameters within the corresponding **NP**-completeness proof.

First of all, we summarize the relevant properties that have been achieved in the preceeding discussion (see paragraph 4.5.2.1):

- $G_1$ is constructed by applying a general local transformation $\text{Trans}_{G_e,G_v}$ to the input graph $G$, using appropriate graphs $G_v$ and $G_e$. This construction is equivalent to transformation $N_x \circ S_t$.

- The number of edges of $G_2$ is chosen in such a way that $|E(G_2)|$ plus the number of edges of a graph $\text{Trans}_{G_e,G_v}(K_k)$ equals $\gamma(k')$.

- The average degree of all subgraphs of $G_2$ is sufficiently large (i.e., some densest subgraph of $G'$ on $k$ vertices contains all vertices of $G_2$).

- The average degree of all subgraphs of $G_3$ is sufficiently small (i.e., some densest subgraph of $G'$ on $k$ vertices contains no vertices of $G_3$).

- Almost all vertices of degree one are element of $G_3$.

The proof of Theorem 4.5 provides an admissible choice and construction of graphs $G_1$, $G_2$ and $G_3$ in order to prove **NP**-completeness of $\gamma$-DSP on $\beta$-PL graphs for functions $\gamma$ with $\gamma(k) \geq \frac{15}{11}\delta \cdot k$. This bound is based on the average degree of a set of vertices of graph $G_2$ that contains almost all vertices of a $(N, \beta)$-PL degree sequence with degree at least two (see Lemma 2.2).

In order to improve this lower bound we either have to find some tighter calculation on the average degree of graph $G_2$ (however, in any case, the average degree is greater than the average degree of the whole $(N, \beta)$-PL degree sequence) or some refined definition of its set of vertices. Due to the former discussion, we can not include vertices of degree one in $G_2$. Thus, in order to decrease avgdeg$(G_2)$ we have to move large-degree vertices from $G_2$ to $G_3$ (without creating dense subgraphs in $G_3$). E.g., we can inter-connect these vertices to stars by using so far pairwise connected vertices of degree one. Analyzing the proposed proof, we observe that number pairwise connected vertices with degree one is at least $0.5N$ (and at most $N$ which is equal to the total number of such vertices). The following calculation shows that all vertices with degree at least $(c^{-1}\frac{\beta-1}{\beta-2})^{\frac{1}{\beta-2}}$ can be satisfied using $c \cdot N$ vertices of degree one, with $0 \leq c \leq 1$.

$$\sum_{i=x}^{N^{\frac{1}{\beta}}} i\lfloor N \cdot i^{-\beta} \rfloor \leq N\frac{\beta-1}{\beta-2}x^{2-\beta} \leq c \cdot N \quad \Rightarrow \quad x \geq \left(c^{-1}\frac{\beta-1}{\beta-2}\right)^{\frac{1}{\beta-2}}$$

However, for $\beta$-PL graphs with values of $\beta$ close to two (similar to graphs for most real-world networks) we know that this value is at least $(\frac{1}{2} \cdot \text{avgdeg}(G))^{\frac{1}{\beta-2}}$. Further, it is possible to observe that the average degree of $G_2$ decreases only slightly and that the lower bound of $\frac{15}{11}\delta \cdot k$ can be improved by at most some constant factor.

All in all, we can summarize that the analysis of Case I of the proposed reduction technique is tight, up the described constant factor. We even assume that the best lower bound that can be proven with this technique, is at least $\frac{1}{2}\delta \cdot k$ (corresponding to some average degree of at least $\delta$ within the desired subgraph). Thus, once again, in order to significantly improve the lower bound for **NP**-completeness (e.g., some bound independent of $\delta$), it seams to be necessary to use different concepts for the construction of $G'$ (see previous discussion).

### 4.5.3 Summary of the discussion

In the two preceeding subsections, we have discussed the bounds of polynomial tractability and **NP**-completeness for $\gamma$-DSP on $\beta$-PL graphs. We have observed that, when using the stated polynomial algorithm (see section 4.2) or the proposed reduction (see section 4.4) we cannot expect to significantly improve the derived bounds in either case. Further, we have given evidence that, for most of the functions $\gamma$ contained in the remaining gap, the problem $\gamma$-DSP is also **NP**-complete.

The exact location of the corresponding thresholds depends on so far unknown properties of $\beta$-PL graphs, i.e. constructions of graphs for given $\beta$-PL degree (sub)sequences that enable to bound the average degree of occurring subgraphs. We suggest that the bound that has been derived in Lemma 2.4 can be significantly improved. Further, we assume that the complexity of $\gamma$-DSP for $\beta$-PL input graphs is similar to the complexity of $\gamma$-DSP for general input graphs.

# Chapter 5

# Approximation of the DENSE-$k$-SUBGRAPH-PROBLEM

The results of the previous chapter have shown that, even when restricting to $\beta$-PL graphs, clustering based on $\gamma$-DSP is not polynomial tractable for a wide range of functions $\gamma$. Thus within this chapter, we focus on the approximability of $\gamma$-DSP by analyzing the MAX-DENSE-$k$-SUBGRAPH-PROBLEM (denoted by MAX-$k$-DSP):

**Approximation-Problem 5.1** MAX-DENSE-$k$-SUBGRAPH-PROBLEM

| | |
|---|---|
| *Input:* | $(G, k)$*, where* $G$ *is some undirected graph, and* $k \in \mathbb{N}$ |
| *Solutions:* | *all subgraphs* $G'$ *of* $G$ *on* $k$ *vertices* |
| *Value:* | $\text{val}(G') = \|E(G')\|$ |
| *Target:* | *MAX* |

For general input graphs, this problem is well studied in the literature (see section 5.1 for a detailed discussion). Similarly to the previous chapter, we also discuss the problem when restricting to $\beta$-PL input graphs. In order to derive results on the approximability for this restricted problem (see section 5.2), we prove equivalence to the general problem, w.r.t. $AP$-reduction. This result guarantees that both problems are contained in the same approximation class (i.e., algorithms with polynomial resp. constant approximation ratio for the first problem imply algorithms with polynomial resp. constant approximation ratio for the second one, and vice versa). Finally (see section 5.3), for the special graph class that represents the hyperlink structure of the WWW, we summarize some heuristics that are used within density based clustering and try to overcome the polynomial (so far best known) approximation ratio.

## 5.1 MAX-$k$-DSP on general graphs

In this section, we present an overview on the approximation results for MAX-$k$-DSP that are stated in the literature. Most of the results consider the edge-weighted version of the problem, i.e. every edge is assigned some weight and the objective function measures the sum of the edge weights of corresponding subgraphs.

While the (weighted or unweighted) problem is obviously contained in the approximation class $\mathcal{NPO}$, there exists no algorithm proving its membership in class $\mathcal{APX}$, so far. Feige, Kortsarz, and Peleg even conjecture that the problem is hard to approximate within factor $n^{\varepsilon}$, for some $\varepsilon > 0$ [FKP01]. Nevertheless, there exist no results stating that MAX-$k$-DSP on general graphs is not approximable within factor $(1+\varepsilon)$, for some $\varepsilon > 0$. However, Feige [Fei02] has given strong evidence for non-membership in the corresponding approximation class (see subsection 5.1.4). All in all, we can summarize that, at the current state of research, it is has to be assumed that there exists no algorithm approximating the problem within some constant factor.

If the class of input graphs is restricted to complete weighted graphs, where the weight function satisfies the triangle inequality, (often denoted as the MAXIMUM-DISPERSION-PROBLEM [WK88]) there exist algorithms with constant approximation ratios. Ravi, Rosenkrantz, and Tayi [RRT94] have proven a 4-approximation that greedily extends a set $S$ of vertices (initially containing the vertices incident to some heaviest edge) with some vertex whose edges, incident to $S$, have maximum sum of weights. This result has been improved with a 2-approximation of Hassin, Rubinstein, and Tamir [HRT97] whose greedy algorithm iteratively adds the vertices of some global heaviest edge (while neglecting all edges incident to so far selected vertices).

Further, for instances on dense graphs, Arora, Karger, and Karpinski [AKK99] proved the existence of a polynomial time approximation scheme ($\mathcal{PTAS}$). If either $k \in \Omega(n)$ and $|E| \in \Omega(|V|^2)$ or the minimum degree of the input graph is in $\Omega(n)$, it is possible to achieve some polynomial-time algorithm with approximation ratio $1+\varepsilon$, for every $\varepsilon > 0$. Using Szemeredi's regularity lemma [Sze78], Czygrinow [Czy00] even proved an $\mathcal{PTAS}$ with calculation time $O(|V|^{2.4})$ if the optimal solution is in $\Omega(|V|^2)$. However, due to a large constant that depends on $\frac{1}{\varepsilon}$ and is hidden by the usage of the Landau symbol $O$, the algorithm is no $\mathcal{FPTAS}$.

In the following, we present several optimization algorithms for MAX-$k$-DSP (on general input graphs) that guarantee approximation ratios of $O(n^{\varepsilon})$, for some $\varepsilon < 1$.[1] Obviously, by choosing vertices corresponding to arbitrarily selected $\lfloor k/2 \rfloor$ edges (plus some additional vertex, if the value of $k$ is odd) it is possible

[1] In order to guarantee this property, for some of the following algorithms, we have to restrict the range of parameter $k$.

to get the trivial approximation ratio of $O(n)$. Therefore, only approximation ratios with $\varepsilon < 1$ are of interest.

### 5.1.1 Greedy algorithms

The above described greedy algorithms for complete weighted graphs satisfying the triangle fail to perform well on general weight functions (see [KP93]). However, Asahiro et al. [AITT00] stated a simple greedy algorithm *GREEDY* for unweighted graphs that iteratively removes vertices with least induced degree until finally there remains a graph on the required number of vertices. Lemma 5.1 states the upper and lower bounds for the approximation ratio.

**Lemma 5.1 ([AITT00])** *The worst-case approximation ratio $R$ of GREEDY satisfies*

$$\begin{array}{rcccll} 2\left(\frac{n}{k}-1\right)-O\left(\frac{1}{k}\right) & \leq & R & \leq & 2\left(\frac{n}{k}-1\right)-O\left(\frac{n}{k^2}\right) & \textit{for } k \leq \frac{n}{3}, \textit{ and} \\ \left(\frac{1}{2}+\frac{n}{2k}\right)^2-O\left(n^{-\frac{1}{3}}\right) & \leq & R & \leq & \left(\frac{1}{2}+\frac{n}{2k}\right)^2+O\left(\frac{1}{n}\right) & \textit{for } \frac{n}{3} < k \leq n. \end{array}$$

Using these bounds, for $k \in \Omega(n)$, the approximation ratio $R$ is seen to be constant. Nevertheless, for other choices of $k$ the ratio $\frac{n}{k}$ results in an exponential approximation ratio $R$ (e.g., $k \in \Theta(n^{\frac{1}{3}})$ results to $R \in \Theta(2n^{\frac{2}{3}})$).

### 5.1.2 LP- and SDP-Relaxations

As a consequence on Goemans and Williams approach to use semidefinite programming (SDP) for approximating the maximum cut and satisfiability problems [GW95], several authors also applied SDP resp. linear programming (LP) to MAX-$k$-DSP.

Typically, MAX-$k$-DSP is described in terms of a the following 0–1 integer quadratic program. Variable $x_i$ indicates for every vertex $v_i$ of the input graph whether $v_i$ is contained in the resulting subgraph or not. The additional variables $w_{i,j}$ represent the possibly present edges weights (once again, values 0 and 1 can be used to model unweighted graphs).

