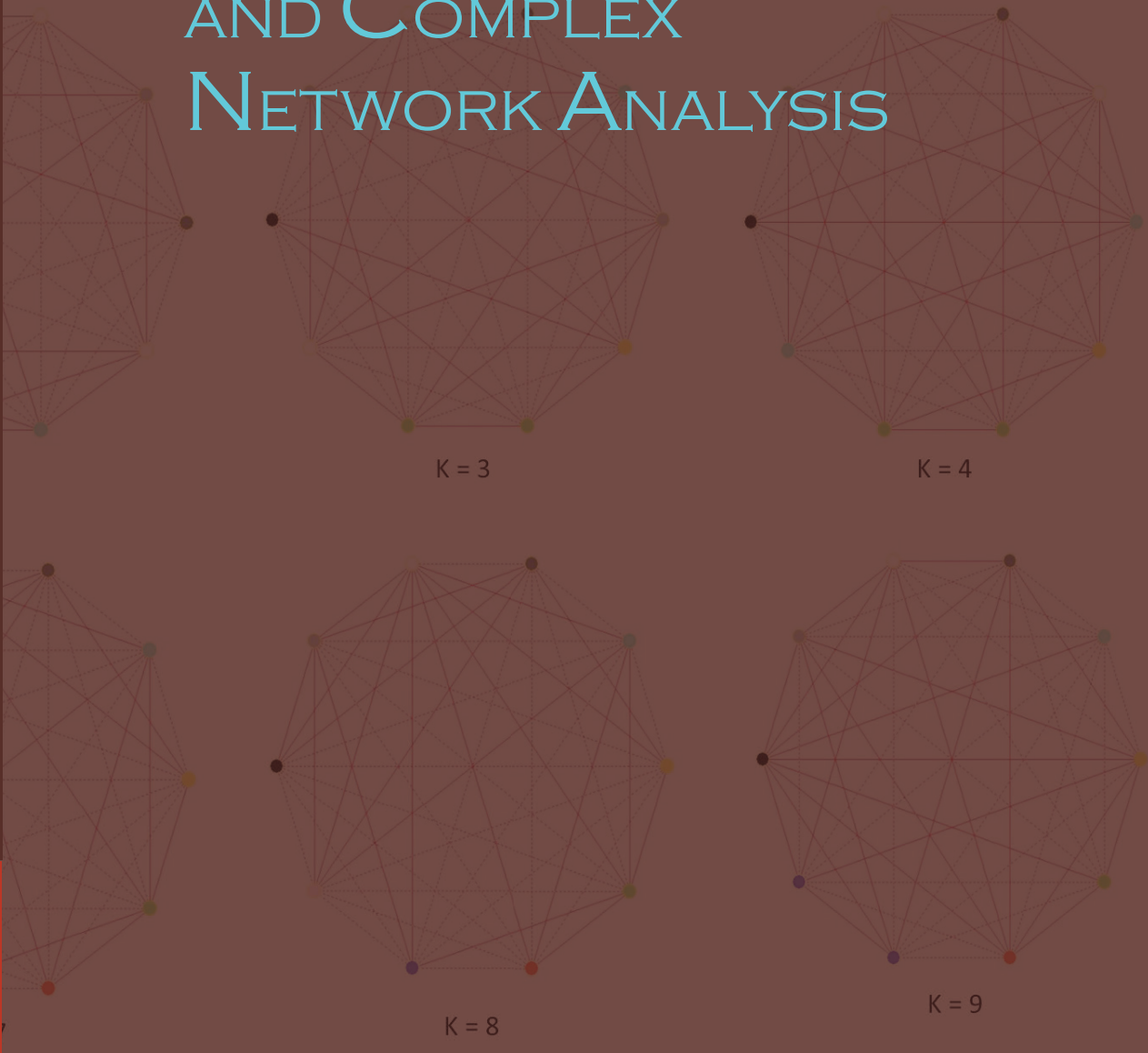
Many objective optimization and complex network analysis

Maulana, A.

Many Objective Optimization and Complex Network Analysis



Asep Maulana

Many Objective Optimization and Complex Network Analysis

Asep Maulana

Many Objective Optimization and Complex Network Analysis

Proefschrift

ter verkrijging van

de graad van Doctor aan de Universiteit Leiden,

op gezag van Rector Magnificus prof. mr. C.J.J.M. Stolker,

volgens besluit van het College voor Promoties

te verdedigen op Woensdag 5 December 2018

klokke 12.30 uur

door

Asep Maulana

geboren te Bandung, Indonesia, in 1974

Promotiecommissie

Promotoren	Prof. dr. T.H.W. Bäck	
	Dr. M.T.M.Emmerich	
Overige leden	Prof. dr. Aske Plaat	
	Dr. Frank Takes	
	Prof. dr. Fons Verbeek	
	Dr. Diego Garlaschelli	LION, Leiden University
	Prof. dr. Budi Ruchjana	Padjadjaran Univerity
	Prof. dr. Jing Liu	Xidian University

ISBN: 978-94-6375-227-5

Het onderzoek beschreven in dit proefschrift is uitgevoerd aan het Leiden Institute of Advanced Computer Science (LIACS), Universiteit Leiden.

This research is financially supported by the Indonesian Endowment Fund for Education (LPDP)

Printed by: Ridderprint BV | www.ridderprint.nl.

Contents

1

Introduction

1.1 • Background

This thesis seeks to combine two different research topics; Multi-Objective Optimization and Complex Network Analysis.

Multi-Objective Optimization aims at finding a set of optimal, non-dominated solutions for optimization problems with multiple (actually, many) conflicting objectives. There is a wide range of applications of multi-objective optimization such as in science, engineering design, network analysis, chemical processes, delivery of products, economics and logistics, medical health and so forth. Since one often faces problems with a larger number of objective functions to be optimized simultaneously, the research topic has shifted to Many-Objective Optimization, which means optimization with (far) more than three objective functions.

Complex network analysis is a research field that deals with analyzing large networks. In this research line, there are some active research topics, such as controlling complex networks, finding communities in a network, and measuring the importance of nodes in networks. Due to the bigger amount of data and more difficult problems arising in complex network analysis, research in this field has increased significantly. To this end, more complex networks has given the challenge of finding better approaches in dealing with the problem to yield some adequate result of an analysis. One active research relating to this problem is known as multiplex network analysis. This is the study of networks that feature different layers of edges for the same set of nodes.

Furthermore, the research in this thesis try to combine many-objective optimization

and complex network analysis. The idea is to attain a benefit for many-objective optimization by applying complex network analysis techniques and the other way around,i.e, to apply many-objective optimization for complex network analysis. Finally, both approaches are combined with an additional contribution to data mining, knowledge discovery and decision analysis.

1.2 · Research Goal and Contribution of this Thesis

The main goal of this research is to study how the fields of complex network analysis and many-objective optimization can benefit from each other.

The first part of this thesis is concerned with using a complex network analysis technique (community detection) for the purpose of many-objective optimization, specifically for visualizing and reducing complexity of many-objective optimization problems. The example of a facility location problem in a city is used in order to demonstrate the applicability and scalability of the approach. Moreover, we study many-objective optimization problems in genetic engineering and complex system design, when multiple traits should be changed to a desired value at the same time. In this study, depending on the structure and intensity of interaction among genes, a complexity transition from simple problems to very difficult problems can be observed. The NK-landscapes model is used as an abstraction of a system of interacting genes (or agents) and by controlling the number of epistatic genes and the radius of interaction the complexity of the problem can be controlled.

The second part deals with applying many-objective optimization for complex network analysis. This part includes the study of modularity- and centrality maximization for multiplex networks, and the combinatorial optimization problem of selecting subsets of nodes to be controlled/immunized when the goal is to prevent the spread of an epidemic throughout the network.
The data used in the experiments is real-world data related to international trade economic, European flight networks, US flight networks, datasets from the UCI network repository, and artificial networks made to resemble real data. In the study with multiplex networks, the network consist of many layers and each layer gives rise to an objective function. Analyzing them in one unity will give an integral depiction on how the network layers correspond with each other. Moreover, this method will help to detect commonalities between clusters and node centrality in the network

layers and thereby help to group them in a meaningful manner. For this purpose, since the network consists of many layers, applying many-objective optimizations is the proper approach. Optimization for all layers of the network will be done simultaneously.

1.3 · Thesis Outline

This thesis consists of eight chapters. The main research result are discussed in Chapter 3,4,5,6 and 7 and divided into two main parts. Part I is composed of Chapter 3 and 4 and part II includes Chapter 5, 6 and 7. Their content is corresponds to the scientific publications that related from the research conducted by the author of this thesis. In the following a brief description of the outline is provided:

- Chapter 2 presents all theoretical preliminaries and definitions for the topics that will be discussed in the subsequent chapters. It includes the definition of multi- and many-objective optimization, network clustering and community detection, network centrality, multiplex networks, and correlation analysis.

- Chapter 3 discusses the use of complex network analysis techniques for reducing complexity in many objective optimizations. This approach is achieved by decomposing many objective functions into a set of independent lower dimensional subproblems, or by aggregating some objective functions into a single objective function. This work introduces a technique from social network analysis for decomposition and aggregation of a system of objective functions. The key idea is to interpret an objective function as a node (agent) in a social network, and a link between nodes to indicate relationships: Negatively weighted links stand for conflicting objectives, zero weighted links for independent objectives, and positively weighted links for objectives that support each other. Using well-known algorithms for community detection it can be shown that, given certain preconditions, it is possible to decompose a many-objective optimization problem into a set of lower dimensional multi-objective optimization problems. This makes it easier to solve the problem and interpret the resulting trade-off (hyper-)surfaces.
  Publication:
  Asep Maulana, Zhongzhou Jiang, Jing Liu, Thomas Bäck, and Michael TM Emmerich. "Reducing complexity in many objective optimization using community

detection." In Evolutionary Computation (CEC), 2015 IEEE Congress on, pp. 3140-3147. IEEE, 2015.

- Chapter 4 presents the empirical study of complexity transitions in interactive networks using community detection techniques. In this research, we investigate NK-landscape models. They are models of the interaction of genes or agents in a network resulting in levels of expression of different phenotypic traits that further cumulate to the overall fitness of the network. We study the phenotypic trait expression levels from the perspective of communities and community detection. The communities are based on the correlation between the phenotypic traits. A single trait illustrates an individual agent which strives to maximize its contributed value to the net value of a community. If high values of one trait occur simultaneously with high values of other traits it regards the traits as high correlated or as supporting each other, and if the value of two traits is uncorrelated, it views their relationship as being neutral, otherwise as conflicting. The work studies what happens to the system of traits when the NK-landscape undergoes a critical transition to a more complex model via the increment of the number of interacting genes and the change of the radius of gene interaction.
  Publication:
  Asep Maulana, André H. Deutz, Erik Schultes, and Michael TM Emmerich. "Community Detection in NK Landscapes-An Empirical Study of Complexity Transitions in Interactive Networks." In EVOLVE-A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation VI, pp. 163-176. Springer, Cham, 2018.

- Chapter 5 shows how to apply many-objective optimization for the analysis of multiplex networks. The work applies different ways for analyzing the community structure in multilayer networks, all relying upon data from many-objective optimization. The first study is a proof of concept that seeks to understand the meaning of the Pareto fronts between modularities of different layers by exact computations of Pareto fronts on three illustrative examples (highly correlated, uncorrelated, and conflicting), which represent important boundary cases. The second step is a study and experiment using trade networks for commodities, by generating data using many-objective optimization, bi-objective optimization (of any pair of layers), and single objective optimization (of any single layer). The

results are analyzed using three tools suggested here: Correlation heatmap, the community of objectives analysis, and the Pareto-front plot matrix. The result shows clearly that a grouping emerges in terms of complementarity and/or in terms of neutrality. As a result, a novel, powerful analysis method for clustering in multilayer networks is proposed. In order to tackle the combinatorial problem, the study and experiment applies state-of-the-art multi-objective optimization algorithms, i.e, the Non-dominated Sorting Genetic Algorithm II (NSGA-II), S-Metric Selection Evolutionary Multi-objective Optimization Algorithm (SMS-EMOA), and a single-objective genetic algorithm.
Publication:

Asep Maulana, Gemmetto, V., Garlaschelli, D., Yevesyeva, I., and Emmerich, M. (2016, December). Modularities maximization in multiplex network analysis using many-objective optimization. In Computational Intelligence (SSCI), 2016 IEEE Symposium Series on (pp. 1-8). IEEE.

Asep Maulana and Michael Emmerich. "Multi-Objective Optimization Approach for Multi-Layers Network Analysis". International conference in Multiple Criteria Decision Making (MCDM) 2017. Page 136, MCDM Society.

Asep Maulana, Gemmetto, V., Garlaschelli, D., Yevesyeva, I., and Emmerich, M. (2016, October). Computing Pareto Fronts of Modularities in Multiplex Economic Network Analysis In International Conference on ENTERprise Information System(CENTERIS), Porto, Portugal, October 2016.

- Chapter 6 presents another many-objective optimization for multiplex networks. Different to the previous chapter, the goal here is to apply it for centrality maximization. In particular, the focus is on the important case of eigenvector centrality. It starts by discussing eigenvector centrality in multiplex networks for the examples of Erdős Rényi random graphs and economic trade networks. Secondly, the non-dominated solutions of the entire network can be computed, the dominance rank for all solutions. In the example of the trade multiplex network, the dominance rank is a rough indicator of how important a node is in the global trade network across different commodities.
  Publication :

Asep Maulana, Michael T. M. Emmerich. "Towards many-objective optimization of eigenvector centrality in multiplex networks." Proc. of IEEE International Conference on Control Decision and Information Technology (CoDIT) April 2017: 729-734, IEEE.

- Chapter 7 presents studies of network immunization: The immunization of complex networks can be formulated as a subset selection problem, where the goal is to select a limited number of nodes to be immunized in order to effectively prevent or decelerate the spread of an epidemic. The drop of the largest eigenvalue (also referred to as 'eigen-drop') is a measure of the impact of an immunization strategy, because it is inversely related to the increase of the critical threshold. The critical threshold decides whether a virus resides in the network or evaporates. It was recently shown that the problem of selecting k out of n nodes from a network such that the eigenvalue drop is maximum belongs to the class of NP-hard problems. Heuristic algorithms have been suggested to solve these problems approximately, most importantly the Netshield algorithm, a greedy approach that approximates the eigenvalue drop by means of a submodular function, the *shield value*, and then maximizes the shield value by means of a greedy approximation algorithm. In this chapter, the topic is to develop and test a problem specific genetic algorithm and compare it to Netshield Plus – an improved variant of Netshield – and show that on six moderate size problems from literature their performance is competitive, and often better.
  Publication:
  Asep Maulana, Marios Kefalas and Michael T.M. Emmerich, "Immunization of Networks Using Genetic Algorithms and Multiobjective Metaheuristics.", Proceedings of IEEE Symposium Series of Computational Intelligence (SSCI) November 27, to December 1, 2017, Honolulu USA, Page 1-8, IEEE.

- Chapter 8 summarizes results of the thesis and provides ideas and promising directions and suggestion for the future work. During the time of the thesis the author also contributed algorithms and analysis techniques which are closely related to the main topic of the thesis, such as group decision making and multi-label classification. It is discussed, how these topics can be further explored and related to the thesis in the future work. These ideas and related works are partially published in the following research articles:

Zhao, J., Fernandes, V. B., Jiao, L., Yevseyeva, I., Maulana, A., Li, R., Emmerich, M. T. Multiobjective optimization of classifiers by means of 3D convex-hull-based evolutionary algorithms. Information Sciences, 367, 80-104, 2016.

Emmerich, Michael, André Deutz, Longmei Li, Asep Maulana, and Iryna Yevseyeva. "Maximizing Consensus in Portfolio Selection in Multicriteria Group Decision Making." Procedia Computer Science 100, 848-855, 2016.

2

Preliminaries

This chapter introduces all fundamental concepts used in this research study.

2.1 · Multi-Objective and Many-Objective Optimization

Multi-objective optimization (also called multi-criteria optimization, multi-objective programming, multi-attribute optimization, vector optimization or Pareto optimization) is an area of decision making that is concerned with mathematical optimization problems involving more than one objective function to be minimized or maximized simultaneously. A multi-objective optimization (MOO) problem is defined by a number of objective functions $f_i: X \to \mathbb{R}$, $i = 1, ..., m$ to be maximized (or maximized) for some search space X. A solution $a \in X$ is said to dominate a solution $b \in X$, if and only if $\forall i$: $f_i(a) \leq f_i(b)$ and $\exists j : f_j(a) < f_j(b)$. Two solutions in $a \in X$ and $b \in X$ are non-dominated w.r.t. each other, if neither a dominates b nor b dominates a. The efficient set X_e is the set of solutions in X that is not dominated by any solution in X. The Pareto front PF is the image set of X_e, i.e. $PF = \{f(x) \mid x \in X_e\} \subseteq \mathbb{R}^m$.