**Definition 5.1 Integer quadratic program for** MAX-$k$-DSP

$$\begin{array}{rl} \textit{Maximize:} & \sum_{i<j} w_{i,j} x_i x_j \\ \textit{subject to:} & \sum_{i=1}^{n} x_i = k \\ & x_i \in \{0,1\} \textit{ for all } 1 \leq i \leq n \end{array}$$

Using relaxations, this problem can be described in terms of SDPs resp. LPs (e.g., see [FS97, SW98, FL01, HYZ02]). Applying rounding techniques to the resulting solutions of these programs it is possible to derive solutions for the corresponding Max-$k$-DSP instances. Usually, the approximation ratios of these algorithms are close to $\frac{n}{k}$. While the earliest attempts performed worse than $\frac{n}{k}$, some of the recently invented programs (e.g. [HYZ02]) even perform slightly better (i.e. achieving approximation ratios less than $\frac{n}{k}$), for some ranges of $k$. However, only if $k$ is linear in $n$ these algorithms can guarantee constant approximation ratios. Feige and Seltser [FS97] even proved that for $k \approx n^{\frac{1}{3}}$ semidefinite programs (or at least some subclass) fail to distinguish between graphs that contain a clique of size $k$ and graphs in which the densest subgraph on $k$ vertices has average degree below $\log n$. This corresponds to a gap (i.e., some lower bound for the approximation ratio) of $\tilde{\Omega}(n^{\frac{1}{3}})$ between the value of the SDP and the optimal value of the Max-$k$-DSP instance.

Finally, we can summarize that it is possible to outperform the standard GREEDY algorithm of Asahiro et al. (see subsection 5.1.1), by using SDP or LP relaxations. Further, for values $k \in \Omega(n)$ the resulting approximation ratios are constant, and thus, the corresponding algorithms guarantee good approximations. However, for $k \in o(n)$ the approximation ratios of these techniques can approach to $O(n)$, which has been seen to be a trivial upper bound.

### 5.1.3 Combined algorithms

In the following, we present optimization algorithms that use combinations of different techniques (e.g. greedy techniques, shortest paths, etc.) and guarantee worst-case approximation ratios better than $O(n)$ (trivial solution), for the whole range of $k$. Some years before the usage of SDP-relaxations, the first exponential approximation ratio of $O(n^{\frac{7}{18}})$ (more precisely $O(n^{\varepsilon})$, within $\varepsilon$ converging to some value close to 0.388445) has been presented by Kortsarz and Peleg [KP93]. In a later work, Feige, Kortsarz, and Peleg derive the so far best known polynomial-time algorithm for Max-$k$-DSP that approximates the maximum average degree of a $k$-vertex subgraphs within factor $O(n^{\frac{1}{3}-\varepsilon})$, for some $\varepsilon > 0$ [FKP01] .

To introduce the main principles of this approach, we outline the overall idea and results of a similar algorithm of Feige et al. (see also [FKP01]) that guarantees an approximation ratio of $O(n^{\frac{1}{3}})$ and is based on the following three procedures:

- Procedure 1 (trivial procedure):

> Select $\frac{k}{2}$ arbitrary edges from $G$. Let $S$ be the set of vertices incident with these edges. We add arbitrary vertices to $S$, which are connected to $S$, if its size is smaller than $k$. Return the subgraph of $G$ that is induced by $S$.

- PROCEDURE 2 (greedy procedure):

  Sort the vertices of $G$ by order of their degree. Let $H$ denote the $\lfloor \frac{k}{2} \rfloor$ vertices with highest degree in $G$ (breaking ties arbitrarily). Sort the remaining vertices by the number of neighbors they have in $H$. Let $C$ denote the $\lceil \frac{k}{2} \rceil$ vertices in $G \setminus H$ with the largest number of neighbors in $H$. Return the subgraph of $G$ that is induced by $H \cup C$.

- PROCEDURE 3 (using walks of length 2, in order to find dense subgraphs that may have only few vertices with high degree)

  Let $W_l(v, w)$ be the number of walks of length $l$ from $v$ to $w$. Compute $W_2(v, w)$ for all pairs of vertices. Construct a candidate graph $\mathcal{H}^v$ for every vertex $v$ of $G$, as follows:

  1. Sort the vertices of $V(G) \setminus \{v\} = \{w_1, w_2, \ldots, w_{|V(G|-1}\}$ by non-increasing order of their number of length-2 walks to $v$, i.e., $W_2(v, w_{i_1}) \geq W_2(v, w_{i_2}) \geq \ldots$, and let $P^v$ denote the set $\{w_{i_1}, \ldots, w_{i_{\frac{k}{2}}}\}$.
  2. Compute for every neighbor $x$ of $v$ the number of edges connecting $x$ to $P^v$ (denoted by $\deg(x, P^v)$) and construct a set $B^v$ containing the $\frac{k}{2}$ neighbors of $v$ with highest value $\deg(x, P^v)$.
  3. Let $\mathcal{H}^v$ denote the subgraph of $G$ induced by $P^v \cup B^v$. If $\mathcal{H}^v$ still contains less than $k$ vertices, then it is completed to size $k$ arbitrarily.

  Select the densest candidate graph $\mathcal{H}^v$ to be the output graph of PROCEDURE 3.

Let $A_i(G, k)$ denote the average degree of the graph returned by PROCEDURE $i$ applied on the input $(G, k)$, for $i \in \{1, 2, 3\}$. The following lower bounds hold:

- $A_1(G, k) \geq 1$
- $A_2(G, k) \geq \frac{k \cdot d_H}{2n}$
- $A_3(G, k) \geq \frac{(d^*(G,k))^2}{2 \max\{k, 2\Delta(G)\}}$

Where $\Delta(G)$ denotes the maximum degree in $G$, $d_H$ is the average degree (w.r.t. $G$) of the vertices in $H$ (definition see PROCEDURE 2), and $d^*(G, k)$ represents the maximum average degree of all subgraphs of $G$ on $k$ vertices.

The overall algorithm applies the three above procedures to the input (where PROCEDURE 3 is applied only to some appropriate subgraph of $G$) and outputs the densest of the three resulting graphs. Using the above properties Feige et al. have shown that the average degree $A(G,k)$ of the resulting graph satisfies

$$A(G,k) \geq \left(1 \cdot \frac{kd_H}{2n} \cdot \frac{(d^*(G,k))^2}{2\max\{k, 2\Delta(G)\}}\right) \geq \frac{d^*(G,k)}{2n^{\frac{1}{3}}}.$$

Therefore, the guaranteed approximation ratio of the algorithm is $O(n^{\frac{1}{3}})$.

In order to improve the worst-case approximation ratio to $O(n^{\frac{1}{3}-\varepsilon})$, for some $\varepsilon > 0$, Feige et al. have defined two more procedures. Similar to PROCEDURE 3, these procedures output graphs that dependent on the occurrences of paths of length 3 and 5. Based on these additional procedures, it is possible to prove better approximation ratio for those parameter settings, where the first three procedures can only guarantee an approximation ratio of $O(n^{\frac{1}{3}})$.

### 5.1.4 R3SAT-hardness of MAX-$k$-DSP

At the beginning of this section, we stated that currently there exists no result on the approximation hardness of the problem MAX-$k$-DSP. However, besides the conjecture of Feige, Kortsarz, and Peleg that the problem is hard to approximate within a factor $O(n^{\varepsilon})$ for some $\varepsilon > 0$ [FKP01], there exists further evidence for hardness of the problem.

We consider the following idea (e.g., see [Fei02]). Assume that there exists some problem $A$, where it is not known how to prove **NP**-hardness for algorithms that approximate problem $A$ within some ratio $R$. Further let $B$ be some problem that is assumed to have no polynomial algorithm. In order to discuss the complexity of problem $A$, we prove that $A$ is hard to approximate within ratio $R$ w.r.t. to $B$. I.e., the existence of an optimization algorithm with corresponding approximation ratio for problem $A$ would imply the existence of a polynomial algorithm of problem $B$, which is not assumed to exist.

Feige considered the following hypothesis on 3CNF formulas (i.e., instances of the 3SAT problem, where every clause contains three literals) that is assumed to hold, in general [Fei02].

> *For every fixed $\varepsilon > 0$, for $\Delta$ a sufficiently large constant independent of $n$, there is no polynomial time algorithm that on most 3CNF formulas with $n$ variables and $m = \Delta n$ clauses outputs typical, but never outputs typical on 3CNF formulas with $(1-\varepsilon)m$ satisfiable clauses.*

Based on this hypothesis, Feige has proven the following result.

> *The* MAXIMUM-DENSE-$k$-SUBGRAPH-PROBLEM *is R3SAT-hard to approximate within some constant $\rho < 1$.*

Using the motivation from above, this results denotes that if there exists an optimization algorithm for Max-$k$-DSP with a constant approximation ratio that dependent on $\rho$, the above hypothesis on random 3CNF formulas does not hold. This would contradict the assumed correctness of the hypothesis. Therefore, such an algorithm is not very likely to exist.

Despite this property of Max-$k$-DSP does not imply Max-$k$-DSP $\notin \mathcal{PTAS}$, it still gives further evidence for the non-membership in this approximation class.

## 5.2 Max-$k$-DSP on $\beta$-PL graphs

In this section, similarly to the discussion of the complexity of $\gamma$-DSP, we restrict the optimization problem Max-$k$-DSP to the class of $\beta$-PL graphs (denoted as Max-$k$-DSP-$\beta$-PL). Due to the specific degree structure of this graph class it might be possible to improve the approximation ratios that have been described in the previous section. In the following (subsection 5.2.1) we prove equivalency of Max-$k$-DSP-$\beta$-PL and Max-$k$-DSP w.r.t. $AP$-reduction. Based on this observation, we know that both problems are contained in the same approximation class. Therefore, we can state similar approximation results for the restricted problem Max-$k$-DSP-$\beta$-PL (subsection 5.2.2).

### 5.2.1 Max-$k$-DSP $\equiv_{AP}$ Max-$k$-DSP-$\beta$-PL

Before proving the equivalence of Max-$k$-DSP and Max-$k$-DSP-$\beta$-PL w.r.t. $AP$-reduction, we formally define both problems according to Definition B.1.

**Definition 5.2** *The two optimization problems* Max-$k$-DSP *and* Max-$k$-DSP-$\beta$-PL *are defined as follows:*

$$\begin{array}{lcl} \text{Max-}k\text{-DSP} & = & \langle\ \mathcal{I}_{\text{DSP}},\ \mathcal{S}ol_{\text{DSP}},\ \text{val}_{\text{DSP}},\ \max\ \rangle \\ \text{Max-}k\text{-DSP-}\beta\text{-PL} & = & \langle\ \mathcal{I}_{\text{DSP-}\beta\text{-PL}},\ \mathcal{S}ol_{\text{DSP-}\beta\text{-PL}},\ \text{val}_{\text{DSP-}\beta\text{-PL}},\ \max\ \rangle \end{array}$$

*where $\beta > 2$ is some fixed rational value, and*

$$\begin{array}{lcl} \mathcal{I}_{\text{DSP}} & = & \{\ (G,k)\mid G \textit{ is a graph} \ \wedge\ 1 \le k \le |V(G)|\ \} \\ \mathcal{I}_{\text{DSP-}\beta\text{-PL}} & = & \{\ (G,k)\mid G \textit{ is a } \beta\textit{-PL graph} \ \wedge\ 1 \le k \le |V(G)|\ \} \\ \mathcal{S}ol_{\text{DSP}}(G,k) & = & \{\ G' \mid G' \text{ is a subgraph of } G\ \wedge\ |V(G')| = k\ \} \\ \mathcal{S}ol_{\text{DSP-}\beta\text{-PL}}(G,k) & = & \{\ G' \mid G' \text{ is a subgraph of } G\ \wedge\ |V(G')| = k\ \} \\ \text{val}_{\text{DSP}}(I,G') & = & |E(G')| \\ \text{val}_{\text{DSP-}\beta\text{-PL}}(I,G') & = & |E(G')| \end{array}$$

The following theorem states the equivalence w.r.t. $AP$-reduction of the two above optimization problems.