Many-objective optimization (ManOO) applies to problems with more than three objective functions to optimize simultaneously [23]. There are new issues arising in many-objective optimization compared to multi-objective optimization. The two main issues in differentiating many-objective problems from multi-objective problems are the following: on the one hand, a large number of objective functions makes visualization of the Pareto front impractical; on the other hand, analysis of Pareto fronts is difficult due to the tendency that a majority of solutions will be non-dominated. Hence, the tradeoff analysis of conflicts between objective functions and the representation of the entire Pareto front can become difficult and in-transparent. Moreover, a high number

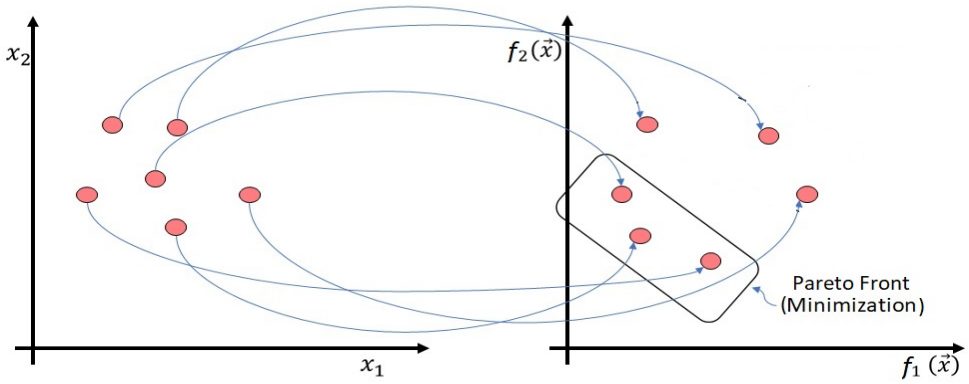


Figure 2.1 An illustration of multi-objective optimization for the decision variables and two objectives (to be minimized).

of objectives can cause to a significant increase in the computational time complexity required to compute Pareto fronts. In spite of these difficulties, many technique and approaches have been proposed to deal with many-objective optimization [33], [50], [40], [57].

2.2 · Networks

Network (or graphs) are one of the most fundamental data structure in computer science. A network can be represented by means of an adjacency matrix $A \in \mathbb{R}^{m \times m}$, where m denotes the number of nodes. Here the entry A_{ij} is zero if there is no connection between node i and node j, and non-zero otherwise. The non-zero number represents the weight of the connection. A network is used to represent the relationship between objects in a certain domain. An object in a network is a named node or vertex and the relationship between objects is called edge or link. The network can be used to describe a relationship between humans and their relationship in social life, countries in the world trading commodities, cities in a delivery problem in logistics, train stations or bus stops in some transportation system, connected computers in the Internet, airports in flight data set, interactions between proteins in biological system, and so forth. Analyzing such types of networks has become an immensely promising research area,and there is a lot of active research in network science, including community detection and network centrality.

2.2.1 · Community Detection

Community detection is a well-known method of social network analysis. A community is a group of nodes with many links between nodes of the group, but not so many links to nodes outside the group. The most popular community detection method for network clustering is the Louvain method, which is based on modularity maximization. Modularity is a concept that originates from social network analysis [42]. It is a quality measure or strength of partitioning of a graph into communities (partitions, groups or clusters). Maximizing modularity groups the nodes of a graph in such a way that intracluster graph distances (or edge weights) are minimized and inter-cluster graph distances (or edge weights) are maximized. Let A_{ij} denote the weight of the edge from node i to node j. Let m denote the number of the nodes, and $K_i{=}\sum A_{ij}$ denote the sum of weights of edges belonging to node i. Moreover, C_i is the community to which node i is assigned. Finally $\delta(.,.)$ is the Kronecker symbol, which is equal to 1, if and only if both arguments are equal to each other. Otherwise, it obtains the value of 0. Now the modularity is defined formally as:

$$Q_{signed} = \frac{1}{2m} \sum_{i,j \in \{1,\ldots,m\}} \left[A_{ij} - \left(\frac{K_i * K_j}{2m} \right) \right] \delta(C_i, C_j) \tag{2.1}$$

Modularity maximization is an NP-Hard problem. This can be shown by polynomial reduction of 3-PARTITION [19], and thus, in general, it is difficult to solve this problem by means of exact methods. There are, however, several fast heuristics available, such as the Louvain method [42], which is a greedy heuristic that finds high modularity partitions of a network in short time. The first phase of the Louvain method begins by placing each node in its own singleton 'community'. Then the looping over all nodes is done in the following way:

For each node i all neighbors, that is, all the nodes j such that A_{ij} is non-zero, are analyzed from the point of view of the gain computed after removing i from its community and placing it into the community of j. The node i is then put into the community for which the increase in modularity is largest. If none of the potential re-assignments of i into other communities is associated with a positive gain in modularity, i stays in its original community and the algorithm moves on to the next node. The loop

is repeated until no further improvements are obtained, i.e. when the modularity has reached a local optimum.
In the next phase of the algorithm, a new network is constructed with the communities of nodes obtained at the first phase of the Louvain method. The weights of the edges between the new nodes are given by the sum of the weights between all nodes between communities of the previous phase. When this phase is finished, a new phase is started, and so on. This creates a hierarchy of communities. The algorithm stops when a maximum of the modularity is obtained, or in practice when the last performed pass did not further increase modularity.

2.2.2 · Network Centrality

Network centrality is an important concept in network studies and analysis. As in everyday reality, a person, or organization in some way has the influence to generate some important decision for a community or even for a human being. Identifying an important person or organization can be recognized as the problem of identifying key players in a community. Many network centrality methods have been proposed to identify different key players in a social setting ([11], [12]) such as the following:

- Degree centrality, which focuses on the number of peers to which a node is connected [21].
- Betweenness centrality, which considers the number of shortest paths in the network that pass through a certain node [6].
- Closeness centrality, which measures distance from a certain node to all other nodes [43].
- Eigenvector centrality and PageRank, which consider the number of links from one node to other nodes, the importance of these nodes, and to how many these nodes themselves point to [10], [44], [49].

In this thesis, the emphasis will be more on eigenvector centrality. The reason behind this decision is that the method represents the most fundamental properties of centrality measures, and is a remarkably long studied method [52]. It is also very similar to the well known Google PageRank Methods.

Formally, the eigenvector centrality can be defined as follows: For a given graph G=(V, E) with $|V|$ being the number of nodes, let $A = (A_{v,t})$, $v \in \{1, ..., |V|\}$, $t \in \{1, ..., |m|\}$ be the adjacency matrix, i.e. $A_{v,t} = 1$ if node v is linked to node t, and $A_{v,t} = 0$ otherwise. The relative centrality score of node v is defined as

$$x_v = \frac{1}{\lambda} \sum_{t \in M(v)} x_t = \frac{1}{\lambda} \sum_{t=1}^{|m|} A_{t,v} x_t$$

where $M(v)$ is a set of the neighbors of v and λ is an eigenvalue of A. In vector notation, the eigenvector centrality can be rewritten in a simple equation as $Ax = \lambda x$ and it becomes clear that x is an eigenvector of A and λ an eigenvalue. As there can be many eigenvectors of A, by convention, the eigenvector that corresponds to the biggest eigenvalue is considered. It consists of only positive components. There are two important factors that influence the eigenvector centrality of the node in the network. They are:

- The number of or total weight of links neighbors that point to the node.
- The centrality of neighbors that point to the node.

There is a possibility that nodes with more neighbors have a lower eigenvector centrality compared to nodes with fewer neighbors. This can happen because the neighbors of the less connected node have a higher centrality.

2.2.3 · Multiplex Networks

Multiplex networks are networks consisting of multiple edge sets for the same set of node. The network is made up of multiple layers, each of which represents a given operation mode. More clearly, it can be defined as graphs that consist of a number of, say n, nodes and m different edge sets for these nodes, called layers. The node set is denoted by V and the edge sets are denoted by E_l, $l \in \{1, ..., m\}$. A multiplex network is represented formally as $G = (G_1, G_2, \ldots, G_l, \ldots, G_m)$.

A visual illustration of the layers is shown in Figure 10. Here we assume that every network G_l is fully described by the adjacency matrix A_l with elements $A^l_{ij} = W^l_{ij} > 0$, where $A^l_{ij} = W^l_{ij} > 0$, if there is a link with a positive weight between nodes i and j in layer l, and $A^l_{ij} = 0$ otherwise.

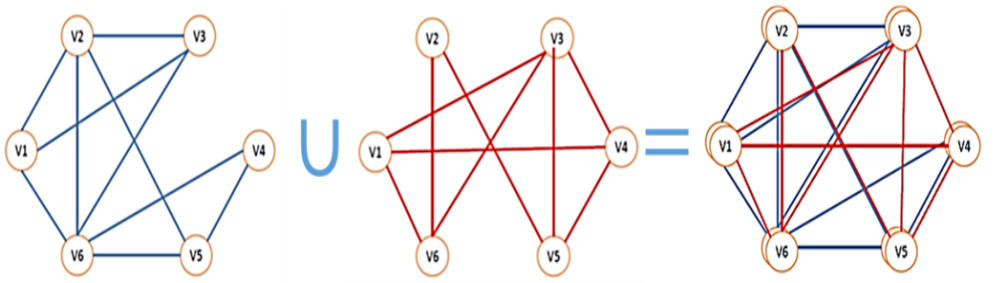


Figure 2.2 A visual illustration of multiplex network consisting of two layers of networks distinguished by blue and red colour. Each layer has different links but the nodes remain the same. The union of these layers, indicated by ∪, forms the multiplex network.

2.3 · Matrix Correlation Analysis

In statistics, the Pearson correlation coefficient, referred to as the Pearson's ρ, Pearson product-moment correlation coefficient (PPMCC) or bivariate correlation, [45] is a way to measure the linear correlation between two object. It has a value between -1 and 1, where 1 is perfect positive linear correlation, -1 is perfect negative linear correlation, and 0 is defined as no correlation among those two objects. This method was developed by Karl Pearson from a related idea introduced by Francis Galton in the 1880s [20] [46] [53] and is widely used in the sciences, providing meaningful comparisons in system analysis.

We will now give the precise definition of how to compute the empirical correlation coefficient of two functions based on a finite number of evaluations. Let $(\Omega, \mathcal{S}, P)$ denote a probability space, where $\mathcal{S}$ is the event space and Ω denotes the set of elementary outputs – here chosen as the input space $X = \{0,1\}^N$. We will only consider singletons as events and write ω instead of $\{\omega\}$. In the following we consider the entire input space $X = \{0,1\}^N$, and a uniform distribution over this set. For each function F_i the random variables $\mathcal{F}_i : \Omega \to \mathbb{R}$ are defined as $\mathcal{F}_i : \omega \mapsto F_i(\omega)$. Next, consider a sample $\Omega' \subseteq \Omega$ and the realizations of random variables $\mathcal{F}_1(\omega), \ldots, \mathcal{F}_m(\omega)$ for $\omega \in \Omega$. Now, for the group of paired evaluations of $\mathcal{F}_i$ and $\mathcal{F}_j$ the empirical correlation coefficient can be computed as:

$$\rho_{ij}^{e} = \frac{\frac{1}{1-|\Omega_s|}\sum\limits_{\omega\in\Omega_s}(\mathcal{F}_i(\omega)-\overline{\mathcal{F}_i})(\mathcal{F}_j(\omega)-\overline{\mathcal{F}_j})}{\sqrt{\frac{1}{1-|\Omega_s|}\sum\limits_{\omega\in\Omega_s}(\mathcal{F}_i(\omega)-\overline{\mathcal{F}_i})^2}\sqrt{\frac{1}{1-|\Omega_s|}\sum\limits_{\omega\in\Omega_s}(\mathcal{F}_j(\omega)-\overline{\mathcal{F}_j})^2}}$$

As the matrix correlation applies to this research, it will utilize in the context of multi-objective and many-objective optimization, and decision making. The correlation can be interpreted as follows:

- Positively correlated objective functions can be interpreted as objective functions that support each other.
- Uncorrelated objective functions are considered to be independent of each other.
- Negatively correlated objective functions are in a strong conflict with each other.

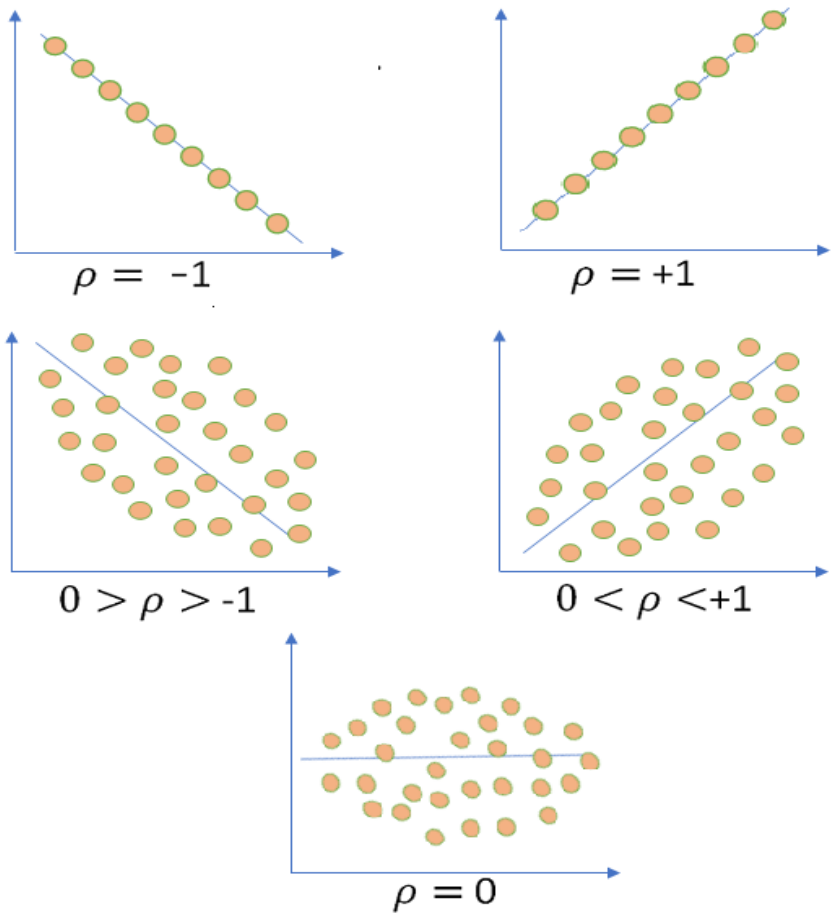


Figure 2.3 A visualization of the Pearson Correlation Coefficient and its range of values.

PART I

3

Community Detection for Reducing Complexity of Many Objective Optimization

3.1 · Introduction

In the last decade, the modern science of networks was probably the most active field within the interdisciplinary research field of complex systems. Many complex systems can be represented as networks, where the elementary parts of a system and their mutual interactions are nodes and links, respectively. Study and analysis of networks is a challenging research topic.

Social network analysis as a subfield of network science that is concerned with the classification and clustering of networks. As such it is powerful in defining the community and correlation among nodes (representing individuals) in the context of social networks. However, community detection in large networks needs a special technique that can help to optimize results. In recent years multi-objective optimization for community detection has been developed by many researchers. Here, multi-objective optimization seeks to find an optimal clustering with respect to criteria such as *modularity* and *parsimony*.

In our research, however, we turn the question around and seek to apply social network analysis, and in particular community analysis, to analyze multi-objective optimization problems with a large number of objectives, as they are, for instance, given in problems of facility location problems, multi-class classification problems, nurse scheduling, or multidisciplinary design.

This chapter starts with a definition of the problem (Section 3.2) and a brief summary of related work (Section 3.3). The approach of mathematical statistics has been described in Chapter 2. Furthermore, a workflow of the decomposition and the applied algorithms will be discussed in Section 3.4. Empirical proof of concept studies illustrate and validate the methodology in Section 3.5. Finally, Section 3.6 closes the chapter with a summary of the main results and a discussion of current limitations of the method.

3.2 · Problem definition

Recall from Chapter 2, that a multi-objective optimization problem (MOP) is defined by a number of objective functions $f_i : X \to \mathbb{R}, i = 1, \ldots, m$ to be minimized (or maximized) for some search space X. Without loss of generality we consider only maximization as a goal. Moreover, we will not consider constrained problems in the first place, that is we assume that X consists only of feasible solutions.

A prevalent problem in multi-objective optimization is that for a large number of objective functions it is not possible anymore to visualize Pareto fronts in a Cartesian coordinate system. The trade-off analysis of conflicts between objective functions can become very intransparent. Moreover, a high number of objectives can yield to a significant increase in the computational time complexity required to compute Pareto fronts.

The aim of our research is to *reduce complexity in many-objective optimization by aggregating complementary objective functions or decomposing the problem into independent subproblems*.

A novel approach to achieving this is studied. This approach seeks to apply *community detection in networks* in order to form groups of objectives that support each other, and to separate groups of objectives that are independent of each other. This will form clusters of objective functions: $\{C_1, \ldots, C_k\}$ with $k \leq m$ forming a partitioning of the set $\{f_1, \ldots, f_m\}$. Moreover, we will establish dependence and independence relations between these clusters, based on whether or not two clusters are conflicting or neutral with respect to each other.

The analysis will be based on an interpretation of objective functions and their correlation with random variables. Positive correlation will be interpreted as a positive link, negative correlation as a negative link, and, finally, zero correlation as no link in the network. In the following, we will detail the components of the approach and

discuss its scope (types of MOPs where it is suitable) and limitations. Moreover, we will provide a benchmark study in order to prove the concept empirically.

3.3 · Related Work

Many-objective optimization was introduced as a class of problems with more than four objective functions [48]. It has been observed that classical evolutionary multi-objective optimization methods (EMOA), such as NSGA-II [17] and SPEA2 [65], typically work well only for a few objective functions (ca. 2 – 4). Firstly, the results of optimization are more difficult to interpret than for problems with, say, only two or three objectives. Moreover, hypervolume-based methods such as SMS-EMOA [8] that perform very well for small numbers of objective functions become computationally infeasible for a higher number of objective functions [55].