**Theorem 5.1** *The two problems* MAX-$k$-DSP *and* MAX-$k$-DSP-$\beta$-PL *are equivalent w.r.t. AP-reduction, i.e.*

$$\text{MAX-}k\text{-DSP} \equiv_{AP} \text{MAX-}k\text{-DSP-}\beta\text{-PL}.$$

The correctness of the Theorem 5.1 is shown using the following two lemmata.

**Lemma 5.2** MAX-$k$-DSP-$\beta$-PL $\leq_{AP}$ MAX-$k$-DSP

*Proof*: Obviously $\mathcal{I}_{\text{DSP-}\beta\text{-PL}}$ is a subset of $\mathcal{I}_{\text{DSP}}$, while all other entries in the quadruples defining MAX-$k$-DSP-$\beta$-PL and MAX-$k$-DSP are equivalent. Therefore, the triple $(\text{id}, \text{id}, 1)$, where id is the identity function, is an $AP$-reduction (see Definition B.6) from MAX-$k$-DSP-$\beta$-PL to MAX-$k$-DSP. □

**Lemma 5.3** MAX-$k$-DSP $\leq_{AP}$ MAX-$k$-DSP-$\beta$-PL

*Proof*: Our proof of this reduction is based on a graph transformation $F$, which transforms the input graph $G$ into a $\beta$-PL graph $G' = F(G)$. Within the proof, we show that for any subgraph $H'$ of $G'$ of size $k'$ we can construct a subgraph $H$ of $G$ on $k$ vertices whose average degree differs by at most some constant factor from the average degree of $H$. We use the following constructive definition of transformation $F$. Let $F(G) = G' = G_1 \uplus G_2$, where the two graphs $G_1$ and $G_2$ are defined as follows:

1. In order to construct $G_1$, we add a clique of size $d = \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} + 1 \right\rceil + 2$ to the input graph $G$. This is done in such a way that all vertices of the clique are fully connected to the vertices of $G$. This operation is equivalent to transformation $R_s$, with $s = d$ (see subsection 3.1.1), that has been used within the **NP**-completeness proofs of $\gamma$-DSP. We define $G_1 = R_s(G)$ and observe the following property.

   > $G$ has a subgraph $H$ on $k$ vertices with at least $\gamma \cdot k$ edges iff $G_1$ has a subgraph $H_1$ on $k+d$ vertices with at least $\gamma \cdot k + k \cdot d + \binom{d}{2}$ edges.

   The correctness of this statement is easily seen. Firstly, assume that $G$ contains such a subgraph $H$ on $k$ vertices. The induced subgraph in $G_1$, whose vertices correspond to the $k$ vertices of $H$ and the $d$ vertices of the clique, has sufficient number of vertices and edges.
   Secondly, assume that $G'$ contains a subgraph $H_1$ on $k + d$ vertices and at least $\gamma \cdot k + k \cdot d + \binom{d}{2}$ edges. Obviously, we can exchange the vertices in $H_1$ in such a way that all vertices of the clique (added by transformation $R$) are selected without decreasing the number of induced edges. Thus, w.l.o.g.

we can assume that $H_1$ contains all vertices of the clique. Consequently, since $H_1$ has at least $\gamma \cdot k + k \cdot d + \binom{d}{2}$ edges, there must exists a subgraph $H$ of $G$ with $k$ vertices and at least $\gamma \cdot k$ edges (this subgraph results when removing all vertices of $H_1$ that are elements of the clique).

2. Now, we define some graph $G_2$ in such a way that $G' = G_1 \uplus G_2$ is a $\beta$-PL graph. To do so, we choose $N = (|V(G)| + d)^{\beta+1}$. This choice of $N$ assures that for all occurring degrees $i$ in $G_1$ the number of vertices in the $(N, \beta)$-PL degree sequence with degree $i$ is large enough to subsume all corresponding vertices of $G_1$. Further, we know that all vertices in $G_1$ have degree at least $d - 1 = \left\lceil 4 \cdot 2^{\frac{1}{\beta-2}} + 1 \right\rceil + 1$. Therefore, when removing the degrees corresponding to all vertices of $G_1$ from the $(N, \beta)$-PL degree sequence, the remaining degree sequence $S$ matches all conditions that are required in Lemma 2.4. Consequently we can apply the lemma and build some graph $G_2$ with the degree sequence $S$ (and size polynomial in the size of $G$) in such a way that any subgraph of $G_2$ has average degree at most $d - 1$.

Based on transformation $F$, we define the following two functions $f$ and $g$, which are used within the $AP$-reduction.

1. $f : \mathcal{I}_{\mathrm{DSP}} \times \mathbb{R}^+ \to \mathcal{I}_{\mathrm{DSP}\text{-}\beta\text{-}\mathrm{PL}}$:

   This function is used to transform an instance of Max-$k$-DSP to an instance of Max-$k$-DSP-$\beta$-PL. Using transformation $F$, we define

$$f(\,(G, k), \delta) = (F(G), k + d)$$

2. $g : \mathcal{I}_{\mathrm{DSP}} \times \mathcal{S}ol_{\mathrm{DSP}\text{-}\beta\text{-}\mathrm{PL}} \times \mathbb{R}^+ \to \mathcal{S}ol_{\mathrm{DSP}}$:

   Within the $AP$-reduction, we map instances $(G, k)$ of Max-$k$-DSP to instances $(G', k')$ of Max-$k$-DSP-$\beta$-PL. Function $g$ is used to define the (reverse) mapping for a solution of $(G', k')$ to a solution of $(G, k)$. In the following we define the output for any input triple $\big((G, k),\, H',\, \delta\big)$, with $(G, k) \in \mathcal{I}_{\mathrm{DSP}}$, $H' \in \mathcal{S}ol_{\mathrm{DSP}\text{-}\beta\text{-}\mathrm{PL}}$ and $\delta \in \mathbb{R}^+$.

   Since we want to use function $g$ within an $AP$-reduction we only consider the case that $H'$ is a subgraph of $f\,\big((G, k),\, \delta\big)$ on $k + d$ vertices (otherwise, in order to define $g$ on the entire input range, we define the output to be an arbitrary subgraph of $G$ on $k$ vertices).

   We know that the vertex set of $H'$ can be partitioned (w.r.t. definition of transformation $F$) into the three sets:

   - vertices of $G$,
   - vertices of the added clique $C$ of size $d$, and
   - vertices of $G_2$.

Further based on the definition of $F$, we know that the average induced degree of the corresponding vertices in $V(G_2)$ is at most $d-1$. Therefore, we can exchange the vertices (without decreasing the induced degree) in such a way, that all vertices $d$ vertices of clique $C$ and $k$ vertices of $G$ are selected. Additionally, we can require that the $k$ vertices of $G$ induce at least $\lfloor \frac{k}{2} \rfloor$ edges in $G$. Otherwise, we perform additional exchanges with vertices incident to that number of arbitrarily chosen edges in $G$.

Finally, we define the output graph of function $g$ to be the subgraph of $G$ that is induced by the $k$ selected vertices of $G$. Based on the above construction we know that this graph has average degree at least $\frac{1}{2} - \frac{1}{k}$.

In the remaining part of the proof, we show that the triple $(f, g, 10d)$ is an $AP$-reduction. In order to do so, we satisfy properties $AP1$ – $AP4$ required in the definition of the $AP$-reduction (see Definition B.6):

- $AP1$, $AP2$: These properties follow directly from the above definitions of $f$ and $g$.
- $AP3$: Both functions are independent of the choice $\delta$. Thus, using the above definitions, we can conclude that $f$ and $g$ are computable polynomial time.
- $AP4$: Let $I = (G, k) \in \mathcal{I}_{\text{DSP}}$, $\delta > 0$ , and $H' \in \mathcal{S}ol_{\text{DSP-}\beta\text{-PL}}(f(I,\delta))$ be any subgraph of $G' = f(I,\delta)$ on $k+d$ vertices. Further, let $\hat{H}$ be the subgraph of $G'$ on $k+d$ vertices that is constructed when applying function $g$ on the triple $(\,(G', k+d), H', \delta)$, i.e., we know that $\hat{H}$ contains all $d$ vertices of the adjoint clique plus $k$ vertices of $G$.

  According to the definition of $g$ we get that $\hat{H}$ has at least as many edges as $H'$. Therefore it holds that
  $$\frac{\text{val}(f(I,\delta), H')}{\text{opt}(f(I,\delta))} \le \frac{\text{val}(f(I,\delta), \hat{H})}{\text{opt}(f(I,\delta))}$$
  and consequently
  $$\frac{1}{1+\delta} \le \frac{\text{val}(f(I,\delta), H')}{\text{opt}(f(I,\delta))} \quad \Rightarrow \quad \frac{1}{1+\delta} \le \frac{\text{val}(f(I,\delta), \hat{H})}{\text{opt}(f(I,\delta))}.$$
  Further, due to the definition of function $g$, it holds that
  $$g(I, H', \gamma) = g(I, \hat{H}, \gamma).$$
  Combining these results we can show property $AP4$, by proving the following implication (trivially, the upper bounds hold for all maximization problems)
  $$\frac{1}{1+\delta} \le \frac{\text{val}(f(I,\delta), \hat{H})}{\text{opt}(f(I,\delta))} \quad \Rightarrow \quad \frac{1}{1+\alpha\delta} \le \frac{\text{val}(I, g(I, \hat{H}, \delta))}{\text{opt}(I)}.$$

First of all, we derive some results on the number of edges of the graphs that are used within the reduction process.

- The number of edges in $\hat{H}$ (i.e., $\mathrm{val}(f(I,\delta),\hat{H})$) is the sum of
  - ∗ the number of edges that are induced from the $k$ vertices of $\hat{H}$ that are also contained in $G$ (i.e., $\mathrm{val}(I,g(I,\hat{H},\delta))$, and
  - ∗ $dk+\binom{d}{2}$ (i.e, the number of edges in $\hat{H}$ that are incident to vertices of the clique).

  Therefore we get:

  $$\mathrm{val}(f(I,\delta),\hat{H})) = \mathrm{val}(I,g(I,\hat{H},\delta)) + dk + \binom{d}{2}$$

- The maximum number of edges of a subgraph of $G'$ on $k+d$ vertices (i.e., $\mathrm{opt}(f(I,\delta)))$ is the maximum number of edges of a subgraph of $G$ on $k$ vertices (i.e., $\mathrm{opt}(I)$) plus $dk+\binom{d}{2}$ (once again, this number corresponding to the number of edges incident to vertices of the clique).

  $$\mathrm{opt}(f(I,\delta)) = \mathrm{opt}(I) + dk + \binom{d}{2}$$

Finally, we define $\gamma$ to be the average degree of $g(I,\hat{H},\delta)$ (i.e., the subgraph of $G$ on $k$ vertices that results, when applying function $g$ on $G'$ and $H'$) and $\gamma_{\mathrm{opt}}$ to be the average degree of some densest subgraph of $G$ on $k$ vertices.