This is the reason why algorithms have been suggested that are especially well suited for many-objective optimizations. Examples of algorithms that perform well in many-objective optimization are HyPE [5] and MOEA/D [64]. However, they do not make use of special properties of the problem such as a correlation between objective functions. Here, it shall be noted that in the general case the Pareto front of a m dimensional problem can be a $m - 1$ dimensional manifold, which in case of a large number of objectives leads to a reduction of the space of alternatives that can be considered as marginal. Moreover, it is very difficult to get any intuition about trade-offs from the data on such high dimensional spaces.

The situation changes if some of the objective functions are correlated or anti-correlated with each other. On the one hand, some of the aforementioned algorithms perform better in such cases [26] and on the other hand, if correlations are suspected a-priori, they can be exploited actively by the algorithm to reduce the problem difficulty. For instance, in [51] it is suggested to use Principal Component Analysis (PCA) for finding a reduced set of objectives. The research presented in this thesis goes in a similar direction, that is it seeks to exploit the correlation between objective functions. In [28] the idea that correlated objective functions cause redundancy was exploited to reduce the number of objective functions during a run.

However, the proposed method will differ from existing approaches in three important aspects:

- We will take a new perspective on how to view the problem of aggregation and decomposition by *viewing objective functions as entities of a community*, and use

community detection algorithms. These methods not only can help to simplify the problem of finding the Pareto front but also provide interesting graph based visualizations of the problem's inherent structure to the decision maker.

- As in [28, 51], we will use correlation, but treat correlation, anti-correlation, and neutrality differently, thereby exploiting certain features that the social community based perspective offers. Informally speaking, objective functions can be friendly, neutral or enemies with each other.

- We will propose a workflow-approach that provides feedback to users in different stages of the optimization process. The method can be successful (in case sufficient structure and modularity are found) or unsuccessful, in which case one has to resort to conventional methods. In case of success, it can provide the user with feedback on the essential structure of the Pareto front by means of low dimensional trade-off surfaces and also generate sets that cover the essential parts of it.

Related to coefficient correlation as explained in Chapter 2, in the context of decision making, we can interpret this correlation as follows:

- Positively correlated objective functions can be interpreted as objective functions that support each other. This means that optimizing one function will imply that also the other function will obtain good values.

- Uncorrelated objective functions are considered to be independent of each other. They can be optimized in isolation from each other. The MOP can be decomposed in MOPs that consist of partitions with less objective functions.

- Negatively correlated objective functions are in a strong conflict with each other. For conflicting objective functions trade-off analysis needs to be conducted, which in turn can be prepared by computing the Pareto front of the problem.

Here it is proposed to compute the correlation between all pairs of objective functions. The information is then interpreted in a relational context and we apply community detection in order to identify communities of objective functions. Communities can be opposed to each other (negative link), or neutral with respect to each other (no link). The fundamental idea of this chapter is how to use techniques from social network analysis in order to decompose and aggregate.

Once the correlation between the objective functions is found, one can make a grouping of the objective functions based on the correlation status of those objective functions either being correlated or conflicting. Moreover, each group that consists of both correlated and conflicting objective functions will pull/attract each other. This information can serve the needs of solving many objective optimization problems, by grouping objective functions and decomposing the problem into subproblems.

3.4 · Workflow and Algorithms

The workflow and algorithmic methods used in the community-detection for many-objective optimization (CoDeMO) approach will be used. As opposed to the term 'algorithm', by a 'workflow' we mean a step-wise, linear process in which the user interacts or inspects data or visualizations at several stages, and possibly is asked for feedback.

The CoDeMo workflow proceeds with the following steps:

1. **Problem sampling**: Generate n samples of decision variable vectors that are evenly distributed in the range of the decision variables.

2. **Evaluation**: Evaluate the m objective functions for each sample, which yields an $n \times m$ matrix.

3. **Correlation Analysis**: Compute the correlation coefficient matrix from the table of objective function values, which yields a $m \times m$ matrix of correlation coefficients in $[-1, 1]$.

4. **Construct Network**: Take each objective function as a node, and the correlation coefficient between two objective functions as a link, which means the correlation coefficient matrix will be taken as the matrix of the constructed signed network.

5. **Detect Communities**: Use a graph-theoretic algorithm to detect communities using the information of the correlation matrix and interpreting them as edge weights (values near zero indicate non-existence of edge).

6. **Decompose**: Based on the result of the community detection, decompose the problem into $k \leq m$ subproblems by partitioning the set $\{f_1, \ldots, f_m\}$ into communities of mutually non-independent objectives.

7. **Aggregate**: Within each community: Aggregate cliques of communities that support each other to a single objective function by summation, resulting for each subproblem i with, say, k_i objective functions, in a similar subproblem with $\ell < i$ pseudo-objective functions.

8. **Solve Subproblems**: Find an approximation and Pareto front for each of the subproblems with aggregated objectives.

9. **Merge**: Merge all results and output them in a Parallel coordinate diagram, first ordered by subproblem and within subproblems by aggregates.

The Algorithm of community detection used in step 5 (as explained in chapter 2), but with more specific measured weight, is called BGLL [29]. The algorithm is designed to use a greedy strategy to improve the modularity on a signed network [24], which is defined as

$$Q_{signed} = \frac{1}{2w} \sum_i \sum_j \left[w_{ij} - \left(\frac{w_i^+ w_j^+}{2w^+} - \frac{w_i^- w_j^-}{2w^-} \right) \right] \delta(C_i, C_j)$$

where w_{ij} is the weight of the adjacency matrix, $w_i^+(w_j^+)$ is the sum of all positive weights of node $i(j)$. $w^+(w^-)$ denotes the sum of absolutes of all positive(negative) weights of the signed network and $w = w^+ + w^-$. $C_i(C_j)$ is the community label of node $i(j)$ and $\delta(C_i, C_j)$ is 1 if $C_i = C_j$; otherwise 0.

BGLL greedily aggregates two nodes into one community if it causes a large modularity increment. The aggregation will repeat many times until the optimum community structure is obtained. The algorithm is implemented in Pajek, were it is called Louvain method [54]. Details can be found in [29] [54].

The introduced nine step workflow involves the user in several parts. First and foremost, the community structure visualization for the objective functions might provide valuable insights into the problem's structure to the user. As we will not find a clear-cut partitioning into cliques, in the current CoDeMO workflow the user needs to perform the merging and aggregation steps. It might be possible, however, to develop robust criteria for doing this automatically.

The visual output of the community detection process is an important result of the workflow, as it supports the decomposition process by providing essential insights into the problem structure. Its graphical elements are:

- Each objective function is a node in a circle.
- The graph visualization places anti-correlated nodes distant to each other and indicates their conflicting nature by blue connectors.
- It places objective functions that are positively correlated close to each other and indicates their supportive relation by red connectors.
- Elements that have close to zero correlation, meaning that they are independent (neutral) of each other (a threshold $1/e$ is used here, where e is the Euler number), and do not have a connecting line segment.

An example of a community detection result for 30 objective functions and subsequent problem decomposition and aggregation is provided in Fig. 3.5. From this graph, the grouping of the objective functions is as follows: Firstly, we identify three independent subproblems. The community detection places the nodes in such a way that the connected components of the graph can be easily identified:
$C_1 = \{1, 2, 3, 4, 6, 7, 8, 5, 9, 10\}$,
$C_2 = \{20, 18, 17, 16, 14, 13, 12, 19, 15, 11\}$,
$C_3 = \{21, 22, 23, 24, 25, 26, 27, 28, 29, 30\}$.
In the aggregation step of the workflow we aggregate the objective functions, which yields three subproblems:
Subproblem 1:
$\Phi_1^{C1} := f_1 + f_2 + f_3 + f_4 + f_6 + f_7 + f_8 \rightarrow \min$,
$\Phi_2^{C1} := f_5 + f_9 + f_{10} \rightarrow \min$.
Subproblem 2:
$\Phi_1^{C2} := f_{11} + f_{15} + f_{19}$,
$\Phi_2^{C2} := f_{20} + f_{18} + f_{17} + f_{16} + f_{14} + f_{13} + f_{12} \rightarrow \min$.
and Subproblem 3:
$\Phi_1^{C3} := f_{24} + f_{25} + f_{30}$,
$\Phi_2^{C3} := f_{12} + f_{13} + f_{14} + f_{16} + f_{17} + f_{19} + f_{20} \rightarrow \min$.
Now, instead of solving a 30-objective problem, three bicriteria optimization problems can be solved instead without losing essential information.

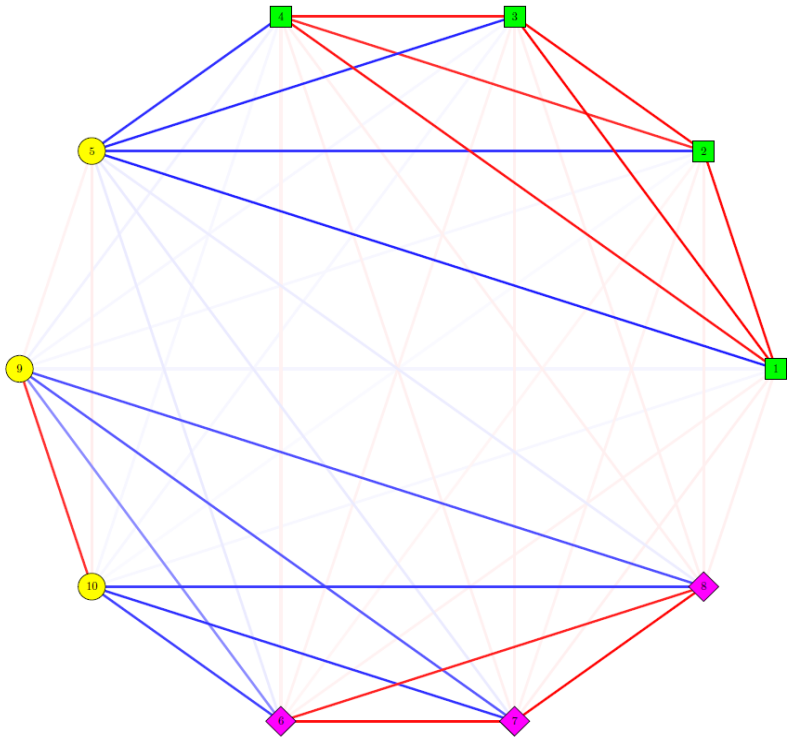


Figure 3.1 Community detection for ten objective function

3.5 · Experimental Analysis

For testing the approach in a proof-of-concept study we test it on a problem for which we already have an idea of the inherent structure of communities, but the algorithm is not provided with this information. Moreover, the problem is scalable and has a practical background, being a special case of a multi-facility location problem.

In the following, we only consider one facility of each type, which provides a benchmark with a known solution in terms of decomposition into subproblems and aggregation of objective functions. The decision vector is then given by $(x_1, x_2, \ldots, x_{2N-1}, x_{2N})$, where (x_1, x_2) is the coordinate of the first facility of type 1, (x_3, x_4) is the coordinate of the second facility of type 2, and so on. We assume the position of the citizens is fixed and each one of the citizens forms a separate objective function. More concretely citizen's needs are of the kind that a citizen with coordinates $(c_1^{(.)}, c_2^{(.)})$ wants to be close to a facility of type 1, a citizen with coordinates $(c_3^{(.)}, c_4^{(.)})$

wants to be close to a facility of type 2, a citizen with coordinates $(c_5^{(\cdot)}, c_6^{(\cdot)})$ want to be close to facility 3, and so on.

Moreover, in order to test the capability of the workflow to detect independent clusters we make the additional assumption that each citizen has only one facility that he or she wants to be located close to.

3.5.1 · Problem with 10 Objectives

As a first example we choose ten nodes as 10 objective functions. The functions have the signatures $f_1 : \mathbb{R}^4 \to \mathbb{R}$, ..., $f_{10} : \mathbb{R}^4 \to \mathbb{R}$. The functions f_1, f_2, ..., f_5 are sensitive to changes of x_1 and x_2 (first facility type) and the functions f_6, ..., f_{10} are sensitive to changes of x_3 and x_4 (second facility type). These functions are distances to centers of facility type 2 ((x_3, x_4)-space). Table 3.1 lists the locations of $c^{(i)}$ (the citizens) for each objective function and a graphical representation of them is found in Fig. 3.2 (for facilities of type 1) and in Fig 3.3 (for facilities of type 2).

Table 3.1 Facility locations and facility types.

10 Objective Functions				
	$c_1^{(i)}$	$c_2^{(i)}$	j_1	j_2
f1	1.1	1.2	1	2
f2	0.9	0.8	1	2
f3	1	1	1	2
f4	0.7	1.1	1	2
f5	0.2	0.1	1	2
f6	0.7	1	3	4
f7	1	1.1	3	4
f8	1	0.9	3	4
f9	0.2	0.4	3	4
f10	0.3	0.2	3	4

In Step 1 of the workflow, we sample the functions randomly within the $[0, 1]-$ range 500 times. Then, in Step 2, evaluate the objective functions for each point, yielding a 500×10 matrix of function values. From this matrix – in Step 3 of the workflow – we determine correlation coefficients.

The full matrix with one correlation coefficient for each pair of objective functions is shown in Table 3.1.

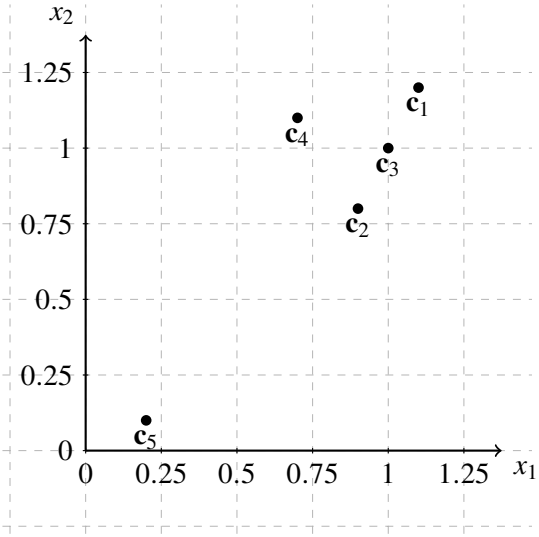


Figure 3.2 The centers in the coordinate space spanned by x_1 and x_2. In a facility location problem, the centers stand for citizens who demand facility of type 1 in their proximity.

Next, in Step 4 of the CoDeMO workflow, we apply the community detection method, which yields Fig. 3.1. Based on the clear structure of the graph, Step 4 (decomposition) and Step 5 are straightforward and we identify two independent subproblems, one involving the set $C_1 = \{1, 2, 3, 4, 5\}$, and the other involving the set $C_2 = \{6, 7, 8, 9\}$. This result, which we obtained from the graphics, is also plausible in the context of the facility location problem because the set of objective functions is partitioned into sets of functions for the same type of facility. The aggregation is done according to the color given to the nodes by the community detection algorithms, yielding Subproblem 1:

$$\Phi_1^{C1}(x) = (f_1 + f_2 + f_3 + f_4)(x) \to \min \tag{3.1}$$

$$\Phi_2^{C1}(x) = f_5(x) \to \min \tag{3.2}$$

$$x \in [0,1]^4, \tag{3.3}$$

and Subproblem 2:

$$\Phi_1^{C2}(x) = (f_6 + f_7 + f_8)(x) \to \min \tag{3.4}$$

$$\Phi_2^{C2}(x) = (f_9 + f_{10})(x) \to \min \tag{3.5}$$

$$x \in [0,1]^4 \tag{3.6}$$

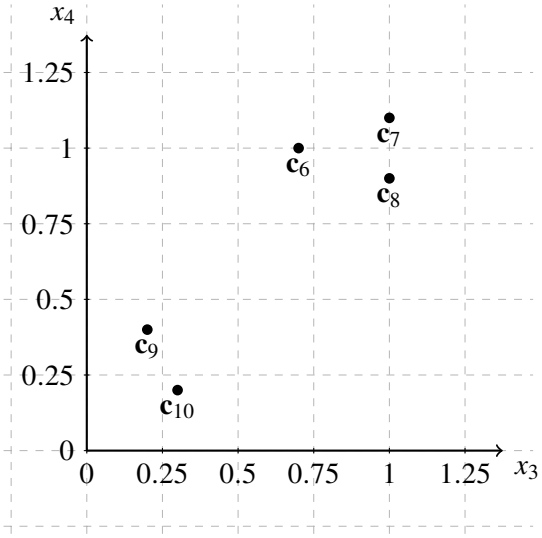


Figure 3.3 The centers in the coordinate space spanned by x_3 and x_4. In a facility location problem, the centers stand for citizens who demand facility of type 2 in their proximity.

Step 7 is to perform optimization on the subproblem and then to visualize results for the objective functions. In Fig. 3.6 the results for Φ_1^{C1} and Φ_2^{C1} using NSGA-II and MOEA/D are depicted. For the optimization, we used JMetal with a population size of 300 and a budget of 15000 evaluations. On the plain many-objective optimization problem MOEA/D achieved better results than NSGA-II (Fig. 3.6, top). For this reason, we show the result of a comparison only between MOEA/D and CoDeMO, using MOEA/D for the lower dimensional subproblem. As expected, the accuracy of the front is much better when the subproblem is optimized directly. This holds for both problems, but for subproblem 1 the difference was most significant. For subproblem 1 (Fig 3.6) the range of Φ_2 is much better captured when CoDeMO is used.