Now, we a ready to prove the above implication:

- The left hand side can be transformed to

$$\begin{array}{rcll}
\frac{1}{1+\delta} & \leq & \frac{\mathrm{val}(f(I,\delta),\hat{H})}{\mathrm{opt}(f(I,\delta))} & \Leftrightarrow \\
\mathrm{opt}(f(I,\delta)) - \mathrm{val}(f(I,\delta),\hat{H}) & \leq & \delta\cdot\mathrm{val}(f(I,\delta),\hat{H}) & \Leftrightarrow \\
(\gamma_{\mathrm{opt}}-\gamma)k & \leq & \delta\cdot\big(\gamma k + 2kd + d(d-1)\big) & \Leftrightarrow \\
\gamma_{\mathrm{opt}}-\gamma & \leq & \delta\cdot\left(\gamma + 2d + \frac{d}{k}(d-1)\right) &
\end{array}$$

- The right hand side results to:

$$\begin{array}{rcll}
\frac{1}{1+\alpha\delta} & \leq & \frac{\mathrm{val}(I,g(I,y,\delta))}{\mathrm{opt}(I)} & \Leftrightarrow \\
\mathrm{opt}(I) - \mathrm{val}(I,g(I,y,\delta)) & \leq & \alpha\delta\mathrm{val}(I,g(I,y,\delta)) & \Leftrightarrow \\
\gamma_{\mathrm{opt}}-\gamma & \leq & \delta\cdot\alpha\gamma &
\end{array}$$

Combining the last predicates, which are equivalent to the original left and the right hand side of the above implication, we can prove $AP4$ if

$$\left(\gamma' + 2d + \frac{d}{k}(d-1)\right) \quad \leq \quad \alpha\gamma'.$$

For large enough values of $k$, we can assume $k > d$ and $\gamma' \geq \frac{1}{3}$. Finally, in order to prove $AP4$, we can evaluate:

$$\left(\gamma' + 2d + \frac{d}{k}(d-1)\right) \;\leq\; \gamma' + 9d\frac{1}{3} \;\leq\; 10d\gamma' = \alpha\gamma'.$$

Thus, we have proven $AP1 - AP4$ and consequently the $AP$-reduction holds. □

### 5.2.2 Approximation results

In the previous subsection, we have proven an important result for characterizing the approximability of MAX-$k$-DSP-$\beta$-PL. Based on this results we know that both problems MAX-$k$-DSP on general input graphs and on $\beta$-PL input graphs are contained in the same approximation class w.r.t. $AP$-Reduction. Thus, based on Lemmata B.1 and B.2, we can state the following corollary:

**Corollary 5.1**

$$\begin{array}{rcl} \text{MAX-}k\text{-DSP} \in \mathcal{APX} & \Leftrightarrow & \text{MAX-}k\text{-DSP-}\beta\text{-PL} \in \mathcal{APX} \\ \text{MAX-}k\text{-DSP} \in \mathcal{PTAS} & \Leftrightarrow & \text{MAX-}k\text{-DSP-}\beta\text{-PL} \in \mathcal{PTAS} \end{array}$$

Combining this corollary and the currently best known results on MAX-$k$-DSP (see previous section), we can state the following results:

1. There exists an optimization algorithm that approximates MAX-$k$-DSP on $\beta$-PL graphs within a ratio of $O(n^{\frac{1}{3}-\varepsilon})$, for some $\varepsilon > 0$.
2. While it is still open whether it is possible to approximate MAX-$k$-DSP on $\beta$-PL within $1 + \varepsilon'$, for some $\varepsilon' > 0$, it is very likely that it is **NP**-complete to approximate the problem within $O(n^{\varepsilon''})$, for some $\varepsilon'' > 0$ (using similar propositions for MAX-$k$-DSP on general graphs).

Due to the numerous studies that have been performed on MAX-$k$-DSP on general graph, we assume that the approximability of problem is well classified. Therefore, we do not expect to derive a better classification based on the analysis of MAX-$k$-DSP on $\beta$-PL graphs.

At this point we stop the details discussion on the approximability and leave the problem to improve the currently best known exponent of $\frac{1}{3} - \varepsilon$ as an open problem. These improvements can be based on further analysis of the power-law degree sequence. E.g. they can either improve the derived approximation solutions, or result to some stricter bound for the optimal solution. As some first indication we state the following easily seen facts. For $k > n^{\frac{3}{4}}$ a densest subgraph on $k$ vertices can have degree at most $n^{\frac{1}{4}}$ and thus even the trivial algorithm guarantees an approximation ratio of $O(n^{\frac{1}{4}})$. Similarly, for $k < n^{\frac{1}{2}}$ we can use Procedure 3 of [FKP01] in order to guarantee the same ratio. Therefore, it is possible to approximate Max-$k$-DSP-$\beta$-PL within $O(n^{\frac{1}{4}})$ if we restrict $k \notin [n^{\frac{1}{2}} .. n^{\frac{3}{4}}]$. However, deriving more refined and tight results is an further extensive problem and thus not covered within this thesis.

Before proceeding with the description of some heuristics that use additional properties, different to power-laws, in order to improve the clustering results, we want to mention the approach of Sagie and Wool [SW03] to cluster the graph representing the structure of autonomous systems (AS) in the Internet [MP01], by applying the algorithm of Feige, Kortsarz, and Peleg [FKP01]. Although, as we have presented, the worst-case approximation ratio is known to be exponential in the number of vertices, the authors derive some empirical good clustering of the AS graph. As already suggested within the discussion on large-scale systems (see chapter 2), this result also indicates the usability of the density-based clustering approaches within real-world large-scale networks.

## 5.3 Heuristics for Max-$k$-DSP on graphs for the hyperlink structure of the WWW

In the above section, we have shown that the optimization problem Max-$k$-DSP on $\beta$-PL graphs is most likely not contained in the approximation class $\mathcal{APX}$. Therefore, when using density-based clustering we cannot refer to some overall sufficient approximation technique (the term sufficient denotes that some small approximation ratio, e.g. a small constant, is guaranteed). Consequently, in order to derive a partitioning of the vertex set that indicates dense subgraphs, we have to make further assumptions. We can either restrict to very special parameter settings (e.g., large values of $k$), or include further properties of the underlying graphs. In this section we present heuristics that use the second approach.

Communities in the WWW usually correspond to dense subgraphs within the hyperlink graph (see section 2.3). Therefore, if it is possible to detecting communities (of given size), we can also assume to derive good approximations of dense subgraphs in the graph representing the hyperlink structure of the WWW. In the following, we explain two approaches [KPRT99b, KRK01a] that have been proposed to locate communities in the hyperlink graph.

The overall idea of these two algorithms is based on the observation that most communities in the WWW can be identified by detecting dense bipartite subgraphs, where the two corresponding sets of vertices are often referred to as hubs and authorities (see subsection 2.3.2). While the first algorithm proposed by Kumar et al. indicates these dense subgraphs using complete bipartite graphs (subsection 5.3.1), the second algorithm proposed by Krishna Reddy and Kitsuregawa is based on dense bipartite subgraphs, i.e., not all possible edges are required to exist (subsection 5.3.2).

Using the resulting dense bipartite graphs, it is possible to expand the corresponding set of vertices to entire subgraphs of desired size. To do so, consider any bipartite graph $G'$ defined on the set $H$ of hub pages (i.e., vertices with outgoing links w.r.t. $G'$) and the set $A$ of authority pages (i.e., vertices with ingoing links w.r.t. $G'$). Kumar et al. [KPRT99a] suggest to build a candidate set $S$ that consists of $H \cup A$ plus all pages that are referenced from pages in $H$, or link to at least two pages in $A$. As a next step, it is proposed to derive an order on the vertices of $S$, by applying the HITS algorithm (see subsection 2.3.2) and using the resulting hub and authority values. Choosing the best $k$ vertices w.r.t. this order, we get the desires web community of size $k$ (dense induced subgraph on $k$ vertices). The next two subsection explain the two different concepts of detecting the initial bipartite graphs.

### 5.3.1 Using complete bipartite graphs

As mentioned above web communities are characterized by dense directed bipartite subgraphs [KPRT99b]. Kumar et al. use the following hypothesis:

> *A random large enough and dense enough bipartite subgraph of the Web almost surely has a core,*[2] *where a core is defined to be a small* $(i, j)$*-sized complete bipartite subgraph, for some values* $i, j \in \mathbb{N}$.

Based on this idea Kumar et al. present the following algorithm to extract $(i, j)$-cores from the graph representing the hyperlink structure of the WWW.

[2] Misleadingly, Kumar et al. have proposed that every random bipartite graph $G = (L \cup R, E)$ with $|L| = |R| = 10$ and $|E| = 50$ has an $(i, j)$-core, with $i, j \geq 5$, with probability more than 0.99. This threshold can be seen to be too large for general random graphs, since even the expected number $E_{5,5}$ of complete bipartite graphs with vertex sets of size 5 in such a random graph calculates to (due to linearity of the expectation):

$$E_{5,5} = \binom{10}{5}\binom{10}{5} \cdot \frac{\binom{25}{25}\binom{75}{25}}{\binom{100}{50}} \approx 0,33 \cdot 10^{-4}.$$

Similarly, using the same number of edges, we get a expectations of approx. 0.161 resp. 19.0 for $(4, 4)$ resp. $(3, 3)$-cores. Therefore, we only can assure the occurrence of $(3, 3)$-cores (and smaller). However, in real-world date, where we (i) can assume higher edge density within the dense bipartite graphs and (ii) link behavior similar to preferential attachment (see paragraph 2.2.2.1), it has been observed that $(i, j)$-cores with $i, j \geq 5$ are a good indicator for communities [KPRT99a].

The algorithm is split in two parts:

- First of all, a preprocessing step cleans the input data. The key idea of this step is to remove duplicates and to eliminate very attractive webpages (so called potential fans, indicated by very large in-degrees), e.g. Yahoo, Netscape, Microsoft Internet Explorer, etc. Due to their high linkage from different communities these pages would distort the quality of the resulting communities. Further an additional process referred to as shingling is applied in order to eliminate duplicates containing minor changes. Applying these techniques Kumar et al. report that about 60% of the pages are removed.

- The second step attempts to extract the cores within the remaining webpages. These pages are stored, using two lists $A$ and $B$, where $A$ is sorted according to the out-degrees of the pages and $B$ according to the in-degrees. Initially, each page is contained once in each list.

  Pages with out-degree less than $j$ resp. in-degree less than $i$ cannot be contained in an $(i, j)$-core. Therefore, at any time throughout the second step, all entries in $A$ and $B$ with too small degree (i.e, less than $i$ for list $B$ and less than $j$ for list $A$) are removed. This process is iterated until non of these degrees remain.

  The main part of step two, iteratively removes a page $p$ with lowest admissible out-degree $j$ and tests if all $j$ neighbors of $p$ share a common neighborhood (w.r.t. in-links) of size at least $i$. If so, an $(i, j)$-core is found and is removed from the process. Otherwise, page $p$ is not contained in a core and can be eliminated from the process. Similarly to pages with out-degree $j$, it is possible to choose pages with in-degree $i$ .

  In each iteration step at least one page is removed. Further, due to the degree distribution of the graph, we have an average constant calculation cost per iteration. Therefore, if we assure that, throughout the whole process, there exists at least one page with least admissible in- or out-degree, the whole graph is processed and the total calculation time is linear in the number of pages. Due to the power-law behavior and the scale-free characteristic of the hyperlink structure of the WWW it may be assumed (and also has been observed by Kumar et al.) that there always exist vertices with degree $i$ resp. $j$. A more detailed discussion on all assumptions and reasoning for the algorithm is presented within the original paper [KPRT99b].