3.5.2 · Problem with 30 and 50 objectives

In a similar manner we created a MOP with 30 objective functions and 50 objective functions. For these problems we will only demonstrate the decomposition process. This time three facility types are featured. Coordinates are given by (x_1, x_2), (x_3, x_4), and, respectively, (x_5, x_6). Table 3.3 shows the centers: By a given correlation matrix, the visualization of the community structure can be seen in Fig. 3.5, and the scatter plot in Fig. 3.7. The three clusters are clearly separated from each other and conflicting

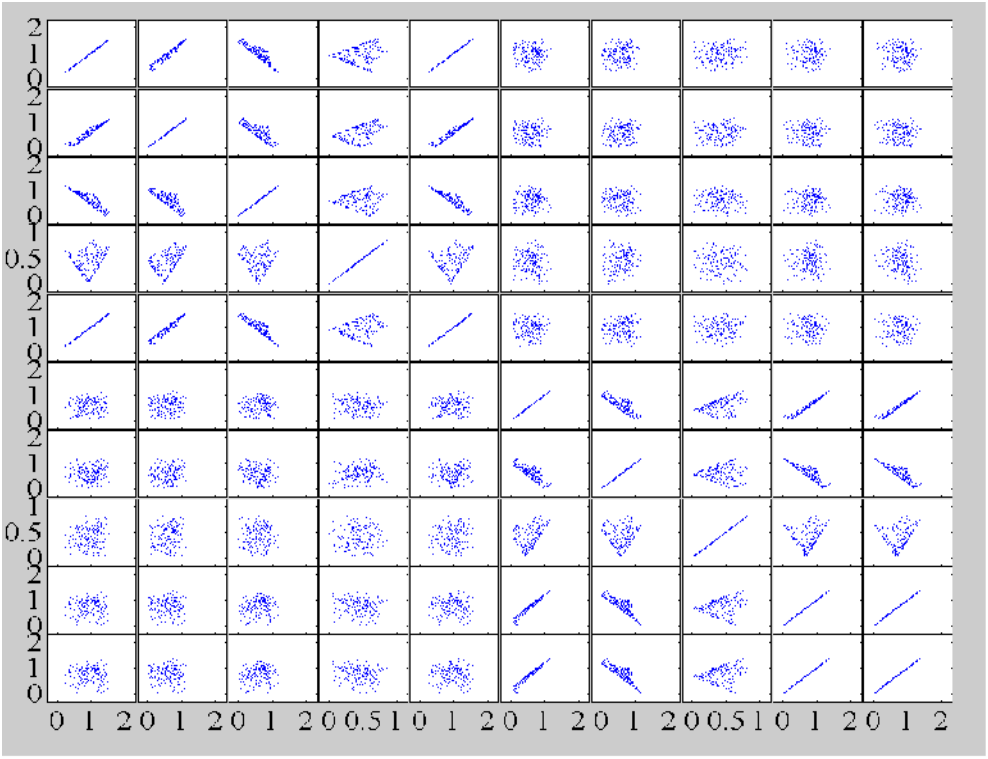


Figure 3.4 Scatter plot matrix for the 10 objective problem.

objectives placed opposite to each other. We can decompose this problem as described in Section 3.4.

The table of the 50 citizen problem exceeds the size of the paper and is made available as support material. We only display the scatter plot matrix of this problem in Fig. 3.9. We used a similar structure than in the 30-dimensional problem but added citizens to the clusters. The resulting community graph is depicted in Fig. 3.8. This scatter plot matrix is almost unreadable and we only show it to contrast it to the clearer picture we obtain from graphs provided by community detection. Again, the community detection algorithm manages to visualize the clusters in a way that allows for simple decomposition and aggregation. The result shows that the community detection also can work for larger size problems if the underlying structure is sufficiently simple.

A note of caution is to be made here: Nodes 37 and 4 in the lower left of the biggest cluster are not easy to place. They are close to two nodes in terms of positive correlation but conflict with all other nodes in the cluster. This 'in-between-ness' makes it difficult to

Table 3.2 Correlation Coeffecients for 10 objective function with 500 inputs.

1.00	0.97	1.00	0.93	-0.88	0.05	0.06	0.06	-0.04	-0.03
0.97	1.00	0.98	0.83	-0.79	0.04	0.05	0.06	-0.03	-0.02
1.00	0.98	1.00	0.90	-0.86	0.05	0.06	0.06	-0.03	-0.03
0.93	0.83	0.90	1.00	-0.82	0.05	0.06	0.06	-0.05	-0.03
-0.88	-0.79	-0.86	-0.82	1.00	-0.08	-0.08	-0.07	0.05	0.07
0.05	0.04	0.05	0.05	-0.08	1.00	0.95	0.88	-0.45	-0.75
0.06	0.05	0.06	0.06	-0.08	0.95	1.00	0.98	-0.65	-0.81
0.06	0.06	0.06	0.06	-0.07	0.88	0.98	1.00	-0.69	-0.76
-0.04	-0.03	-0.03	-0.05	0.05	-0.45	-0.65	-0.69	1.00	0.81
-0.03	-0.02	-0.03	-0.03	0.07	-0.75	-0.81	-0.76	0.81	1.00

decide to which aggregate objective function they belong. In CoDeMO the assignment to clusters is done manually and the user could aggregate them as a new (third) objective function thereby making the visualization and analysis of the subproblem more difficult. Alternatively, (s)he could place them in the lower community – ignoring their positive correlation with two objective functions. The latter approach has the advantage that the subproblem can be solved as a problem with only two objectives. To avoid loss of essential information for decision making, we recommend using the first approach.

3.5.3 • Limitations of the Approach

In cases where much less community structure is available than in our examples, our approach is unfortunately only of limited use. In such cases, it at most provides negative results, that is the insight that the problem's correlation structure is inherently complex. There might be approaches that are able to exploit types of structures based on correlation other than those exploited in CoDeMO, but this falls beyond the scope of this study.

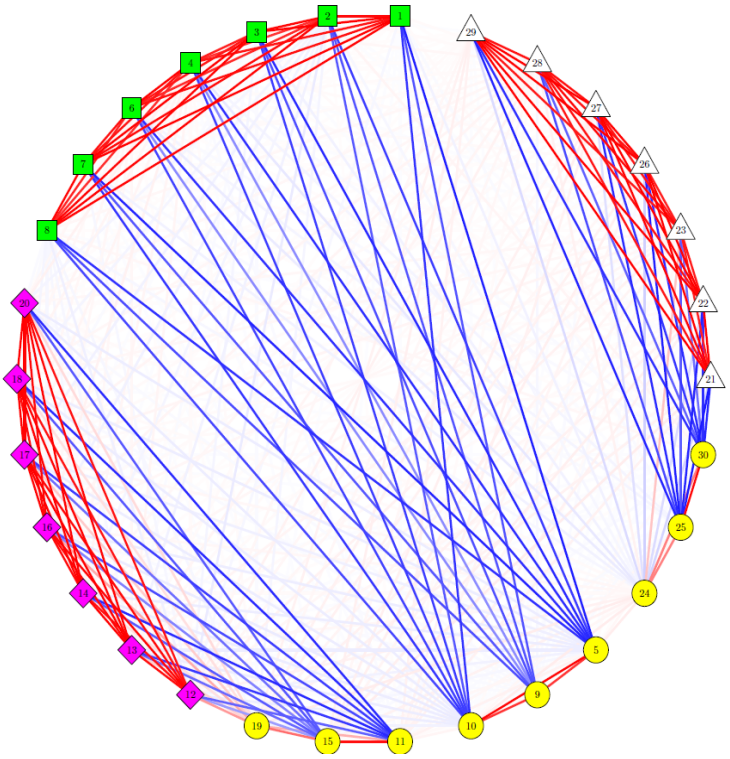


Figure 3.5 Example of a community detection result for 30 objective functions.

An even more subtle case of a difficult problem structure occurs in case of a phenomenon we will refer to as *higher order conflict*: It is possible to construct problems where every pair of objective function is non-conflicting but there is no solution that satisfies all objective functions. For instance the problem $min\{||\mathbf{x},(0,0,1)||,||\mathbf{x}-(0,1,0)||\} \rightarrow \min$, $min\{||\mathbf{x}-(1,0,0)||,||\mathbf{x}-(0,1,0)||\} \rightarrow \min$, and $min\{||\mathbf{x}-(1,0,0)||,||\mathbf{x}-(0,1,0)||\} \rightarrow \min, \mathbf{x} \in [0,1]^3 \subset \mathbb{R}^3$ has this characteristic. An appropriate technique to visualize this information would be to use community detection in hypergraphs.

3.6 · Summary

In this chapter, we applied a community detection technique to reduce the number of the objective functions in many-objective optimization. A workflow called Community Detection for Many-objective Optimization (CoDeMO) was discussed that uses graph-theoretic community detection to reveal the structure of many-objective black-box

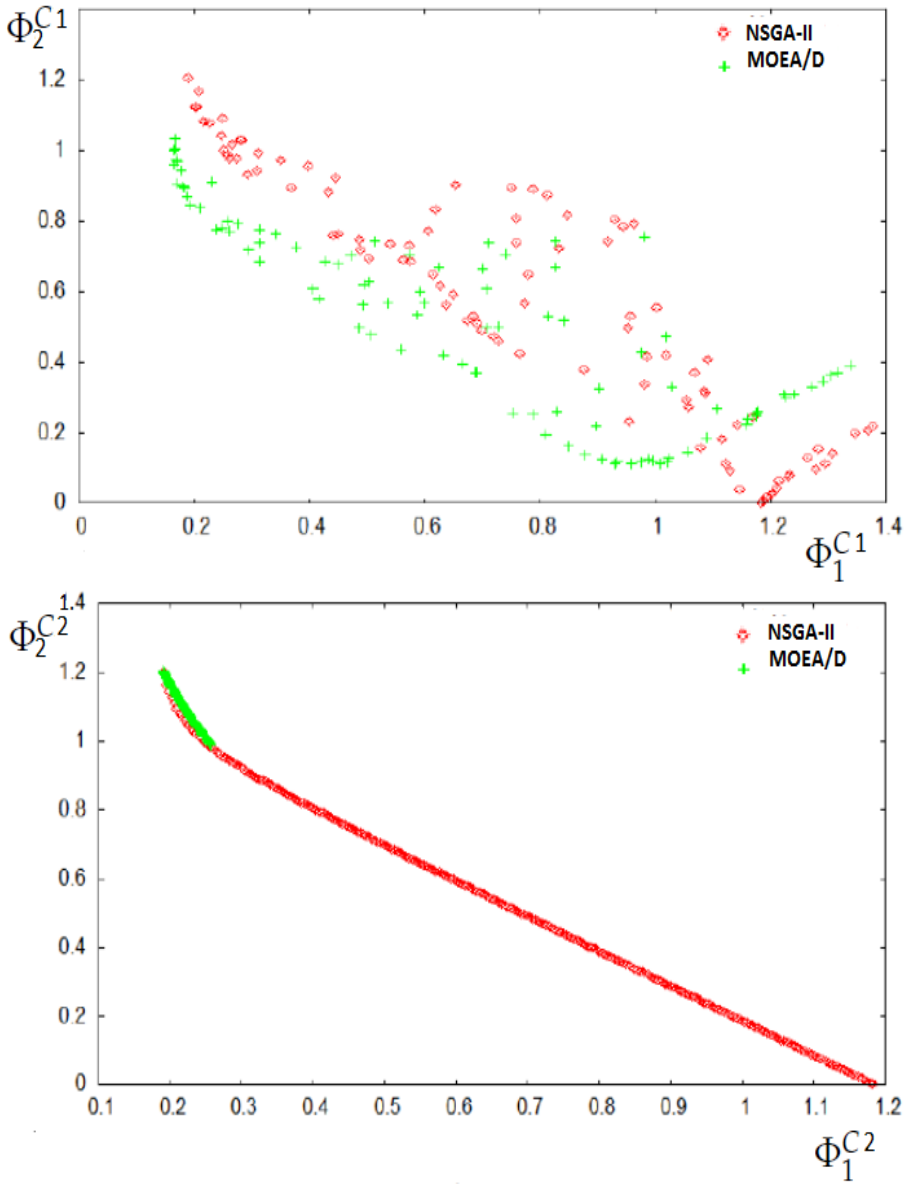


Figure 3.6 Comparison with CoDeMO optimization results (using MOEA/D to solve the subproblems compared to results from conventional many objective optimizations (NSGA-II, MOEA/D) for the facility location problem with 10 objective functions. Here,◇ stands for NSGA-II and + stands for MOEA/D.

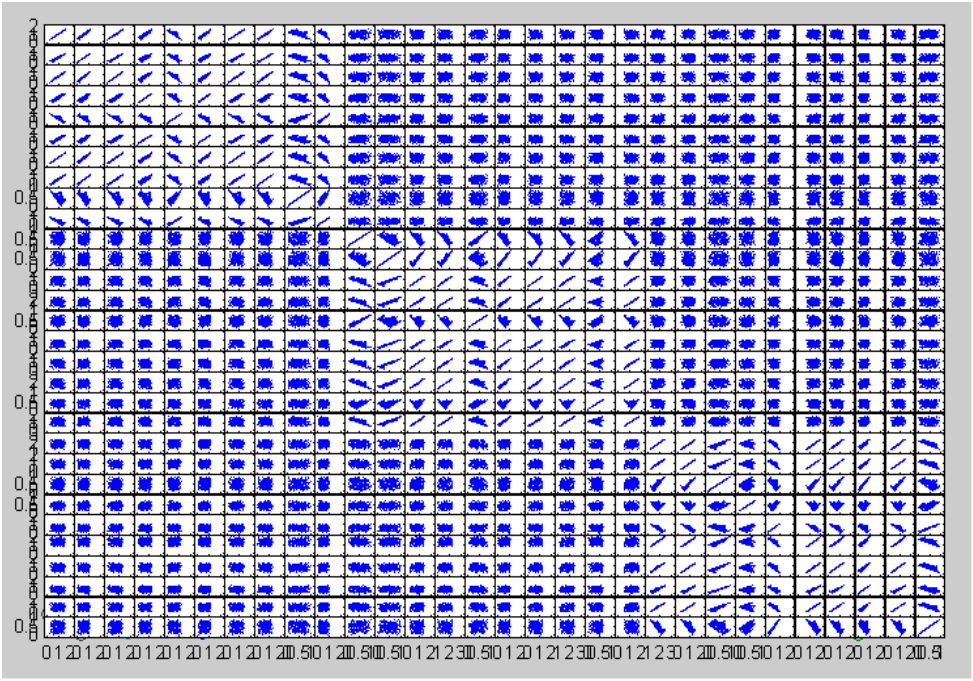


Figure 3.7 Scatter plot matrix for the 30 objective problem.

optimization problems. It interprets objective functions as actors in a social network (complimentary), which might be friends, enemies (conflicting) or neutral with respect to each other. The proof of concept study shows that for problems with a relatively simple underlying this approach works both to reveal the structure and to exploit it by providing more interpretable and also more accurate optimization results. The community detection works well for many-objective optimization and was tested with up to 50 objective functions.

In addition, we pointed to some limitations of the approach. We assume that in many cases the result of the community structure reveals whether or not a decomposition is possible. By pointing at the possibility of higher order interactions we also show that in some non-linear problems, simple structured problems have complex interactions. More examples for using the CoDeMO approach will be given in the subsequent chapters, namely for NK landscape analysis (Chapter 4) and for multilayer network analysis (Chapter 5 and Chapter 6).

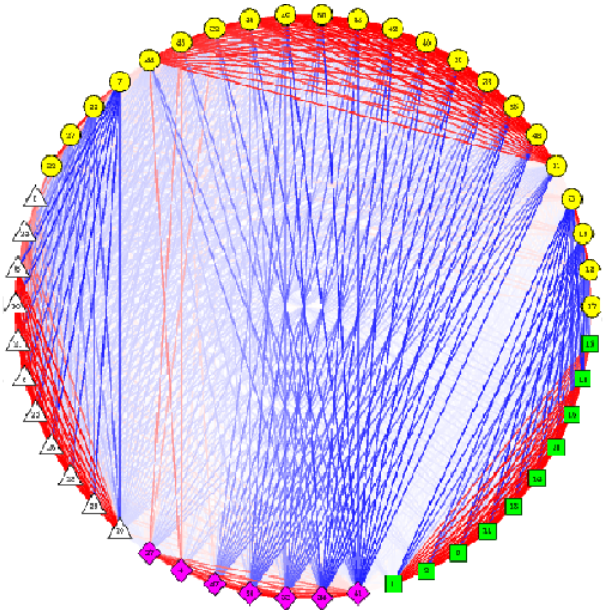


Figure 3.8 Example of a community detection result for 50 objective functions.

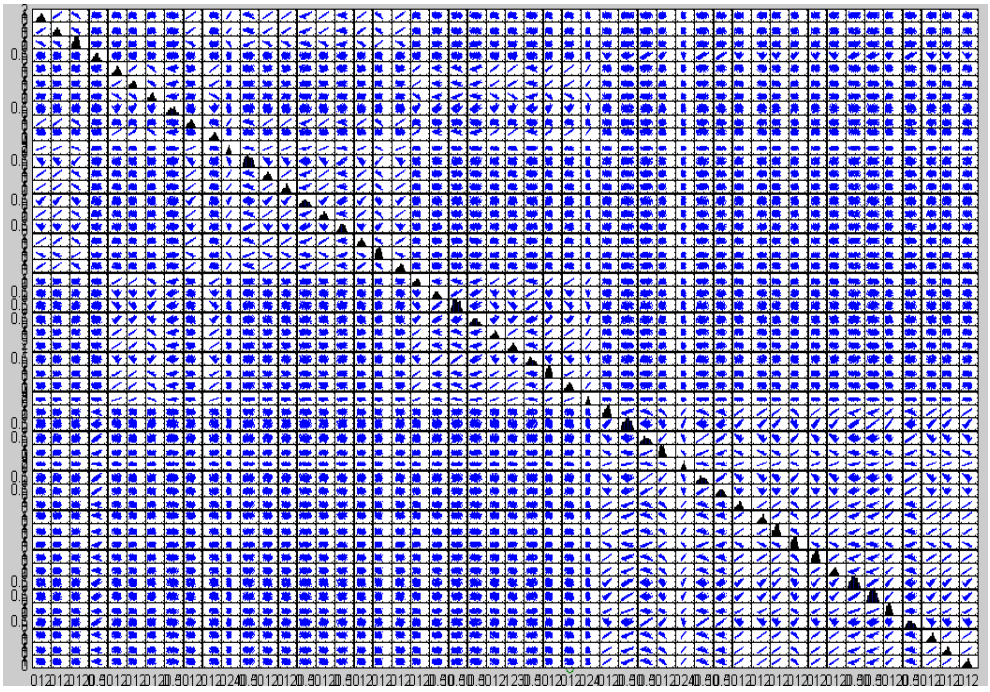


Figure 3.9 Scatter plot matrix for the 50 objective problem.

Table 3.3 Data on centers for the 30-Dimensional problem.

Data: 30-Objective Function	$c_1^{(i)}$	$c_2^{(i)}$	j_1	j_2
f1	1.1	1.2	1	2
f2	0.9	0.8	1	2
f3	1	1	1	2
f4	0.7	1.1	1	2
f5	0.2	0.1	1	2
f6	0.7	1	1	2
f7	1	1.1	1	2
f8	1	0.9	1	2
f9	0.2	0.4	1	2
f10	0.3	0.2	1	2
f11	0.2	0.3	3	4
f12	0.7	0.8	3	4
f13	1	1.1	3	4
f14	1.6	1.7	3	4
f15	0.3	0.4	3	4
f16	0.8	0.8	3	4
f17	0.9	1	3	4
f18	1.7	1.8	3	4
f19	0.5	0.5	3	4
f20	1.3	1.3	3	4
f21	1.7	1.8	5	6
f22	1.1	1.1	5	6
f23	0.7	0.7	5	6
f24	0.4	0.5	5	6
f25	0.2	0.2	5	6
f26	1.5	1.4	5	6
f27	1	1	5	6
f28	0.8	0.7	5	6
f29	1.1	1.1	5	6
f30	0.3	0.2	5	6

4

Community Detection in NK-Landscapes -An Empirical Study of Complexity Transitions in Interactive Networks

NK-Models (or NK-Landscapes) were introduced in Kauffman and Levin, 1987 [30] as models of how the fitness of an organism is related to gene interaction. They also gained popularity in the study of complex organizations (see Anderson, 1999 [3]) and innovation networks (see Frenken, 2000 [22]).

In the classical NK-model the parameter N describes the number of components in a network (genes, agents, or nodes in an distributed system) and each component is associated with a control variable x_i and a trait function or output value f_i. K describes the degree of interaction between these components. For each component of the system ($i \in \{1,\ldots,N\}$) , the value of f_i depends on the value of the variable x_i that is associated with it and k other variables $x_{e(i,1)},\ldots,x_{e(i,k)}$. These k variables are called the *epistatic variables*. The fitness of the NK-landscape is the sum of these values:

$$F(x_1,\ldots,x_N) = \sum_{i=1}^{N} f_i(x_i, x_{e(i,1)}, ..., x_{e(i,k)}) \tag{4.1}$$

In Equation 4.1 the values of $E = e(i,j), i \in \{1,\ldots,N\}, j \in \{1,\ldots,K\}$ are the epistatic matrix that determines which variable interacts with which other variable.

Based on the locality of the epistatic matrix two variants of the NK-landscapes

are distinguished: The epistatic variables of gene i can be *adjacent* with respect to the index i (local interactions) or their choice can be *random* (global interactions). Note, that in case of adjacent epistatic genes the indices are mapped cyclically, i.e. gene N is a direct neighbor with gene 1 (wrap around). This way every gene has two direct neighbors. If more than two epistatic genes need to be defined we collect the genes in an increasingly big radius (2-step neighbors, 3-step neighbors and so on). The notion of neighborhood stems here from the idea of physical location on a DNA, that, for the sake of simplicity is viewed as a ring.

Figure 4.1 shows a visualization of the epistasis structure that arises based on these choices.

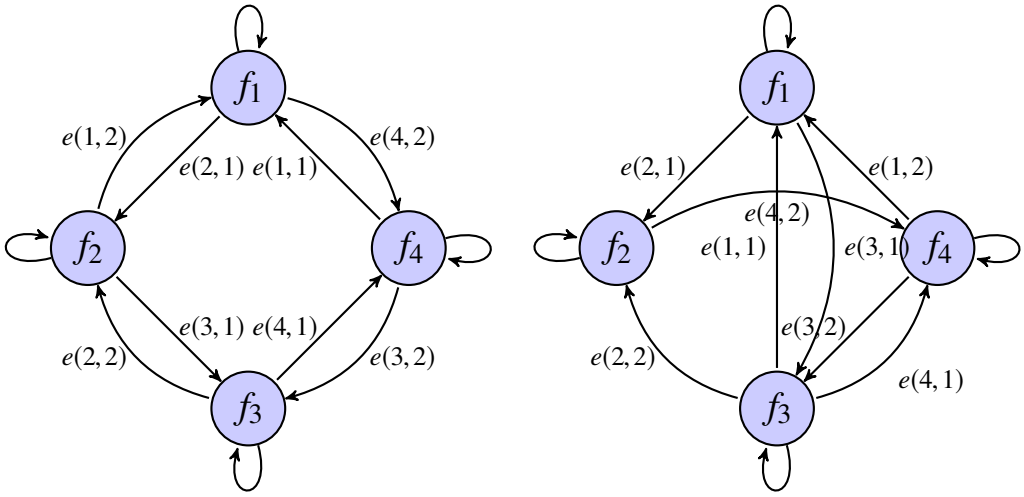


Figure 4.1 (Left hand side) Example of an NK-Landscape epistasis network for $N = 4$, $k = 2$ and adjacent epistatic genes. (Right hand side) Example of an NK-Landscape epistasis network for $N = 4$, $k = 2$ and randomly assigned epistatic genes. The arrows labeled with $e(i, j) \in \{1, \ldots, N\}$ indicate the epistatic genes that influence the gene with index $i \in \{1, \ldots, N\}$ for $j \in \{1, ..., k\}$.

We can now introduce a concrete realization of a function F, for instance by using the classical binary NK-landscape where each variable can only obtain the values 0 and 1. For this we will define 'upper case' F_i component functions that will accept a bit string the components of which correspond to the values of x (Goedel encoding of

the vector):

$$F(x_1, \ldots, x_N) = \sum_{i=1}^{N} F_i(2^0 x_i + 2^1 x_{e(i,1)} + \cdots + 2^k x_{e(i,k)}), \quad \mathbf{x} \in \{0,1\}^N \tag{4.2}$$

Each one of the functions F_i is looked up in a table of size 2^{k+1}. Each of these tables comprises 2^{k+1} random numbers that are sampled from a uniform distribution in $[0, 1]$ (see, e.g. [2]). This was done in order to make the model simple and to not introduce additional complexity in its construction [30]. Subsequent analysis revealed interesting, emergent behavior of NK-Landscapes already on this very basic level. One interesting feature of NK-landscapes is that their properties critically depend on the choice of k. Some interesting properties that depend on the choice of k are summarized in the following list (see also [2]):

- $k = 0$ (no epistasis):
  - The problem is separable.
  - There exists a unique global and local optimum[1].
- $k = 1$
  - A global optimum can be found in polynomial time.
- $k \geq 1$
  - Adjacent epistatic genes: Time complexity for finding a global optimum is in $O(N^k)$.
  - Randomly assigned epistatic genes: Finding a global optimum becomes NP complete; time complexity is in $O(2^N)$ under the assumption that $P \neq NP$.
- $k = N - 1$
  - Random function value assignment; causality is lost and finding the global optimum takes $\Omega(2^N)$ time.

An interesting research question is: What happens to the structure of the problem at the critical transitions from simple ($k = 0$), via polynomially optimizable $k = 1$

[1]Except for degenerate cases, which occur with probability of zero-measure.

or problems with adjacent epistatic genes for greater k, to complex networks $k = 2$ (global interactions) and how does this differ from complete random function value assignments at the level of $k = N - 1$.

In order to study this, we propose to study in more depth the interaction between the components of F, namely the functions F_i. A new perspective to look at this question is to view the F_i as objective functions in a many-objective optimization problem. The novel perspective taken is to view these trait functions as objective functions that seek to contribute to F with the highest possible contribution, or, that seek to obtain the best adaptive value. This yields a multi-objective optimization problem, which can be written as:

$$F_1(\mathbf{x}) \rightarrow \max, \ldots, F_N(\mathbf{x}) \rightarrow \max, \quad \mathbf{x} \in \{0,1\}^N \tag{4.3}$$

The correlation between trait functions can be determined if the input vector is viewed as a random sample. Different trait functions F_i can support each other (positive correlation), be neutral with respect to each other (zero correlation), or conflict with each other (negative correlation).

It is noted that the maximization of each single component function takes time $\Omega(2^k)$ due to the random assignment process. By introducing interaction the complexity of the optimization task grows. So far it is unknown what exactly happens at the transition from binary to ternary interactions, that is from polynomially time solvable to NP-complete problems, and we hope that correlation and community analysis will shed some new light on this.

In summary, the contributions of our work will be as follows. In order to better understand the transitions in complexity in NK-landscapes from the perspective of communities of component functions we will

1. visualize the community structure among the different F_i trait functions using state-of-the-art algorithms from community detection.

2. provide statistics on number of communities and modularity for different values of k, and

3. discuss correlation and squared correlation as a measure of connectedness in both adjacent and random NK-landscapes.

In Section 4.1 the approach will be discussed in detail, i.e. how to perform community detection among the different F_i trait functions, and how to use the squared correlation measure for global statistics. The results will be discussed in Section 4.2.

4.1 · Approach

The correlation matrix used in this chapter is similar to the statistical concept of correlation explained in chapter 2 but more specific to the problem we investigated. The graph that we consider has N nodes. Each node is associated with a component function F_i. Links between the nodes are weighted by the correlation between the two function values. Now, ρ^e_{ij} serves as an estimate of the correlation ρ_{ij} between $\mathcal{F}_i$ and $\mathcal{F}_j$. The value of the correlation ρ_{ij} ranges from perfect anti correlation (-1), via independence (0), to perfect correlation ($+1$).

In the context of optimization of the F_i functions, we can interpret this correlation as follows:

- Positively correlated trait functions can be interpreted as trait functions that support each other. This means that maximizing one function will imply that also the other function will obtain high values.
- Uncorrelated trait functions are considered to be independent of each other. They can be maximized in isolation from each other. The NK-landscape value can be maximized by independently maximizing these traits.
- Negatively correlated objective functions are in a strong conflict with each other, that is an increment of one value typically will lead to a deterioration of the other value. Intuitively, one might expect that if there are many conflicting trait functions the maximization of the NK-landscapes gets more difficult.

The next two examples of a correlation matrix of NK-Landscapes for $k = 0$ and $k = 1$ with $N = 10$ are provided with Table 4.2 and Table 4.2, respectively. Note, that in case $k = 0$ not all correlations between different trait functions are 0, although from the construction of the NK-Landscape all trait functions should be independent in the case $k = 0$. The finiteness of the sampling space makes it however unlikely that also the statistical correlation is exactly the same. For $k = 1$, clearly, some correlations have values higher than zero and higher than all correlations for the previous case. As we will see, this strong correlation will vanish again quickly for values of $k \gg 1$.

Table 4.1 Correlation coefficient matrix for N = 10 and k = 0

1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	1.000	-0.014	0.013	-0.014	0.012	-0.012	-0.032	0.000
0.000	0.000	-0.014	1.000	0.013	-0.014	0.012	-0.012	-0.032	0.000
0.000	0.000	0.013	0.013	1.000	-0.043	0.035	-0.035	-0.097	0.000
0.000	0.000	-0.014	-0.014	-0.043	1.000	0.058	-0.058	-0.161	0.000
0.000	0.000	0.012	0.012	0.035	0.058	1.000	-0.128	-0.358	0.000
0.000	0.000	-0.012	-0.012	-0.035	-0.058	-0.128	1.000	0.358	0.000
0.000	0.000	-0.032	-0.032	-0.097	-0.161	-0.358	0.358	1.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000

In summary, correlation analysis is applied to derive the weights of links between the pairs of component functions. The $N \times N$ correlation matrix is interpreted as a weighted graph with weighs in $[-1, 1]$. This way, it can be analyzed using graph theoretic algorithms and in particular by community detection. Thereafter, statistics on macroscopic properties of the community graphs can be applied to find regularities that might reveal new insights in the critical transition(s) of the landscape's complexity as k grows.

4.2 · Results

Results depicted in Figure 4.4 to 4.8 visualize the concrete results of the community detection obtained and visualized with `Pajek` using Louvain and, respectively, VOS clustering. First, let us summarize results for the Louvain method. Figure 4.4 and, respectively, 4.5 show the transition of community structures for randomly assigned genes and, respectively, for adjacent epistatic genes. The value of k is chosen from 0 to

Table 4.2 Correlation coefficient matrix for N = 10 and k = 1

1.000	-0.237	0.000	0.000	0.000	0.000	0.000	0.000	0.000	-0.246
-0.237	1.000	0.325	0.015	0.013	-0.011	0.010	0.011	0.030	0.000
0.000	0.325	1.000	0.455	-0.005	0.004	-0.004	-0.004	-0.011	0.000
0.000	0.015	0.455	1.000	-0.033	0.020	-0.020	-0.020	-0.058	0.000
0.000	0.013	-0.005	-0.033	1.000	-0.065	0.040	0.044	0.124	0.000
0.000	-0.011	0.004	0.020	-0.065	1.000	-0.334	-0.065	-0.184	0.000
0.000	0.010	-0.004	-0.018	0.039	-0.334	1.000	0.804	0.319	0.000
0.000	0.011	-0.004	-0.020	0.044	-0.065	0.803	1.000	0.355	0.000
0.000	0.030	-0.011	-0.057	0.124	-0.184	0.319	0.355	1.000	0.000
-0.246	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000

$N - 1$.

From the visual impression, it is clear that the highest degree of separation is obtained in the case $k = 0$ for both epistatic link structures (random, adjacent).

The nodes that belong to the same community are indicated by nodes which share the same color. The number of communities reaches its minimum for $k = 2$. A confirmation of this can be obtained when plotting the number of communities over different values of k, which is done in Figure 4.2 for randomly assigned epistatic genes and in Figure 4.3. For adjacent ones, the value of $k = 2$ is a clear optimum in both cases.

This is surprising because it might be expected that for $k = N-1$ all nodes merge to one big community. This is not the case and – in first approximation – might be explained by the fact that nodes can also negatively influence each other (conflicting nodes). Note, that the number of communities grows at a slower rate for adjacent epistatic genes. This coincides with the slower increase of computational complexity [58].

Analogous findings have been made with the VOS clustering method for community detection. Figure 4.7 and 4.8 show the community structures, whereas Figure 4.9 and 4.10 show the results. The correspondence between the two different approaches for community detection underpins that the findings are not an artifact of the method. More dissonance between Louvain and VOS methods is found for the number of communities for high levels of k. However, in all methods the general trend can be observed that the number of communities first decreases and then grows again.

A conjecture we obtained from the pictures is that the correlations or anti-correlations are first very strong and then weaken again. This can be measured by the squared correlation. It is an indication of how much the results of two nodes depend on each other (either positively or negatively). Values close to zero indicate independence of the results at two different nodes. The average squared correlation between nodes in the network is shown in Figure 4.6 – both for adjacent and randomly assigned epistatic genes.

Clearly, both landscapes have a peak at low values of k. It is striking that the peak for the NK-landscape with low values of k has a sharp decay in average squared correlation, whereas the decay of the adjacent case is gradual. Again this coincides with the finding that for randomly assigned epistatic genes a sharp transition in computational complexity appears whereas the transition is gradual for the case of adjacent epistatic genes.

Viewing the plot one might even speculate that the observed phenomena are a *sawtooth transition*. This is found in other complex systems at the edge of chaos and is conjectured to be a universal law for macroscopic observations at the transition from systems with complex, but still predictable behavior, to chaotic and unpredictable systems (see for instance Adriaans [1]). Further analysis of larger models and the theoretical analysis of analogies between the models will be required to either confirm or reject this interesting hypothesis.

In the plots of Figure 4.2 and 4.3 (Louvain) and Figure 4.9 and 4.10 (VOS) we also show the observed modularity of community components. Here we shift a bit more emphasis on the results of the Louvain method as it explicitly seeks to find communities based on modularity. However, the graphical results are less clear for this and to find a trend we also put the tables with the numerical results in Table 4.3 (Random) and Table 4.4 (Adjacent). From these numerical results, it can be obtained that the modularity of the communities first decreases slightly and then goes up again. Clearly, the highest

average modularity is achieved for $k = 0$ in which case nodes are relatively isolated. Again, the peak is pronounced stronger for randomly assigned epistatic genes.