The resulting cores can be expanded to communities within the WWW (see above), which describe some same topic. In Tabular 5.1 we restate an example for best hubs and authorities of a community on Australian fire brigades (further examples are stated in [KPRT99a, KPRT99b]).

| Authorities | Hubs |
|---|---|
| NSW Rural Fire Service Internet Site | New South Wales ... Australian Links |
| NSW Fire Brigades | Feuerwehrlinks Australien |
| Sutherland Rural Fire Service | FireNet Information Network |
| Redirection Advice | The Cherrybrook ... Brigade |
| The National Cente...the Children's ... | Australian Fire Services Links |
| CRAFTI Internet Connexions- INFO | Fire Departments ... Information |
| Welcome to Blackwoo... Fire Safety Ser... | The Australian Firefighter Page |

Table 5.1: Example for hubs and authorities of web communities, based on the technique of Kumar et al. [KPRT99b]

### 5.3.2 Using dense bipartite graphs

The above approach of Kumar et al. has been refined by Krishna Reddy and Kitsuregawa [KRK01a, KRK01b] who required only dense (instead of complete) bipartite graphs. In the content of the Web, they consider community as a group of content creators that manifests itself as a set of interlinked pages. Further they abstract a community as a set of pages that form a dense bipartite graph. In order to formalize their approach they define a *dense bipartite graph* (DBG) as follows:

> *Let $p$ and $q$ be nonzero integer variables. A* DBG$(T, I, p, g)$ *is a bipartite graph on the vertex sets $T$ and $I$, where*
>
> *(i) each node of $T$ establishes an edge with at least $p$ ($1 \leq p \leq |I|$) nodes of $I$, and*
>
> *(ii) at least $q$ ($1 \leq q \leq |T|$) nodes of $T$ establish an edge with each node of $I$.*

Further they define a *dense bipartite graph set* DBGS$(r, s)$ to be the set of all DBG$(T, I, p, g)$ with $p \geq r$ and $s \geq q$. Obviously any graph corresponding to an $(i, j)$-core (see approach of Kumar et al.) is contained in DBGS$(j, i)$.

Based on this definition of DBGS, Krishna Reddy and Kitsuregawa define the following algorithm to extract most of the occurrences of corresponding subgraphs. Starting with some initial page $p$ they determine a dense bipartite graph that represents the community page $p$ is contained in, if existent. Applying the algorithm for all webpages, it is possible to enumerate a large set of dense bipartite subgraphs of the hyperlink graph.

- First of all, a set of pages that are related to the initial page $p$, is gathered. This process is based on the following definitions.

  For any pair of webpages $(p_i, p_j)$ we define that $p_i$ *cocites* $p_j$, if the number of webpages, both pages point to, is greater or equal to some given threshold.

| $(p,q)$ | # of DBG$(T,I,p,q)$ | avg$(T)$ | avg$(I)$ |
|---|---|---|---|
| (2,3) | 110422 | 36.21 | 162.60 |
| (2,4) | 81135 | 36.98 | 109.65 |
| (2,5) | 61566 | 36.15 | 83.47 |
| (3,3) | 90129 | 32.86 | 192.00 |
| (3,4) | 59488 | 32.26 | 140.56 |
| (3,5) | 40708 | 30.17 | 114.93 |
| (4,3) | 66670 | 34.29 | 244.81 |
| (4,4) | 49051 | 27.75 | 159.62 |
| (4,5) | 32309 | 24.97 | 134.33 |
| (5,5) | 28296 | 21.07 | 145.09 |
| (6,6) | 17335 | 19.03 | 161.67 |
| (7,7) | 10960 | 18.97 | 198.17 |

Table 5.2: Number of dense DBG$(T,I,p,q)$ and average number of pages in $T$ and $I$ on the TREC data collection according to [KRK01a]

Similarly, a single web page $p_i$ *cocites a set of webpages* $S$, if it holds that $p_i$ cocites $p_j$, for all $p_j \in S$.

The set of related pages $T$ is defined using the following iteration (for some predefined value *max-iterations*):

(a) Set $T = \{p\}$

(b) While *num-iterations* $\leq$ *max-iterations*

   i. For some given threshold, find all web pages $p_i$ that cocite with set $T$.

   ii. $T = \{p_i\} \cup T$.

(c) Output $T$.

- Consider the bipartite graph that is defined on the set $T$ and the set $I$ (i.e., all pages that are linked from $T$). Iteratively, all vertices with induced degree less than the global values $r$ and $s$ are removed (similar to the algorithm of Kumar et al.). Finally, the converged hyperlink graph, if it exists, is returned.

Using this process it is possible to related every webpage to a dense bipartite graph. Krishna Reddy and Kitsuregawa verified the algorithm on a hyperlink graph on 1.7 million pages and 21.5 million links provided by TREC (Text REtrieval Conference). Table 5.2 recites the number of corresponding subgraphs DBG$(T,I,p,q)$ for different thresholds $p,q$.

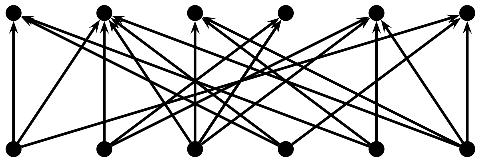

Figure 5.1: Community example "Comedy" represented by DBG(6, 6, 3, 3) without a (3, 3)-core [KRK01a]

Examing the dense bipartite graphs returned by their algorithm, Krishna Reddy and Kitsuregawa have compared the results to the heuristic that has been proposed by Kumar et al. (see above). Different to the suggestion of Kumar et al., some of the detected dense bipartite graphs, which sufficiently represent a community (evaluated by examing the links), do not contain a corresponding complete bipartite graph. As a case in point, Figure 5.1 illustrates the graph that corresponds to a community on the topic "Comedy" that is represented by a DBG(6, 6, 3, 3) but does not contain a (3, 3)-core. However, there is no detailed information on the ratio of dense bipartite graph representing a community that is not represented by some complete bipartite graph. Therefore, it is difficult to measure the overall improvement of this heuristic. Especially, when taking into consideration that the algorithm has super-linear runtime (dependent on the choice of the thresholds), both algorithms cannot be adequately compared.

## 5.4 Summary

In this section we have analyzed the approximability of the MAX-DENSE-$k$-SUBGRAPH-PROBLEM for general and $\beta$-PL input graphs. Within the discussion, we have proven equivalence of both problems with respect to $AP$-reduction, and thus, we have derived similar approximation results. While it is not known whether there exists a $\mathcal{PTAS}$ for these problems, it is assumed that the best approximation ratio is polynomial in the number of vertices of the input graph [FKP01].

Based on this assumed poor approximation ratios we have presented two heuristics from the literature [KPRT99b, KRK01a] in order to show that, when including further properties of the underlying graphs, it is possible to develop fast and efficient algorithms to detect dense subgraphs.

# Chapter 6

# Conclusion

## 6.1 Summary of results

In this thesis, a general analysis on the complexity of density-based clustering has been performed, where the clustering problem has been approached as the problem of detecting a subgraph on a given number of vertices with at least some corresponding number of edges. Within this discussion, we have focused on the class of power-law graphs, which has been shown to be a general abstraction of many large-scale networks. While the general problem of detecting the densest subgraph on some given number of vertices is known to be **NP**-hard, the complexity, if requiring only some suitable proportion of the possible edges within the subgraph (denoted by the problem $\gamma$-DSP), has not been classified entirely, so far.

The main contribution of this thesis is the almost complete classification of the computational complexity of the fixed-density decision problem $\gamma$-DSP (see chapter 4). Dependent on the function $\gamma$ that describes the required number of edges of a subgraph on $k$ vertices, in order to be a valid solution, we have derived the following results. For all functions $\gamma \in k + O(1)$ the problem is solvable in time polynomial in the number of vertices of the input graph, whereas the problem is **NP**-complete, for general graphs and $\gamma \in k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$. I.e., even the existence of a subgraph on $k$ vertices with average degree at least $2 + \varepsilon'$, with $\varepsilon' > 0$, can not be decided in time polynomial in the number of vertices of the input graph (unless $\mathbf{P} = \mathbf{NP}$). Additionally, we have derived that it is very likely that the problem is not **NP**-complete for $\gamma \in k + k^{o(1)}$ (based on an algorithm with subexponential time complexity). When restricting to $\beta$-PL graphs, we have also proven a lower-bound for **NP**-completeness linear in $k$, namely $\gamma(k) \geq \frac{15}{11}\delta \cdot k$ (i.e., subgraphs with average degree at least $\frac{30}{11}\delta$). The constant factor depends on the maximum average degree $\delta$ of $\beta$-PL graphs (which is bounded by a constant that only depends on $\beta$) and can be significantly larger than factor $2+\varepsilon'$ for general input graphs. These different values result due to the unknown characteristic of subgraphs of general $\beta$-PL graphs. Research on the properties of this graph class is located in a different research area and thus has not been extensively considered within this thesis. However, as a first indication, we have proven a constant upper bound for the average degree of any subgraph of a $\beta$-PL graph. The corresponding value, which is still too large to improve the constant factor, has been used for deriving the approximation results stated below. The actual lower bound for **NP**-completeness of $\gamma$-DSP on $\beta$-PL graphs has been achieved by using some sophisticated analysis on the degree sequence. Within our discussion, we initially found evidence that the lower bound for **NP**-completeness of $\gamma$-DSP on $\beta$-PL graphs is similar to that for general input graphs, unless every $\beta$-PL instance has a trivial solution. Therefore, we summarize that even if restricting to power-law graphs, the problem of detecting dense subgraphs with $\gamma \in k + \Omega(k^{\varepsilon})$, with $\varepsilon > 0$, is considered to be computationally hard.

In chapter 5, we have discussed the problem of approximating the densest subgraph on some given number of vertices. Once again, analogous to $\gamma$-DSP, we have proven that the problem is contained in similar approximation classes when the input is chosen from either general graphs or $\beta$-PL graphs. Since the problem on general input graphs has already been extensively discussed in the literature, our result has enabled to derive so far unknown approximation results when restricting to $\beta$-PL graphs. Thus, we know that the densest subgraph can be approximated in $O(n^{\frac{1}{3}-\varepsilon})$, where $n$ is the number of vertices of the input graph. Further, we may assume that there exists no constant approximation ratio. However, it is still open whether the problem can be approximated within $1 + \varepsilon$.

Finally, we can conclude that density-based clustering with size constraints in large-scale networks is inherently hard. In most cases, deciding on the existence on dense subgraphs on a given number of vertices and fixed density is **NP**-complete. Similarly, when searching some densest subgraph on a given number of vertices, it is supposed that no optimization algorithm with constant approximation ratio exists. The best currently known approximation ratio is polynomial in the number of vertices. However, we have been able to stress the importance of using properties, different to power-law (e.g. specific substructures), in order to construct good heuristics for finding density-based clusterings in large-scale networks.

## 6.2 Future work

In the following, we list some possible extentions to the work that has been done in this thesis.

In the case of general input graphs, the remaining gap for the computational complexity of $\gamma$-DSP could be examined and possibly reduced in size. For instance, a classification of the problem for functions $\gamma \in k + \Theta(\log k)$ could give more information on the dichotomy of the problem. Similarly, further discussion or even introduction of additional graph transformations might result in tighter lower bounds for **NP**-completeness of the problem.

When restricting the problem to $\beta$-PL graphs, although we have evidence on the characterization of the corresponding complexity, it would be worthwhile to state some improved, theoretical precise lower bound. This discussion could also help to better understand the nature of power-law graphs. We suppose that a good way to decrease the lower bound for **NP**-completeness is to derive advanced construction methods for power-law graphs that bound the average degree of corresponding subgraphs (similar to our approach within this thesis). Further, some deeper analysis on the degree sequence of power-law graphs, e.g. by applying improved summation techniques, could also result to some smaller lower bound.