Table 4.3 Comparison of clustering algorithms applied in community detection, Louvain clustering Algorithm compare to VOS clustering algorithm, both with randomly assigned epistatic genes (NC is a number of community and Q is the Value of modularity for community detection)

Louvain Clustering			VOS Clustering		
k	NC	Q	k	NC	Q
0	9	0.842457	0	8	0.8645009
1	6	0.459726	1	5	0.6995902
2	5	0.613504	2	4	0.5636842
3	7	0.630355	3	7	0.6158102
4	7	0.606203	4	6	0.6416844
5	7	0.741914	5	7	0.6716613
6	10	0.701109	6	8	0.6608056
7	10	0.718037	7	10	0.7180642
8	10	0.717084	8	10	0.7285837
9	9	0.757455	9	10	0.7050733

4.3 • Summary

This chapter looked at the graph derived from the correlation structure among the component functions of an NK-landscape model that were treated as objectives of a many objective optimization problem. The results show that the community structure that is detected for this 'correlation graph' does not correspond with the community structure of the epistatic link network which has many components for small values of k and only one big component for $k = N - 1$ (every gene is linked to every other gene). Instead, the correlation network has the lowest number of components

Table 4.4 Comparison clustering algorithm applied in community detection, Louvain clustering Algorithm compare to VOS clustering algorithm with adjacent epistatic genes (NC is a number of community and Q is the Value of modularity for community detection)

Louvain Clustering			VOS Clustering		
k	NC	Q	k	NC	Q
0	9	0.842457	0	8	0.8645009
1	6	0.677228	1	5	0.7196617
2	6	0.714034	2	4	0.6285909
3	6	0.713723	3	5	0.6452446
4	8	0.663882	4	8	0.6670689
5	7	0.656833	5	7	0.6213816
6	9	0.679317	6	7	0.6804879
7	8	0.691417	7	10	0.6805063
8	9	0.680426	8	9	0.7017337
9	10	0.715158	9	7	0.6856205

for $k = 2$. For values lower and higher the number of communities clearly grows. As the critical transition from polynomial time, solvable maximization problems to NP-complete maximization problems appears at the transition from $k = 1$ to $k = 2$ (for random networks) we suspect that these findings might be not coincidental. We show also that the average squared correlation reaches a sharp peak near this value of k. This peak is less pronounced for adjacent epistatic genes which do not undergo a critical transition but a gradual transition in terms of complexity. So far we have only studied the case $N = 10$ and studies on larger networks are required in the future to improve the generality of the findings. A problem that needs to be solved for such studies is how to tame the 'explosion' in the size of the random number tables needed to generate the NK-landscapes. A useful proposal has been made by Altenberg [2], who suggested to re-generate the random numbers on-the-fly when needed and provided a function that can be used for this.

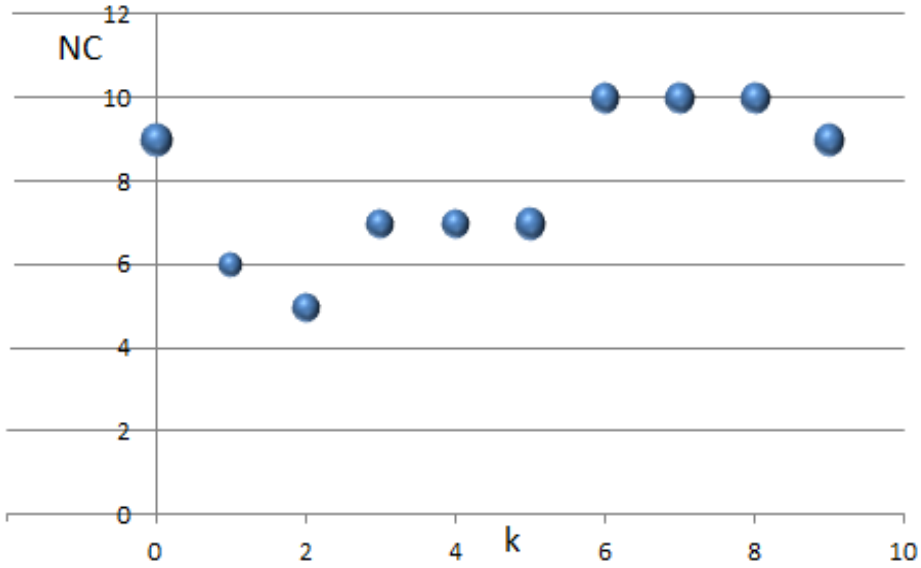


Figure 4.2 Community detection by Louvain clustering algorithm based on randomly assigned epistatic genes.

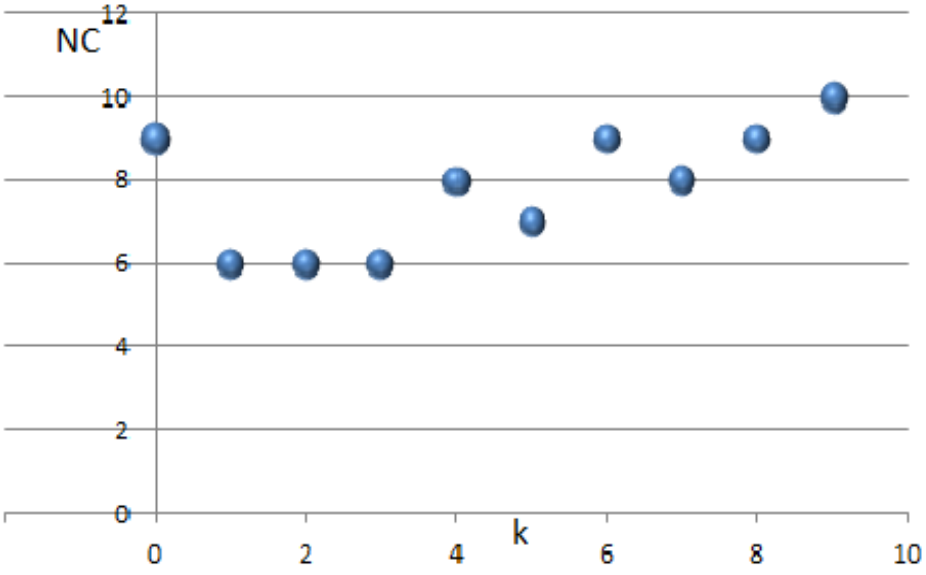


Figure 4.3 Community detection by Louvain clustering algorithm based on adjacent epistatic genes. NC denotes the number of communities found.

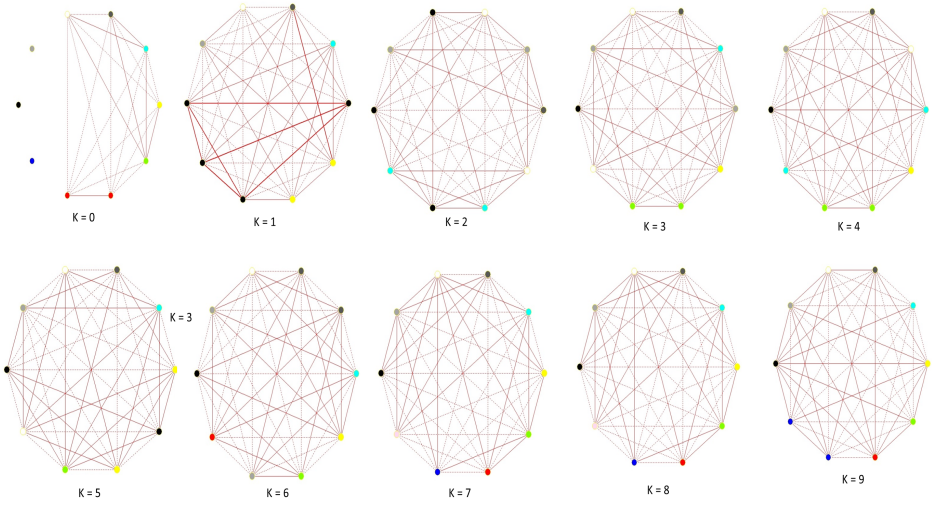


Figure 4.4 Community detection by Louvain clustering algorithm based on randomly assigned epistatic genes. NC denotes the number of communities found.

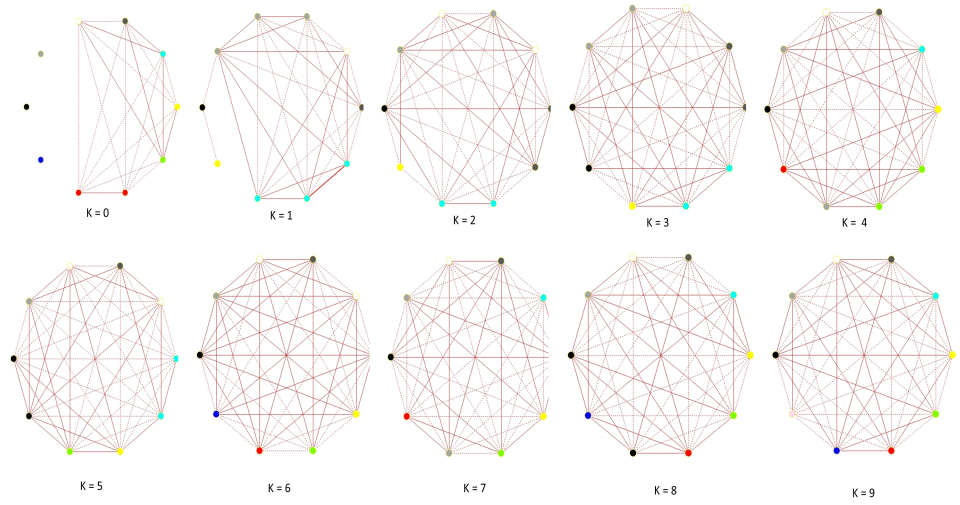


Figure 4.5 Community detection by Louvain clustering algorithm based on neighborhood selection with adjacent epistatic genes.

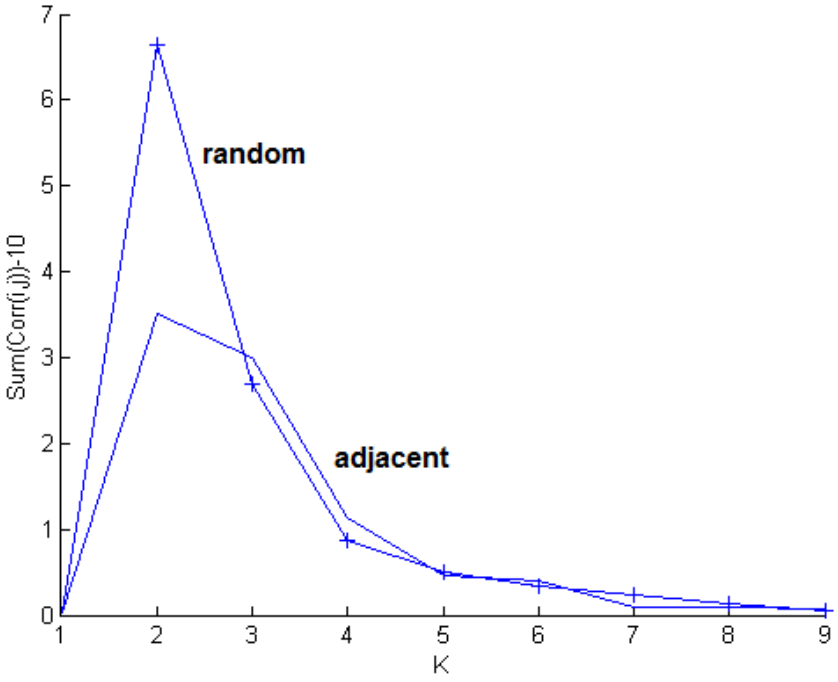


Figure 4.6 Community detection by Louvain clustering algorithm based on randomly assigned epistatic genes.

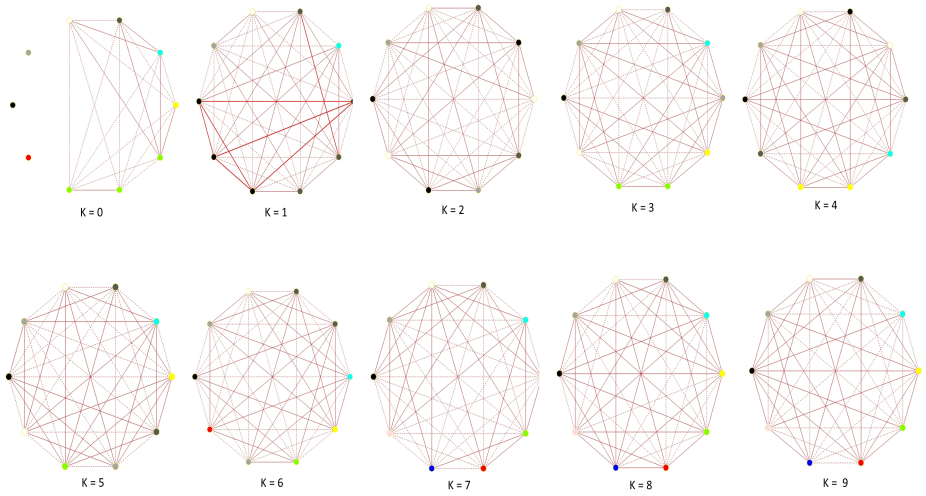


Figure 4.7 Community detection by VOS clustering algorithm based on neighborhood selection with randomly assigned epistatic genes.

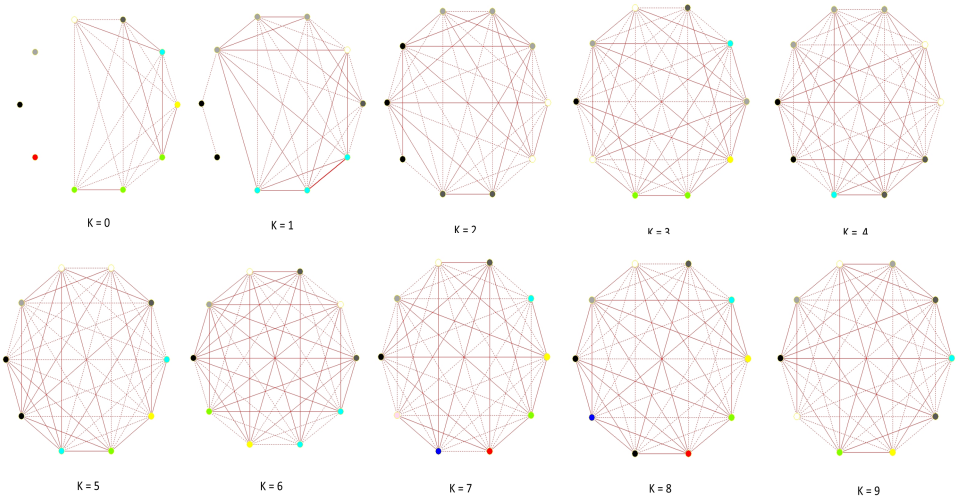


Figure 4.8 Community detection by VOS clustering algorithm based on neighborhood selection with adjacent epistatic genes.

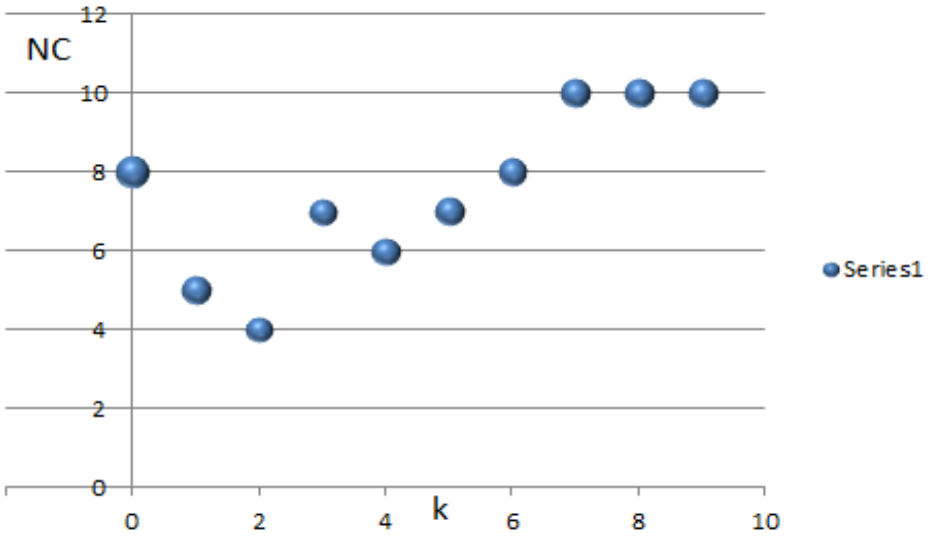


Figure 4.9 Community detection by VOS clustering algorithm based on randomly assigned epistatic genes.

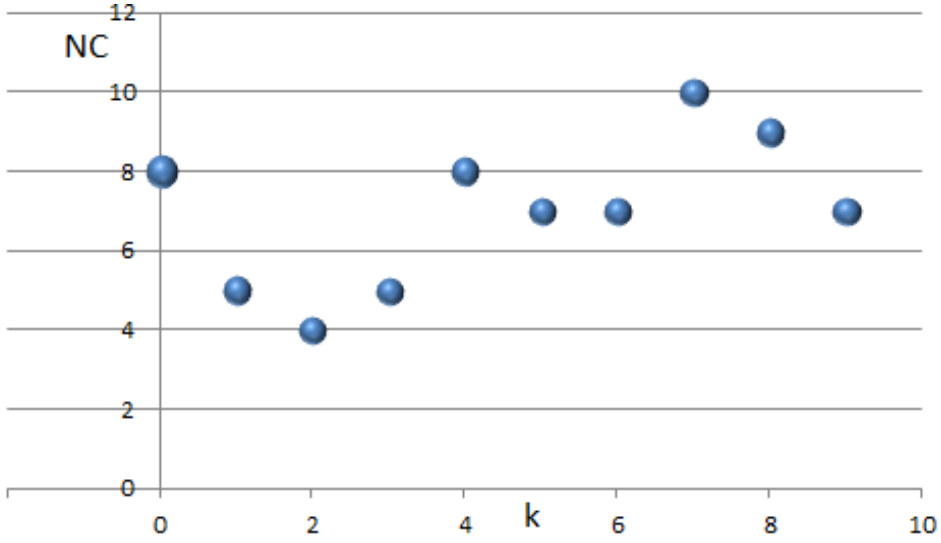


Figure 4.10 Community detection by VOS clustering algorithm based on adjacent epistatic genes.