Finally, as briefly mentioned in the discussion of approximability, in spite of expecting only polynomial approximation ratios, further analysis of approximation techniques could lead to smaller exponents, in particular when restricting to $\beta$-PL input graphs. Based on the properties of the degree sequence, it may be possible to either bound the value of the optimum solution or to improve the quality of the derived approximation results. While it might not be feasible to apply this technique in general, we may aim to improve the results for different ranges of the subgraph size (e.g., for very large subgraphs, we can restrict to remove vertices of degree one and thus derive optimal solution). If it is possible to cover the whole range of $k$, these results could be combined in order to derive some general improvement.

# Appendix A

# General definitions

## A.1 Graphs

### (Undirected) Graphs

An *(undirected) graph* $G$ consists of a set of vertices $V$ and a set of edges $E \subseteq V^2$ (denoted by $G = (V, E)$). If we want to define some orientation on the edges, we use *directed graphs* with $E \subseteq V \times V$ (i.e., edge $(v, w) \in E$ represents an edge from $v$ pointing to $w$). A *simple graph*, is an undirected graph without multiple edges and self-loops (i.e., edges that start and end at the same vertex). In this thesis, unless stated otherwise, we consider undirected simple graphs.

### Subgraphs

A *subgraph* $H = (V_H, E_H)$ of some graph $G = (V_G, E_G)$ is a graph with $V_H \subseteq V_G$ and $E_H \subseteq E_G$. It is not required that $E_H = E_G \cap V_H^2$ (i.e., it is not required that all possible edges between vertices in $V_H$ must exist in $H$, if they exist in $G$). However, when referring to the *induced* subgraph, the existence of all possible edges that are also element of $E_G$ is required. Let $G(V, E)$ be a graph and $V' \subseteq V$. The subgraph $G'$ of $G$ induced by $V'$ is defined to $G' = (V', E')$, with $E' = E \cap V'^2$. If $G$ is clear from the context we also refer to $G'$ as the induced subgraph of $V'$.

### Sets of vertices and edges

For any graph $G$, we use $V(G)$ and $E(G)$ to denote the sets of vertices and edges of $G$. For the cardinality of a set $S$ we use $|S|$, e.g., $|V(G)|$ and $|E(G)|$ are the cardinalities of the sets of vertices and edges of $G$. For any simple graph $G$, it holds that $0 \leq |E(G)| \leq \binom{|V(G)|}{2}$.

### Neighbors and degree

Let $G = (V, E)$ be an undirected graph. The set of *neighbors* $N(v)$ of a vertex $v \in V$ is defined to $N(v) = \{ w \mid \{v, w\} \in E \}$. Further the *degree* of a vertex $v$ is number of its neighbors and is denoted by $\deg(v) = |N(v)|$. The *average* *degree* avgdeg$(S)$ of a set $S$ of vertices is defined to be the average value of the degrees of all vertices in $S$. Similarly, the average degree avgdeg$(G)$ of a graph $G$ equals the average degree of $V(G)$. For a vertex $v$ of a directed graph the in-degree resp. out-degrees is defined to the number of edges $(v, w) \in E(G)$ resp. $(w, v) \in E(G)$, with $w \in V(G)$. Similarly, we define the in/out neighbors and the average in/out-degree.

### Connected components

Let $G = (V, E)$ be an undirected graph. Two vertices $v, w \in V$ are referred to be *connected*, if there exist vertices $v_1, \dots, v_k$, with $k \leq |V| - 2$, in such a way that $\{v, v_1\}, \{v_i, v_{i+1}\}, \{v_k, w\} \in E$, for $1 \leq i \leq k-1$. Further, it is always possible to partition set $V$ into subsets $V_1, \dots, V_l$ (i.e., $V_i \cap V_j = \emptyset$, for all $i \neq j$, and $V_1 \cup \dots \cup V_l = V$) in such a way that $v \in V_i$ and $w \in V_j$ are connected iff $i = j$. The subgraphs induced of sets $V_i$, for $1 \leq i \leq l$, are referred to as *connected components*. A graph is connected if all possible pairs of vertices are connected, or, equivalently, it has only one connected component.

### Isomorphism

Two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$ are said to be *isomorphic* (denoted by $G \cong H$) if and only if there exists a bijection $f : V_G \to V_H$ in such a way that

$$\{v, w\} \in E_G \iff \{f(v), f(w)\} \in E_H.$$

### Breadth first search (BFS)

A *breadth first search* (BFS) is an algorithm that is used to traverse a connected graph (resp. connected component). Throughout the whole algorithm, we keep a first-in-first-out queue that initially contains a single start vertex. The process iterates until the queue is empty. Within each iteration, the next vertex $v$ from the queue is removed and all so far not visited neighbors of $v$ are added to the queue. Similarly, we define a *parallel BFS* to be the corresponding algorithm whose queue initially contained several start vertices.

After running either of the algorithms all vertices of the graph (resp. connected component) have been visited exactly once, while every edge has been considered at most twice. Therefore, the run time of the algorithms on a connected graph (resp. connected component) $G$ is proportional to $|V(G)| + |E(G)|$.

## A.2 Computational complexity

Throughout the whole thesis, when considering the complexity of a problem, we discuss the time complexity of the problem assuming logarithmic costs (i.e., bit complexity and Turing machines).

### Complexity classes P and NP, reduction

The complexity class of problems computable in polynomial time of the input size is denoted as **P** , while the set of problems computable in non-deterministic polynomial time is referred to as **NP** . For two problems $A, B \in \mathbf{NP}$ we define that $A$ can be reduced to $B$ if there exists a polynomial-time many-one reduction of $A$ to $B$ (i.e., there exists a function $f$ computable in polynomial time such that for all $x$ it holds that $x \in A \Leftrightarrow f(x) \in B$). Based on this type of reduction, a problem $A$ is said to be **NP**-complete (also referred to as **NP**-c), if every problem in **NP** can be reduced to the $A$. Two well-known **NP**-complete problems are SATISFIABILITY (SAT) [Coo71] and CLIQUE [Kar72] (for more examples see also [GJ79]).

### Landau symbols

In order to describe the asymptotic growth of a function $f : \mathbb{N} \to \mathbb{R}$ we use the Landau symbols:

$$O,\ \Omega,\ \Theta,\ o, \text{and } \omega$$

Let $f, g : \mathbb{N} \to \mathbb{R}$ be two functions. The sets of functions $O(g)$, $\Omega(g)$, $\Theta(g)$, $o(g)$, and $\omega(g)$ are defined as follows:

$$\begin{array}{lcl}
f \in O(g) & \Leftrightarrow & (\ \exists\, c \in \mathbb{R}\ \ \exists\, n_o \in \mathbb{N}\ \ \forall\, n \geq n_o\ )\ \ [\ |f(n)| \leq |c \cdot g(n)|\ ] \\
f \in \Omega(g) & \Leftrightarrow & (\ \exists\, c \in \mathbb{R}\ \ \exists\, n_o \in \mathbb{N}\ \ \forall\, n \geq n_o\ )\ \ [\ |f(n)| \geq |c \cdot g(n)|\ ] \\
f \in \Theta(g) & \Leftrightarrow & \ \ f \in O(g)\ \ \wedge\ \ f \in \Omega(g) \\
f \in o(g) & \Leftrightarrow & (\ \forall\, c \in \mathbb{R}\ \ \exists\, n_o \in \mathbb{N}\ \ \forall\, n \geq n_o\ )\ \ [\ |f(n)| < |c \cdot g(n)|\ ] \\
f \in \omega(g) & \Leftrightarrow & (\ \forall\, c \in \mathbb{R}\ \ \exists\, n_o \in \mathbb{N}\ \ \forall\, n \geq n_o\ )\ \ [\ |f(n)| > |c \cdot g(n)|\ ]
\end{array}$$

Instead of defining functions $f$ and $g$ explicitly, we use implicit declarations, e.g. $2n \in O(n^2)$ in order to express that $f \in O(g)$, with $f(n) = 2n$ and $g(n) = n^2$. Further examples are: $0.001\sqrt{n} \in \Omega(n^{0.25})$, $(n+2)^3 \in \Theta(n^3)$, $log(n) \in o(n)$, and $n^2 + log(n) \in \omega(n)$.

Further, once again assuming $f$, $g$, and $h$ to be functions $\mathbb{N} \to \mathbb{R}$, we use the following abbreviation (only stated for Landau symbol $O$, the sets for all other Landau symbols are defined accordingly).

$$f \in g + O(h) \quad \Leftrightarrow \quad (\ \exists\, c \in \mathbb{R}\ \exists\, n_o \in \mathbb{N}\ \forall\, n \geq n_o\ )\ [\ |f(n)| \leq |g(n)| + |c \cdot h(n)|\ ]$$

Using this definition it is easy to see that, e.g., it holds $n + 141 \in n + O(1)$ and $1.5\,n \in n + \Omega(n^{\frac{1}{2}})$.

# Appendix B

# Optimization Problems

Within this thesis, we use the following formal definitions and lemmata on optimization problems (adopted from [Ste03]). A more detailed description is given in standard text books [ACG$^+$99, MPS98, Vaz01].

**Definition B.1** *An optimization problem* $\Pi$ *is a four-tuple* $\langle \mathcal{I}, \mathcal{S}ol, \mathrm{val}, \mathrm{goal}\rangle$ *such that, for some alphabet* $\Sigma$

- $\mathcal{I} \subseteq \Sigma^*$ *is the set of instances.*
- *For every instance* $I \in \mathcal{I}, \mathcal{S}ol(I) \subseteq \Sigma^*$ *denotes the set of solutions of* $I$ *and is non-empty.*
- *For every instance* $I \in \mathcal{I}$ *and solution* $x \in \mathcal{S}ol(I)$*, the value* $\mathrm{val}(I, x)$ *is a positive integer. The function* $\mathrm{val}(\,\cdot\,;\,\cdot\,)$ *is called the objective function.*
- $\mathrm{goal} \in \{\min, \max\}$.

**Definition B.2** *An optimization problem* $\langle \mathcal{I}, \mathcal{S}ol, \mathrm{val}, \mathrm{goal}\rangle$ *belongs to the class* $\mathcal{NPO}$ *iff*

- $\mathcal{I} \subseteq \Sigma^*$ *is a set of instances that is recognizable in linear time.*
- *The size of the solutions is polynomially bounded in the length of* $I$*, i.e., there exists a polynomial* $p$ *such that*

$$|x| \leq p(|I|) \qquad \textit{for all } I \in \mathcal{I} \textit{ and } x \in \mathcal{S}ol(I).$$

- *The question "Is* $x \in \mathcal{S}ol(I)$*?" is decidable in polynomial time.*
- *The function* $\mathrm{val}(\,\cdot\,;\,\cdot\,)$ *is computable in polynomial time.*

Based on these definitions, it is possible to define the sets $\mathcal{APX}$, $\mathcal{PTAS}$, and $\mathcal{FPTAS}$ of optimization problems.

**Definition B.3** *$\mathcal{APX}$ is the set of all optimization problems in $\mathcal{NPO}$ which admit a polynomial time approximation algorithms with performance ratio $\rho$ for some constant $\rho \geq 1$.*

**Definition B.4** *$\mathcal{PTAS}$ is the set of all optimization problems in $\mathcal{NPO}$ which admit a polynomial time approximation scheme.*

**Definition B.5** *$\mathcal{FPTAS}$ is the set of all optimization problems in $\mathcal{NPO}$ which admit a fully polynomial time approximation scheme.*

Within this thesis, when comparing the approximation classes of two optimization problems, we use the following reduction (referred to as $AP$-reduction).

**Definition B.6** *An Optimization problem $\Pi = \langle \mathcal{I}, \mathcal{S}ol, \mathrm{val}, \mathrm{goal}\rangle$ is $AP$-reducible to an optimization problem $\Pi^* = \langle \mathcal{I}^*, \mathcal{S}ol^*, \mathrm{val}^*, \mathrm{goal}^*\rangle$, referred to as $\Pi \leq_{AP} \Pi^*$, if and only if there exist functions $f$ and $g$ and constant $\alpha > 0$ such that:*

- *(AP1) For any $\delta > 0$, and for any $I \in \mathcal{I}$, it holds $f(I, \delta) \in \mathcal{I}^*$.*
- *(AP2) For any $\delta > 0$, for any $I \in \mathcal{I}$, and $y \in \mathcal{S}ol^*(f(I), \delta)$, it holds $g(I, y, \delta) \in \mathcal{S}ol(I)$.*
- *(AP3) For any fixed $\delta > 0$, the functions $f$ and $g$ are computable in polynomial time.*
- *(AP4) For any $I \in \mathcal{I}$, and for any $\delta > 0$, and for any $y \in \mathcal{S}ol^*(f(I))$,*

$$\frac{1}{1+\delta} \leq \frac{\mathrm{val}(f(I,\delta),y)}{\mathrm{opt}(f(I,\delta))} \leq 1+\delta \quad \Rightarrow \quad \frac{1}{1+\alpha\cdot\delta} \leq \frac{\mathrm{val}(I,g(I,y,\delta))}{\mathrm{opt}(I)} \leq 1+\alpha\cdot\delta.$$

*The triple $(f, g, \alpha)$ is an $AP$-reduction from $\Pi$ to $\Pi^*$.*
*If $\Pi \leq_{AP} \Pi^*$ and $\Pi^* \leq_{AP} \Pi$, we say that $\Pi$ is equivalent to $\Pi^*$ w.r.t. $AP$-reduction (denoted by $\Pi \equiv_{AP} \Pi^*$).*

Using this definition of $AP$-reduction the following two lemmata hold.

**Lemma B.1** *Let $\Pi, \Pi^* \in \mathcal{NPO}$. If $\Pi^* \in \mathcal{APX}$ and $\Pi \leq_{AP} \Pi^*$ then $\Pi \in \mathcal{APX}$.*

**Lemma B.2** *Let $\Pi, \Pi^* \in \mathcal{NPO}$. If $\Pi^* \in \mathcal{PTAS}$ and $\Pi \leq_{AP} \Pi^*$ then $\Pi \in \mathcal{PTAS}$.*

# Bibliography

[AB02] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Review of Modern Physics*, 74(1):47–97, 2002.

[ABJ99] R. Albert, A.-L. Barabasi, and H. Jeong. Diameter of the World Wide Web. *Nature*, 401:130–131, 1999.

[ACG$^+$99] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation.* Springer, 1999.

[ACL01] W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.

[ACL02] W. Aiello, F. Chung, and L. Lu. *A random graph model for massive graphs*, chapter 4, pages 97–122. Kluwer Academics, 2002.

[Adm99] L. Admic. The small world wide web. In *Proceedings of the 3rd European Conference on Digital Libraries (ECDL'99)*, volume 1696 of *Lecture Notes in Computer Science, LNCS*, pages 443–452. Springer, 1999.

[AF99] N. Alon and E. Fischer. Refining the graph density condition for the existence of almost k-factors. *Ars Combinatoria*, 52:296–308, 1999.

[AHI02] Y. Asahiro, R. Hassin, and K. Iwama. Complexity of finding dense subgraphs. *Discrete Applied Mathematics*, 121(1-3):15–26, 2002.

[AITT00] Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama. Greedily finding a dense subgraph. *Journal of Algorithms*, 34(2):203–221, 2000.

[AK95] C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: A survey. *Integration: The VLSI Journal*, 19(1–2):1–81, 1995.

[AKK99] S. Arora, D. Karger, and M. Karpinski. Polynomial time approximation schemes for dense instances of NP-hard problems. *Journal of Computer and System Sciences*, 58(1):193–210, 1999.

[Alo86] N. Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, 1986.

[ALPH01] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman. Search in power-law networks. *Physical Review E*, 64(4), 2001. Article 046135.

[ASBS00] L. A. N. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley. Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):284–293, 2000.

[BA99] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[BAJ99] A.-L. Barabasi, R. Albert, and H. Jeong. Mean-field theory for scale-free random networks. *Physica A*, 272(1–2):173–187, 1999.

[BAJ00] A.-L. Barabasi, R. Albert, and H. Jeong. Scale-free characteristics of random networks: The topology of the world-wide web. *Physica A*, 281(1–4):69–77, 2000.

[BBCR03] B Bollobás, C. Borgs, J. Chayes, and O. Riordan. Directed scale free graphs. In *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms (SODA 2003)*, pages 132–139. Society for Industrial and Applied Mathematics, 2003.

[BDJ99] M. W. Berry, Z. Drmac, and E. R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Review*, 41(2), 1999.

[BDO95] M. W. Berry, S. T. Dumais, and G. W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4), 1995.

[Ber58] C. Berge. *Théorie des graphes et ses applications*. Dunod, 1958.

[Ber03] M. W. Berry. *Survey of Text Mining: Clustering, Classification, and Retrieval*. Springer, 2003.

[BMZ99] V. Batagelj, A. Mrvar, and M. Zaversnik. Partitioning approach to visualization of large graphs. In *Graph Drawing: 7th International Symposium (GD'99)*, volume 1731 of *Lecture Notes in Computer Science, LNCS*, pages 90–97. Springer, 1999.

[BO03] G. Buckley and D. Osthus. Popularity based random graph models leading to a scale-free degree distribution, 2003. to appear in Discrete Mathematics.

[Bol80] B. Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal on Combinatorics*, 1:311–316, 1980.

[Bol85] B. Bollobás. *Random Graphs*. Academic Press, 1985.

[BP98] S. Brin and L. Page. The anatomy of a large-scale hypertextual (web) search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[BR02a] B. Bollobás and O. Riordan. The diameter of a scale-free random graph. *Combinatorica*, 2002. to appear.

[BR02b] B. Bollobás and O. Riordan. Mathematical results on scale-free random graphs. In *Handbook of Graphs and Networks: From the Genome to the Internet*, pages 1–34. Wiley-VHC, 2002.

[BRM$^+$00] A. Broder, Kumar. R., F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the Web. In *Proceedings of the Ninth International World Wide Web Conference/Computer Networks*, volume 33, pages 1–6. Elsevier, 2000.

[BRST01] B. Bollobás, O. Riordan, J. Spencer, and G. Tusnády. The degree sequence of a scale-free random graph process. *Random Structures and Algorithms*, 18(3):279–290, 2001.

[BS02] S. Bornholdt and H. G. Schuster, editors. *Handbook of Graphs and Networks*. Wiley-VCH, 2002.

[CEbAH00] R. Cohen, K. Erez, D. ben Avraham, and S. Havlin. Resilience of the internet to random breakdown. *Physical Review Letters*, 85(21):4626–4628, 2000.

[CF03] C. Cooper and A. Frieze. A general model of web graphs. *Random Structures and Algorithms*, 22(3):311–335, 2003.

[CH03] R. Cohen and S. Havlin. Scale free networks are ultrasmall. *Physical Review Letters*, 90(5), 2003. Article 058701.

[Cha00] S. Chakrabarti. Data mining for hypertext: a tutorial survey. *SIGKDD Explorations: Newsletter of the Special Interest Group on Knowledge Discovery & Data Mining*, 1(2):1–11, 2000.

[CHK91] J. Cong, L. Hagen, and A. B. Kahng. Random walks for circuit clustering. In *4th IEEE International ASIC Conference (ASIC'91)*, pages 14.2.1–14.2.4, 1991.

[CL70] D. Cvetković and R. P. Ličić. A new generalization of the concept of the p-sum of graphs. *Univerzitet u Beogradu Publikacije Elektrotehničkog Fakulteta, Serija Matematika, Univerzitet Beograd, Belgrade*, 302:67–71, 1970.

[CL01] F. Chung and L. Lu. The diameter of random sparse graphs. *Advances in Applied Mathematics*, 26:257–279, 2001.

[CLV03] F. Chung, L. Lu, and V. Vu. Eigenvalues of random power law graphs. *Annals of Combinatorics*, 7:21–33, 2003.

[Coo71] S. A. Cook. The complexity of theorem-proving procedures. In *Conference record of third annual ACM Symposium on Theory of Computing (STOC'71)*, pages 151–158. ACM Press, 1971.

[CSZ93] P. K. Chan, M. D. F. Schlag, and J. Y. Zien. Spectral $k$-way ratio-cut partitioning and clustering. In *Proceedings of the 30th ACM/IEEE Design Automation Conference (DAC'93)*, pages 749–754. ACM Press, 1993.

[Czy00] A. Czygrinow. Maximum dispersion problem in dense graphs. *Operations Research Letters*, 27(5):223–227, 2000.

[DCG99] R. De Castro and J. W. Grossman. Famous trails to paul erdos. *MATHINT: The Mathematical Intelligencer*, 21:51–63, 1999.

[DDSW03] J. Diaz, N. Do, M. J. Serna, and N. C. Wormald. Bounds on the max and min bisection of random cubic and random 4-regular graphs. *Theoretical Computer Scence.*, 307(3):531–547, 2003.

[DEM01] E. Drinea, M. Enachescu, and Mitzenmacher. M. Variations on random graph models for the web. technical report TR-06-01, Harvard University, Department of Computer Science, 2001.

[DFP93] M. Dyer, A. Frieze, and B. Pittel. The average performance of the greedy matching algorithm. *Annals of Applied Probability*, 3(2):526–552, 1993.

[DMS00] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin. Structure of growing networks with preferential linking. *Physical Review Letters*, 85(21):4633–4636, 2000.

[DMS01] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin. Size-dependent degree distribution of a scale-free growing network. *Physical Review E*, 63(6), 2001. Article 062101.

[ER59] P. Erdös and A. Rényi. On random graphs. *Publications Mathematical Debrecen*, 6:290–297, 1959.

[ESB99] J. Edachery, A. Sen, and F. J. Brandenburg. Graph clustering using distance-k cliques. In *Graph Drawing: 7th International Symposium (GD'99)*, volume 1731 of *Lecture Notes in Computer Science, LNCS*, pages 98–106. Springer, 1999.

[FDBV01] I. J. Farkas, I. Derenyi, A.-L. Barabasi, and T. Vicsek. Spectra of "real-world" graphs: Beyond the semi circle law. *Physical Review E*, 64(2), 2001. Article 026704.

[Fei02] U. Feige. Relations between average case complexity and approximation complexity. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC 2002)*, pages 534–543. ACM Press, 2002.

[Fel00] A. Feldmann. Characteristics of TCP Connection Arrivals. In *Self-Similar Network Traffic and Performance Evaluation*. Wiley, 2000.

[FF56] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.

[FFF99] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *ACM Special Interest Group on Data Communications (SIGCOMM'99)*, volume 29 of *Computer Communications Review*, pages 251–262, 1999.

[FK94] U. Faigle and W. Kern. Computational complexity of some maximum average weight problems with precedence constraints. *Operations Research*, 42(4):1268–1272, 1994.

[FKP01] U. Feige, G. Kortsarz, and D. Peleg. The dense $k$-subgraph problem. *Algorithmica*, 29(3):410–421, 2001.

[FL01] U. Feige and M. Langberg. Approximation algorithms for maximization problems arising in graph partitioning. *Journal of Algorithms*, 41:174–211, 2001.