PART II

5

Modularity Maximization in Multiplex Network Analysis Using Many-Objective Optimization

5.1 • Introduction

In many disciplines, complex systems can be studied through network modeling and analysis. This yields a better understanding of complex phenomena, including conflicting sociology phenomena, spreading of disease, conflicting economic situations, telecommunication systems, biological systems, and networks in engineering. The networks or a collection of nodes are joined in pairs by edges. Clustering such groups of nodes in the network has become an important area of research. Network data becomes increasingly available but is also complex due to the omnipresence of data measurement and inquiry as a recent trend. In this chapter, we will focus on a special class of networks so-called multiplex networks. Often for the same set of network nodes, several or many network layers can be defined. Networks defining trade of different types of commodities is an example and it will provide a case study for this chapter. Other examples include multiplex networks

- in communication via different channels (social media, telephone, peer-to-peer),
- in biology, the different types of signaling networks of trees or plants (via scents, via insects, via underground root networks),
- in sociology, defined by different types of relationships, such as personal friends, relatives, business relationships, which might partially overlap.

This chapter presents a first step in the analysis of such multiplex networks by means of modularity optimization, where modularity is a measure of the quality of how well a partition of a network is representing communities. We consider the optimization of modularity for the different layers as the objective functions. Optimizing several $(2,3)$ objectives simultaneously can be addressed by multi-objective optimization and many (>3) objectives by many-objective optimization resulting in a high dimensional Pareto front. By computing the Pareto fronts of pairs of different layers we find relationships between the objectives. Layers can be in conflict with each other, meaning that they yield very different optimal modularity structures. They can be also complementary, meaning that maximizing the modularity of the one layer also maximizes the modularity of the other layer. In this case, it is possible to merge the layers without losing essential information. Finally, it is also possible that the maximization of modularity of one layer does not affect the optimization of the modularity of another layer, in which case the problem could be easily decomposed.

5.2 · Related Work

To optimize many objectives simultaneously various approaches have been developed. Some of them aim at reducing complexity, such as Objective Reduction in Many-objective Optimization: Linear and Nonlinear Algorithms [50], Reducing Complexity in Many-Objective Optimization Using Community Detection [40], and Objective Reduction Based on Nonlinear Correlation Information Entropy [57]. Other approaches are based on Evolutionary Multi-objective optimization (EMO) extended for dealing with many objectives, cf. [33]. In this chapter, the CoDEMO framework from Chapter 3 is applied. The objective functions are the modularities achieved for different layers.

5.3 · Many Objective Optimization Approach to Community Detection in Complex Networks

Our research approach is to perform many-objective optimization of network modularity by computing and visualizing a matrix of Pareto fronts for pairs of objectives. Then we use community detection algorithms to group objective functions in order to understand and visualize the conflict or correspondence of community structures w.r.t. different edge sets. For every edge set, one objective function is defined, which is to maximize the modularity of this edge set. The search space X is the space of all partitioning of the node sets. In this way for a multiplex network G with layers G_1,...,G_M

we define M objective functions $Q_1 : X \to \mathbb{R}_0^+$, $Q_2 : X \to \mathbb{R}_0^+$,...,$Q_M : X \to \mathbb{R}_0^+$. All objective functions are to be maximized. Our first goal is to compute Pareto optimal solutions. Then we analyze projections to pairs of objective functions (corresponding to pairs of layers), in order to understand the relationship between layers in terms of modularity structure. In this way, we aim to gain insight into essential aspects of the community structure of a given multiplex network.

5.4 · Network Analysis Method

Given as an input a multiplex network with M layers represented by a set of graphs G_1,..., G_M, the approach is called Pareto front Modularity for Multiplex Network (PaMoPlex). Similar to the CoDeMO approach, discussed in chapter 3, it is a workflow consisting of several subsequent analysis steps. It is summarized in a work flow which consists of two major phases: (1) Preparation of data by optimization, (2) Analysis of data. The preparation of data in step (1) of the analysis consists of solving optimization tasks to find non-dominated solutions. In order to get more precise results, we also compute single objective optima and marginal Pareto fronts for every pair of two objective functions (between modularities as objective functions associated with two layers, each). The first phase is summarized in the next three steps:

- Single Objective Optimization: Optimize the modularity of each network separately using evolutionary single objective optimization based on a genetic algorithm.
- Many-Objective Optimization: Optimize the modularity of network, all layers together, as one unit in a multiplex network. For this, we use M-objective optimization algorithms.
- Pairwise Pareto-Front Computation: Optimize modularity for pairs of objectives.

The optimization methods are evolutionary multi-objective optimization based on NSGA-II [16], MOEA/D [31] and SMS-EMOA [56], [8] (population size: 100, number of generations: 2000). For small examples, we use a complete enumeration of partitions. Since NSGA-II is not really appropriate for a Many-Objective Optimization problem we rely on the MOEA/D and SMS-EMOA algorithms for the experiment.

In the second phase, the obtained data are analyzed. This is conducted in the following three steps:

- Matrix of Pareto Fronts Analysis: Visualization of Pareto Fronts is done on a plot matrix, where each tile with $j \in 1, ..., M, i \in 1, ..., M, j > i$ consists of a plot of a Pareto front of tradeoffs between objectives Q_i and Q_j (see Figure 5).

- Correlation Heat Map Analysis: Computation of the correlation coefficients matrix from the projections of the output of many-objective optimization. The heat map has as many rows and columns as the number of network layers (or objectives). The Pearson correlation coefficients of the projected 2-objective function vectors have values in the range of [-1,1] for each pair of objective functions; see Table 2 for an example. In the heat map, see Figure 5.6 for an example, blue color represents positive correlations, whereas red color represents negative correlations. The intensity (darkness) and size of the colored square in each matrix cell grow with the absolute value.

- Community Analysis: This tool is based on the result of the correlation analysis. The correlation matrix is used for community detection by the graph-theoretic algorithm to detect communities using the information of correlation coefficients matrix and interpreting it as edge weights. Here the analysis proposed by Maulana et al. [40] is used, where the edge weight is determined by the absolute value of the correlation coefficient. This leads to a separation of independent communities of layers. Conflicting communities are placed opposite to each other (see Figure 5.5).

Further details on the analysis of examples and interpretation of results will be discussed in the subsequent sections.

5.5 · Case Study and Analysis

As an illustrative example on how to interpret results of multi-objective modularity optimization, we computed the exact Pareto fronts for three synthesized multiplex networks consisting of only two layers each. The networks and the corresponding Pareto fronts are displayed in Figure 5.1, Figure 5.2, and Figure 5.3. Red edges denote edge weights of 3, blue edges represent edge weights 1, and omitted edges have weight 0. A complete enumeration of all 203 possible partitioning was used to compute the exact Pareto fronts (cf. Bell 1934 [7]).

5.5.1 · Analysis on Synthesized Multiplex Networks

The first network in Figure 5.1 is a multiplex network where the maximization of modularity is conflicting, due to non-overlapping communities w.r.t. both layers. The linear Pareto front indicates a strong conflict between the maximization of two types and it is difficult to find a compromise solution that optimizes both objectives at the same time.

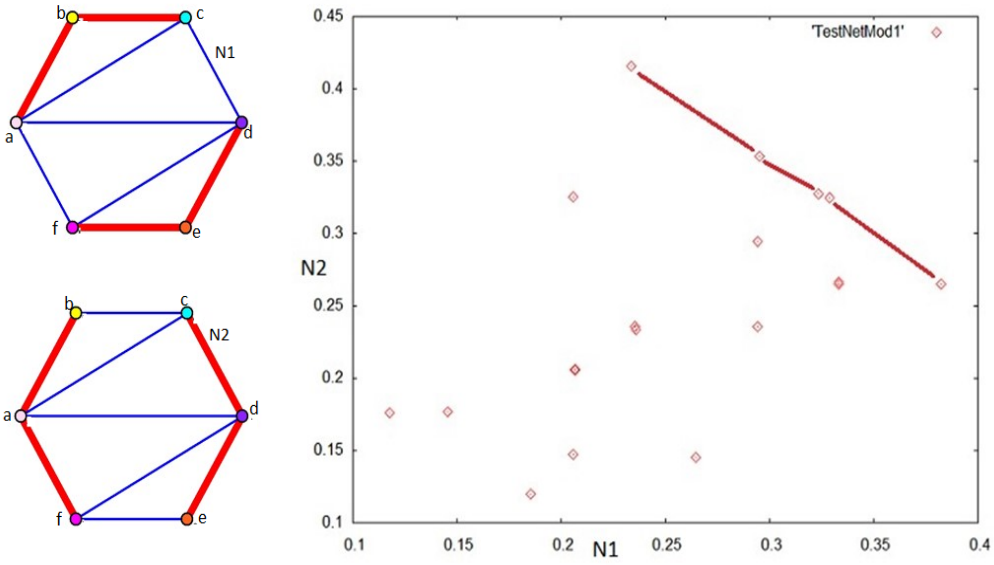


Figure 5.1 A visual depiction of the Pareto front for network modularity between two network layers N1 and N2 corresponding to highly conflicting objectives function

In the second example, in Figure 5.2, the optimal modularity for the first network is achieved by grouping the upper nodes in the graph, while for the second network it is important to group the lower nodes. Thereby the value of the modularity is widely indifferent to how the remaining nodes are grouped. This represents a case where the modularity optimization for the two layers is almost independent and the Pareto front has a knee point solution where both objective functions almost obtain their maximum. The correlation is close to zero. Finally, the third example in Figure 5.3 shows a multiplex network consisting of two equal edge sets. Here, solutions can be found that cluster for one layer optimally w.r.t. modularity necessarily also do so for the modularity of the second network. In other words, optimizing one network coincides

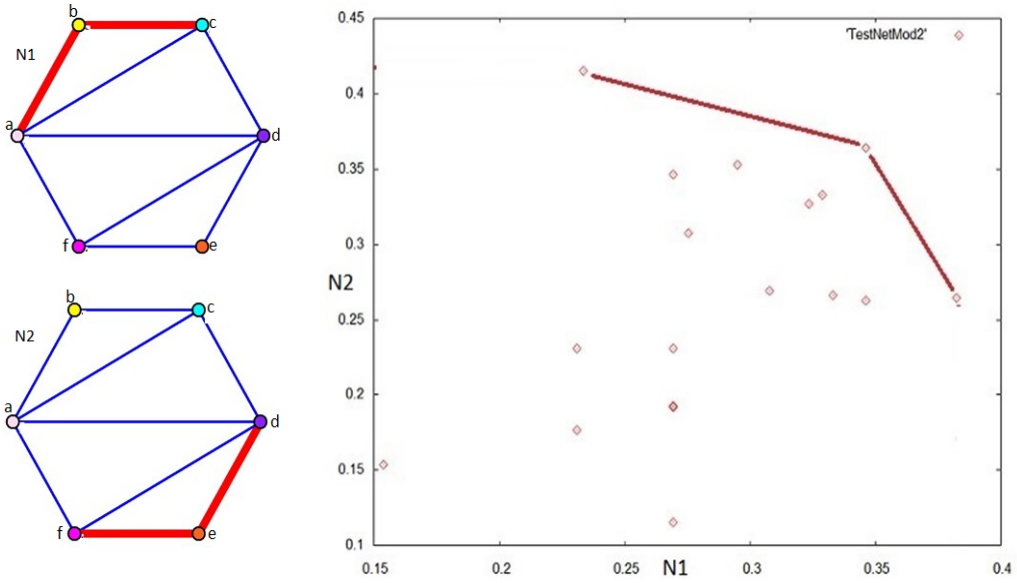


Figure 5.2 A visual depiction of the Pareto front for network modularity between two network layers N1 and N2 corresponding to highly correlated objectives function

with optimizing the other network. This is indicated by a perfect correlation between the modularities of sampled points even for random inputs. The Pareto front consists of only a single solution. In real-world applications, it is of course not so obvious how the structure of the Pareto front looks like. These three examples should be seen as boundary cases, which can help to interpret and understand the observed shape of Pareto fronts in such real-world networks.

5.5.2 · Economic Trade Multiplex Network Analysis

Next, a full PaMoPlex analysis on an economic dataset is provided. The data originates from network economy (trade data) using import-export Commodityodities network between countries in 2011 (see [39], Appendix). The data represents the import-export relationships between countries of the world, disaggregated for different traded commodities. This network can be defined as a multiplex network composed of many layers, where each layer is given by a different commodity. The nodes are given by 207 countries. A link between two countries in the $i-th$ layer defined as the weight will exist if there is trade between them in the $i-th$ commodity, for $i \in 1, ..., 11$. Data are presented in matrix form: rows and columns represent countries, and the entries of

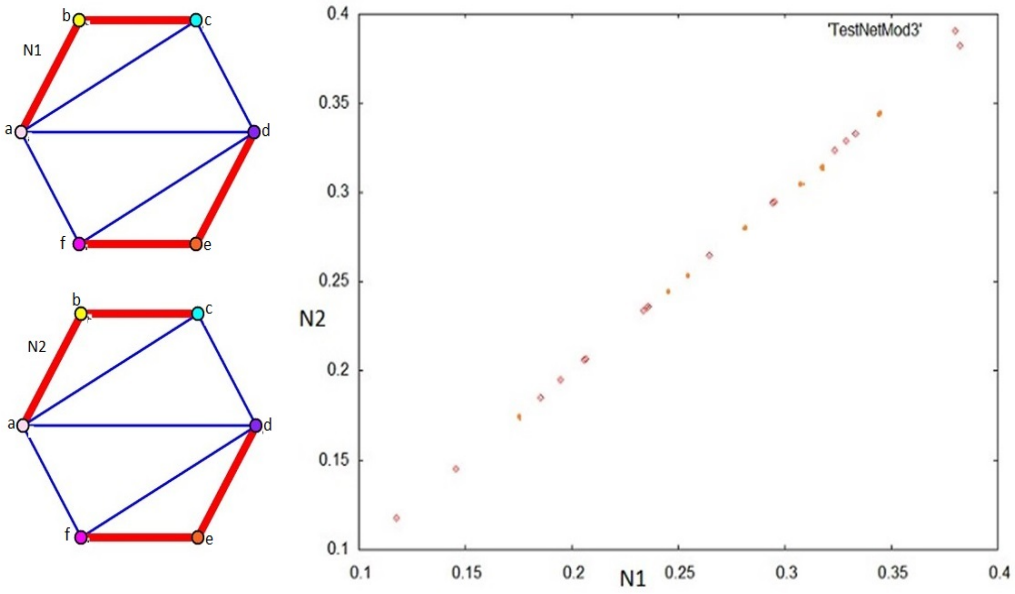


Figure 5.3 A visual depiction of the Pareto front for network modularity between two network layers N1 and N2 corresponding to very strong correlation of objectives function

the matrices are the volumes of trade. It is, therefore, a weighted multiplex network. The general classification is based on 96 different commodities. The classification is performed by grouping together similar commodities; this procedure leads to 11 aggregated 'super-commodities'.

The single objective optimization was conducted by a standard Louvain method and by a genetic algorithm. In all cases, the genetic algorithm found a better result. The results are summarized in Table 5.1 A typical number of communities when maximizing modularity are between 5 and 9.

The genetic algorithm is from the software package JMetal (gGA). It has population size 2000 and 100 generations were conducted. The default parameter settings for the genetic operators were used (http://jmetal.sourceforge.net/, February 2015). We suppose that by tuning of parameters better results can be achieved, but defer such studies to future research in order to focus more on the overall analysis method in this chapter.

The many-objective optimization yields a Pareto front that is embedded in an 11-dimensional space. The analysis of the correlation and community between objectives was conducted following the approach mentioned in [40]. From this, we compute the

Trading network	Louvain method		Genetic Algorithm	
	Modularity	NC	Modularity	NC
Trade 1	0.34392	9	0.35162	9
Trade 2	0.34794	9	0.35225	9
Trade 3	0.30513	9	0.30801	8
Trade 4	0.33691	7	0.33771	7
Trade 5	0.29084	6	0.29968	6
Trade 6	0.26811	5	0.27008	5
Trade 7	0.24781	7	0.24873	7
Trade 8	0.18622	6	0.18863	5
Trade 9	0.29881	5	0.29882	4
Trade 10	0.22961	5	0.22966	4
Trade 11	0.15493	4	0.15494	4

Table 5.1 A modularity for each single network based on single objective optimization using genetic algorithm. From the table, NC is a number of community

heat map of correlations between objectives (Figure 5.6) and the community structure (Figure5.5). The results are also reflected in the Pareto front plot matrix (Figure 5.4). Our interpretation of these results is as follows: Strong conflicts occur between Q_3 and Q_8, Q_3 and Q_9, Q_1 and Q_8, Q_4 and Q_5. Q_1 and Q_2, Q_1 and Q_3, Q_4 and Q_{11}, Q_4 and Q_{10}, Q_4 and Q_{11}. From the analysis we can, for instance, conclude that for trade-networks of Q_3 and Q_8 the countries cannot be clustered in a way that community structures for both groups of commodities are well represented. On the contrary, for Q_1 and Q_2 there exists a clustering that represents the community structures for both communities very well (See the description of the data). It seems logical that the main agricultural products of a group Q_1 and Q_2 appear to adhere to similar trade community structures, whereas for the very disjoint products in group Q_3 and Q_8, it might have been difficult to predict a priori how their trade networks will overlap.