[FLG00] G. W. Flake, S. Lawrence, and C. L. Giles. Efficient identification of web communities. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2000)*, pages 150–160, 2000.

[FLGC02] G. W. Flake, S. Lawrence, C. L. Giles, and F. M. Coetzee. Self-organization and identification of Web communities. *Computer*, 35(3):66–71, 2002.

[FM82] C. M. Fiduccia and R. M. Mattheyses. A linear-time heuristic for improving network partitions. In *19th Design Automation Conference (DAC'82)*, pages 175–181. ACM/IEEE, 1982.

[FS97] U. Feige and M. Seltser. On the densest $k$-subgraph problem. Technical Report CS97-16, Department of Applied Mathematics and Computer Science, The Weizmann Institute of Science, Rehovot, Israel, 1997.

[GGT89] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan. A fast parametric maximum flow algorithm and applications. *SIAM Journal on Computing*, 18(1):30–55, 1989.

[Gil59] E. N. Gilbert. Random graphs. *Annals of Mathematical Statistics*, 30:1141–1144, 1959.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability, A Guide to Theory of NP-Completeness*. Freeman, 1979.

[GKR98] D. Gibson, J. Kleinberg, and P. Raghavan. Inferring web communities from link topology. In *Proceedings of the 9th ACM Conference on Hypertext (Hypertext'98)*, Structural Queries, pages 225–234, 1998.

[GNY94] O. Goldschmidt, D. Nehme, and G. Yu. On the set union knapsack problem. *Naval Research Logistics*, 41(6):833–842, 1994.

[Gol84] A. V. Goldberg. Finding a maximum density subgraph. Technical Report UCB/CSB 84/171, Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, 1984.

[GPS90] J. Garbers, H. J. Prömel, and A. Steger. Finding clusters in VLSI circuits. In *Proceedings of the IEEE International Conference on Computer-Aided Design (ICCAD'90)*, pages 520–523. IEEE Computer Society Press, 1990.

[GVL89] G. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.

[GW95] M. X. Goemans and D. P. Williamson. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. *Journal of the Association for Computing Machinery*, 42(6):1115–1145, 1995.

[Hås99] J. Håstad. Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Mathematica*, 182(1):105–142, 1999.

[Hay00] B. Hayes. Graph theory in practice: Part 2. *American Scientist*, 88(2):104–109, March–April 2000.

[HK95] D. J.-H. Huang and A. B. Kahng. When cluster meet partitions: New density-based methods for circuit decomposition. In *European Design and Test Conference (EDTC'95)*, pages 60–64. IEEE Computer Society Press, 1995.

[HKMT02] K. Holzapfel, S. Kosub, M. Maaß, and H. Täubig. The complexity of detecting fixed-density clusters. Technischer Bericht TUM-I0212, Technische Universität München, Institut für Informatik, 2002.

[HKMT03] K. Holzapfel, S. Kosub, M. Maaß, and H. Täubig. The complexity of detecting fixed-density clusters. In *Algorithms and Complexity, 5th Italian Conference (CIAC 2003)*, number 2653 in Lecture Notes in Computer Science, LNCS, pages 201–212. Springer-Verlag, 2003.

[HRT97] R. Hassin, S. Rubinstein, and A. Tamir. Approximation algorithms for maximum dispersion. *Operations Research Letters*, 21(3):133–137, 1997.

[HS00] E. Hartuv and R. Shamir. A clustering algorithm based on graph connectivity. *Information Processing Letters*, 76(4-6):175–181, 2000.

[HYZ02] Q. Han, Y. Ye, and J. Zhang. An improved rounding method and semidefinite programming relaxation for graph partition. *Mathematical Programming*, 92(3):509–535, 2002.

[IMK$^+$03] S. Itzkovitz, R. Milo, N. Kastan, G. Ziv, and U. Alon. Subgraphs in random networks. *Physical Review E*, 68(2), 2003. Article 026127.

[JD88] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, 1988.

[JTA$^+$00] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000.

[Kar72] R. M. Karp. *Reducibility Among Combinatorial Problems*, pages 85–103. Plenum Press, 1972.

[Kho01] S. Khot. Improved inapproximability results for max clique, chromatic number and approximate graph coloring. In *Proceedings 42nd Annual Symposium on Foundations of Computer Science (FOCS 2001)*, pages 600–609. IEEE Computer Society Press, Washington, D.C., 2001.

[KKR$^+$99] J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins. The web as a graph: Measurements, models, and methods. In *Computing and Combinatorics:5th Annual International Conference (COCOON'99)*, volume 1627 of *Lecture Notes in Computer Science, LNCS*, pages 1–17. Springer-Verlag, 1999.

[KL70] B. W. Kernighan and S. Lin. An efficient heuristic for partitioning graphs. *Bell Systems Technical Journal*, 49:291–307, 1970.

[Kle99] J. M. Kleinberg. Authoritive Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

[Kle00] J. M. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proceedings 32nd Annual ACM Symposium on Theory of Computing (STOC 2000)*, pages 163–170. ACM Press, 2000.

[Kle02] J. S. Kleinfeld. Could it be a big world after all? the "six degrees of separation" myth. *Society*, April 2002.

[KP93] G. Kortsarz and D. Peleg. On choosing a dense subgraph. In *Proceedings 34th Symposium Foundations of Computer Science (FOCS'93)*, pages 692–703. Institute of Electrical & Electronics Engineers, 1993.

[KPRT99a] R. Kumar, Raghavan P., S. Rajagopalan, and A. Tomkins. Extracting large-scale knowledge bases from the web. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB'99)*, pages 639–650, 1999.

[KPRT99b] R. Kumar, Raghavan P., S. Rajagopalan, and A. Tomkins. Trawling the Web for emerging cyber-communities. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1481–1493, 1999.

[KRK01a] P. Krishna Reddy and M. Kitsuregawa. An approach to relate the web communities through bipartite graphs. In *Second International Conference on Web Information Systems Engineering (WISE 2001)*, pages 301–310, 2001.

[KRK01b] P. Krishna Reddy and M. Kitsuregawa. Inferring web communities through relaxed cocitation and dense bipartite graphs. In *Proceedings of the 12th Database Engineering Workshop (DEWS 2001)*, 2001.

[KRR$^+$00a] R. Kumar, P. Raghavan, R. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the Web graph. In IEEE, editor, *41st Annual Symposium on Foundations of Computer Science (FOCS 2000)*, pages 57–65. IEEE Computer Society Press, 2000.

[KRR$^+$00b] R. Kumar, P. Raghavan, R. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. The web as a graph. In *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2000)*, pages 1–10. ACM Press, 2000.

[KV02] M. Krivelevich and V. H. Vu. Approximating the independence number and the chromatic number in expected polynomial time. *Journal of Combinatorial Optimization*, 6(2):143–155, 2002.

[Len90] T. Lengauer. *Combinatorial Algorithms for Integrated Circuit Layout.* Wiley, 1990.

[LM00] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. *Computer Networks*, 33(1–6):387–401, 2000.

[Luc92] T. Luczak. Sparse random graphs with a given degree sequence. *Random Graphs*, 2:165–182, 1992.

[Mil67] S. Milgram. The small world problem. *Psychology Today*, 1(1):60–67, 1967.

[Mit03] M. Mitzenmacher. A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2), 2003. to appear.

[MP01] D. Magoni and J.-J. Pansiot. Analysis of the autonomous system network topology. *ACM SIGCOMM Computer Communication Review*, 31(3):26 – 37, 2001.

[MPS98] E. W. Mayr, H. J. Prömel, and A. Steger, editors. *Lectures on Proof Verification and Approximation Algorithms*, volume 1367 of *Lecture Notes in Computer Science.* Springer, 1998.

[New03] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[NMW00] M. E. J. Newman, C. Moore, and D. J. Watts. Mean-field solution of the small-world network model. *Physical Review Letters*, 84(14):3201–3204, 2000.

[NSW01] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2), 2001. Article 026118.

[NY97] D. Nehme and G. Yu. The cardinality and precedence constrained maximum value sub-hypergraph problem and its applications. *Discrete Applied Mathematics*, 74(1):57–68, 1997.

[Par96] V. Pareto. *Cours d'Economie Politique.* Univerité de Lausanne, 1896.

[Per02] P. Perner, editor. *Data Mining on Multimedia Data*, volume 2588 of *Lecture Notes in Computer Science, LNCS*. Springer, 2002.

[Ree03] W. J Reed. The pareto law of incomes - an explanation and an extension. *Physica A*, 319:469–486, 2003.

[RGW02] J. W. Raymond, E. J. Gardiner, and P. Willett. Heuristics for similarity searching of chemical graphs using a maximum common edge subgraph algorithm. *Journal of Chemical Information and Computer Sciences*, 42(2):305–316, 2002.

[RRT94] S. S. Ravi, D. J. Rosenkrantz, and G. K. Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, 1994.

[SCbA$^+$02] N. Schwartz, R. Cohen, D. ben Avraham, A.-L. Barabasi, and S. Havlin. Percolation in directed scale-free networks. *Physical Review E*, 66(1), 2002. Article 015104(R).

[SFFF03] G. Siganos, M. Faloutsos, P. Faloutsos, and C. Faloutsos. Power laws and the as-level internet topology. *IEEE/ACM Transaction on Networking*, 11(4):514–524, 2003.

[Ste03] A. Steger. Approximability of np-optimization problems. In B. A. Reed and C. Linhares-Sales, editors, *Recent advances in algorithms and combinatorics*, volume 10 of *CMS books in mathematics*, chapter 7. Springer, 2003.

[Str01] S. H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, 2001.

[SW98] A. Srivastav and K. Wolf. Finding dense subgraphs with semidefinite programming. In *Proceedings International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX'98)*, volume 1444 of *Lecture Notes in Computer Science, LNCS*, pages 181–191. Springer-Verlag, 1998.

[SW03] G. Sagie and A. Wool. A clustering approach for exploring the internet structure. Technical Report EES2003-7, Department of Electrical Engineering Systems, Tel Aviv University, Israel, 2003.

[Sze78] E. Szemerédi. Regular partitions of graphs. In *Proceedings of the Colloques Internationaux du Centre National de la Recherche Scientifique (CNRS)*, pages 399–402, 1978.

[Tur41] P. Turán. On an extremal problem in graph theory. *Matematikai és Fizikai Lapok*, 48:436–452, 1941. In Hungarian.

[Vaz01] V. Vazirani. *Approximation Algorithms*. Springer, 2001.

[vD00] S. van Dongen. *Graph Clustering by Flow Simulation*. Phd thesis, University of Utrecht, 2000.

[Viz63] V. G. Vizing. The cartesian product of graphs (in russian). *Vyčisl. Sistemy*, 9:30–43, 1963. (English translation: Comp. El. Syst. 2 (1966) 352-365).

[Wat99a] D. J. Watts. Kevin bacon, the small-world, and why it all matters. *Santa Fe Institute Bulletin*, 14(2), 1999.

[Wat99b] D. J. Watts. *Small Worlds: The Dynamics of Networks between Order and Randomness*. Princeton University Press, 1999.

[Wig58] E. Wigner. On the distribution of the roots of certain symmetric matrices. *Annals of Mathematics*, 67:325–327, 1958.

[WK88] D. W. Wang and Y. S. Kuo. A study on two geometric location problems. *Information Processing Letters*, 28(6):281–286, 1988.

[WS98] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:397–498, 1998.

[YDL94] S. B. Yang, S. K. Dhall, and S. Lakshmivarahan. A processor efficient connectivity algorithm on random graphs. *Parallel Processing Letters*, 4:29–36, 1994.

[Zip49] G. K. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley, 1949.

# Index