5.6 · Summary

This chapter showed how to apply many-objective optimization for the analysis of multiplex networks. Different ways on how to analyze the community structure in multilayer networks were introduced, all relying upon data from many-objective optimization. First, we discussed the meaning of the Pareto fronts between modularities

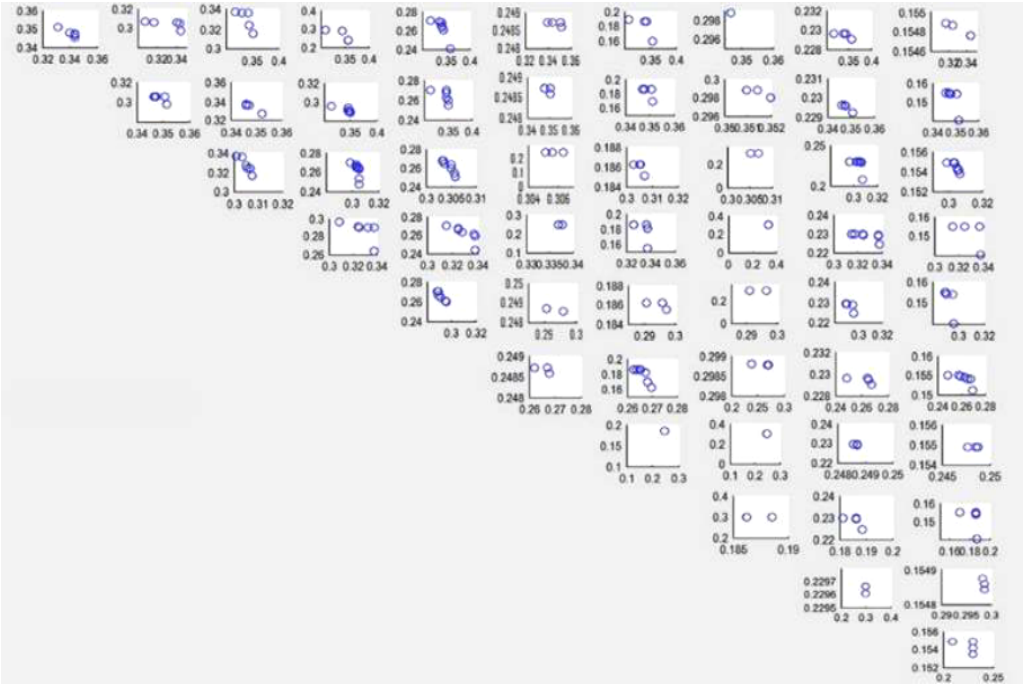


Figure 5.4 Pairwise Pareto Fronts Matrix for Economic Trade Network Analysis

by exact computations of Pareto fronts on three illustrative examples, which represent important boundary cases. Then, on the example of trade networks for commodities, we performed a full analysis. First, we generated data using many-objective optimization, bi-objective optimization (of any pair of layers), and single objective optimization (of any single layer). The results were analyzed using three tools suggested here: Correlation heatmap, the community of objectives analysis, and the Pareto-front plot matrix. These were computed for an economic trade network with 11 groups of commodities. Clearly, a grouping emerges in terms of complementarity and/or in terms of indifference. NSGA-II, SMS-EMOA, and single-objective genetic algorithms can be used as a search engine.

Description of the data

This section describes The description of the data from selected commodities in trade network: Due to space limitations, we will not go in detail about economic trade

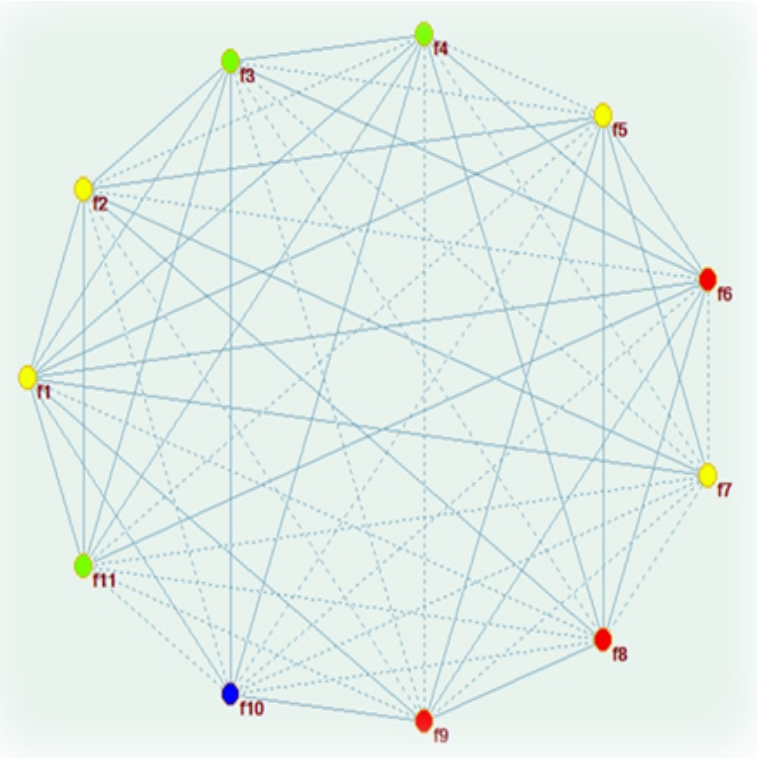


Figure 5.5 community structure for many-objective optimizations of the 11 node trade network

data, but briefly describe those mentioned above:. The data are about economics trading commodities between countries in 2011. Number of country are 207, the numbers of commodities are 96 commodities and grouping in 11 group of commodities described by Q_1 to Q_{11}. For brief explanation, we describe some group of commodities

- Q_1 = Live animals, Meat and edible meat offal, Fish, crustaceans and aquatic invertebrates, Dairy produce; birds eggs; honey and other edible animal products
- Q_2 = Live trees, plants; bulbs, roots; cut flowers and ornamental foliage tea and spices; Edible vegetables and certain roots and tubers; Edible fruit and nuts; Citrus fruit or melon peel; Coffee, tea, mate and spices; Cereals; Milling products; malt; starch; insulin; wheat gluten; Oil seeds and oleaginous fruits; miscellaneous grains, seeds and fruit; Industrial or medicinal plants; straw and fodder
- Q_3 = Lac; gums, resins and other vegetable sap and extracts Vegetable plaiting

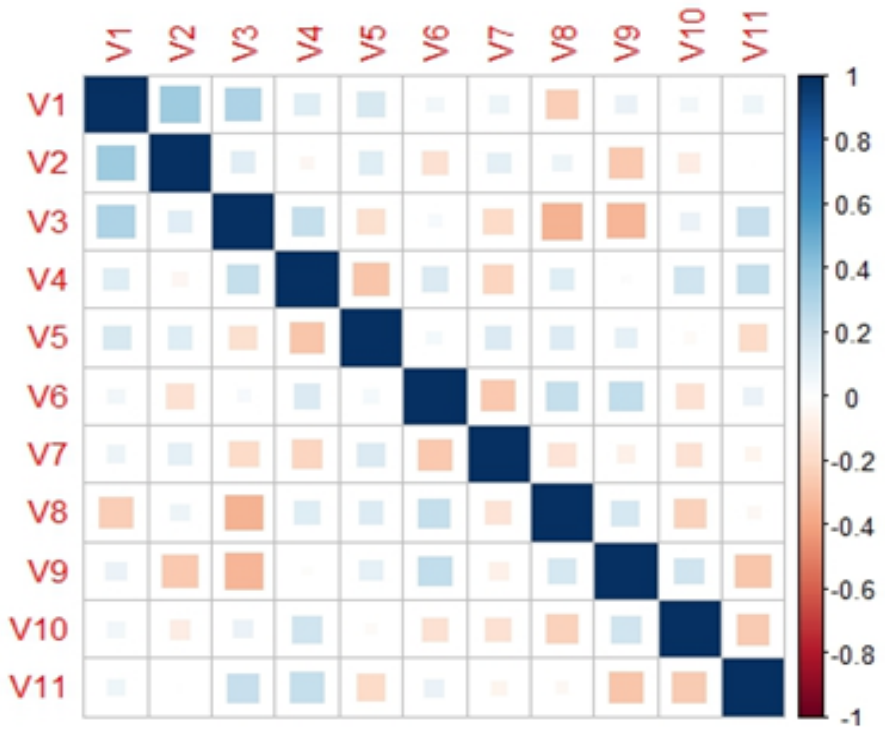


Figure 5.6 Correlation heat map for many-objective optimizations of 11 node trade network

materials and other vegetables products; Animal, vegetable's fats and oils, cleavage product, etc; Edible preparations of meat, fish, crustaceans, mollusc's's or other aquatic invertebrates; sugars and sugar confectionery; Cocoa and cocoa preparations; Preparation of cereals, flour, starch or milk; bakers wares; Preparations of vegetables, fruit, nuts or other plant parts; Miscellaneous edible preparations; Beverages, spirits and vinegar; Food industry residues and waste; prepared animal feed; Tobacco and manufactured tobacco substitutes

- Q_4 = Salt; sulfur; earth and stone; lime and cement plaster, Ores, slag and ash, Mineral fuels, mineral oils and products of their distillation; bitumen substances;mineral wax, Inorganic chemicals; organic or inorganic compounds of precious metals, rare-earth metals, of radioactive elements or of isotopes, Organic chemicals, Pharmaceutical products, Fertilizers, Tanning or dyeing extracts; tannins and derivatives; dyes, pigments and coloring matter; paint and varnish; putty and other mastics; inks, Essential oils and resinoids; perfumery,

cosmetic or toilet preparations, Soap; waxes; polish; candles; modeling pastes; dental preparations with basic of plaster, Albuminoidal substances; modified starch; glues; enzymes

- Q_7 = Silk, including yarns, woven, fabric thereof Wool, animal hair, including yarn and woven fabric, Cotton, including yarn, woven fabric thereof, Other vegetable textile fibers; paper yarn and woven fabrics of paper yarn, Man-made filaments, including yarns and woven fabrics, Man-made staple fibers, including yarns and woven fabrics, Wadding, felt and non-wovens; special yarns; twine, cordage, ropes and cables and article thereof.

- Q_{11} = Optical, photographic, cinematographic, measuring, checking, precision, medical or surgical instruments/apparatus; parts and accessories, Clocks and watches and parts thereof, Musical instruments; parts and accessories thereof, Arms and ammunition, parts and accessories thereof, Furniture; bedding, mattresses, cushions, etc.; other lamps and light fitting, illuminated signs and nameplates, prefabricate buildings, Toys, games and sports equipment; parts and accessories, Miscellaneous manufactured articles, Works of art, collectors pieces and antiques.

(See COMTRADE 96 Classification of commodities for 2011 on http://comtrade.un.org/db/mr/rfCom
Moreover we use the following grouping of commodities

- from 1 to 5: Commodity01;
- from 6 to 12: Commodity02
- from 13 to 24: Commodity03
- from 25 to 35: Commodity04
- from 36 to 40: Commodity05
- from 41 to 49: Commodity06
- from 50 to 56: Commodity07
- from 57 to 67: Commodity08

- from 68 to 82: Commodity09
- from 83 to 88: Commodity10
- from 89 to 96: Commodity11

The commodity data we used was from 2011 for all 207 countries.

6

Towards Many-Objective Optimization of Eigenvector Centrality in Multiplex Networks

6.1 · Introduction

Identifying a set of key players in a network is an important research problem in many disciplines:

- In trading economy, it is important to know which countries are central in trade routes and networks of commodities and need to be stable in order to guarantee the long-term economic sustainability of economic networks.
- In political campaigning, it is important to identify a key player for reaching a large number of potentially interested people or people that should be made aware of a news item or political idea.
- In biology, ecosystems can be understood as networks of organisms. For instance, when maintaining a forest, it is critical to know which trees or organisms are most important to keep the forest in a healthy state by protecting them. There can be multiple networks that need to be considered, such as food webs, and signaling networks for communication and finding mating partners.

In the above problems, each node participates in multiple networks and high centrality in one network might not imply high centrality in another network. To better understand centrality concepts in such problems, in this work we will focus on a special class of

networks – so-called multiplex networks – which are sets of networks which share the same set of nodes but differ in their links. As practical application domain We consider networks defining trade in different types of commodities.

Our paper presents the first step in the analysis of such multiplex networks by means of network centrality maximization, where network centrality is the most influential node in the network. We consider the optimization of network centrality in different layers (edge sets) as the objective functions. Optimizing several (3; 4) objectives simultaneously can be addressed by multi-objective optimization and many (> 3) objectives by many-objective optimization resulting in a high dimensional Pareto front. Since such high dimensional Pareto fronts are difficult to analyze, we also compute the Pareto fronts in pairwise different layers and analyze relationships between the objectives.

Layers can be in conflict with each other, meaning that they yield very different structures of centrality. They can be also complementary, meaning that maximizing the centrality in one layer also maximizes the centrality of the other layer. In this case, it is possible to merge the layers without losing essential information. Finally, it is also possible that the maximization of centrality of one layer does not affect the optimization of the centrality of another layer, in which case the problem could be easily decomposed.

6.2 · Related Work

Many-objective optimization is to optimize many objectives simultaneously; in this direction, various approaches have been developed. Some of them aim at reducing complexity, such as Objective Reduction in Many-objective Optimization: Linear and Nonlinear Algorithms [9], Reducing Complexity in Many-Objective Optimization Using Community Detection [39], and Objective Reduction Based on (Non Linear Correlation Information Entropy) [18]. Other approaches are based on Evolutionary Multi-objective optimization (EMO) extended to deal with many-objectives, cf. [32]. Related to multi-objective and many-objective optimization for network analysis such as Multi-Objective Optimization to Identify Key Players in Large Social Networks [25]. Some researchers did Multi-Objective Optimization for community detection / network clustering such as in [56], [8], [16], [31]. A Maximal Clique Based Multi-Objective Evolutionary Algorithm for Overlapping Community Detection [59], Overlapping Community Detection Through an Improved Multi-Objective Quantum-

Behaved Particle Swarm Optimization [36], and Community Detection From Signed Social Networks Using a Multi-Objective Evolutionary Algorithm [63].

In this chapter, we will follow the a similar approach to the multiplex network community detection (PaMoPlex) that was outlined in Chapter 5. Different to Chapter 5, not modularity, but centrality is considered as a maximization objective.

6.3 · Many-Objective Optimization of Network Centrality in Multiplex Networks

Our research approach in this chapter is to perform many-objective optimization of network centrality by computing and visualizing a matrix of Pareto fronts using two different approaches:

- By computing Pareto fronts for each pair of objectives and analyzing the results in a correlation matrix.
- By computing the Pareto front of the set of full-length objective vectors in $\mathbb{R}^m$.

Recall, that for every layer of the network one objective function is defined, which is to maximize the eigenvector centrality of that layer. In this way for a multiplex network G with layers $G_1, ..., G_m$ we define m as the number of objective functions.

Each node in the network is either dominated or non-dominated. A point of the node is said to be non-dominated if there is no other point which is better or equal in all criteria (all centralities in different layers) and better in at least one criterion (one layer). To compute the non-dominated subset from a finite set of n solutions, the algorithm by Kung, Luccio and Preparata is the fastest known approach [34], It accomplishes this task with a time complexity $O(n \log n)$ for $m = 2, 3$ and $O\left(n(\log n)^{m-2}\right)$ for $m > 3$.

We use the computation of pairwise Pareto fronts in order to understand and visualize the relationship of different layers with respect to centrality, i.e. whether or not and to which extent they share central nodes. Computing Pareto fronts of all objective functions can be used to easily recognize how many nodes are in the Pareto front. In order to compute a more fine-grained ranking of nodes, we compute the dominance rank of a node in the second analysis. Non-dominated solutions are of rank 1, solutions that are only dominated by rank 1 solutions are of rank 2, and so on. The computation of the rank only marginally increases the time complexity [13].

6.4 · Case Study and Implementation

6.4.1 · Analysis on Artificial Multiplex Networks

As an illustrative example on how to interpret results of the many-objective optimization based on network centrality, we started with computing the exact Pareto fronts for artificial multiplex networks. First, we did so for pairwise network layers, and then for all layers of the network.

The network was generated as a random graph based on the Erdős and Rényi model. Starting from a complete graph, each edge has a probability of $1 - p$ to be removed from the network. For a certain number of nodes ($m = 100$) and a certain probability ($p = 0.1$) we generated 11 layers for the experiment. We will denote them with g_1 to g_{11}. We chose eleven layers of the multiplex network, in order to later compare our results with the empirical economic trade multiplex network, which also has eleven layers.

The synthetic network based on Erdős-Rényi can be seen in Figure 6.1. From layer g_{11} of this artificial network, we compute the degree distribution. It is visualized as a histogram in Figure 6.2 and it has the shape of a binomial distribution, as it is typical for Erdős-Rényi graphs. Then we compute the centrality based on eigenvector centrality and the results are the vector of centralities of each node (100 nodes) in each layer. By dividing by the biggest eigenvalue (λ), for each layer, all values are normalized from 0 to 1.

For those 11 layers of the network, with eigenvector centralities of nodes in each layer, the next step is to compute non-dominated sets either by pairwise optimization or by many-objective optimization. The Pareto front is generated by full enumeration of node centrality of every node in every layer. Since the number of nodes in the Erdős-Rényi random graph is 100, we have 100 inputs for each layer. Each node has different centrality in different layers (objective functions) and by computing Pareto fronts for each pair of two layers, insight into the complementarity of layers can be gained.

For instance, in Figure 6.3, we can see that pairwise optimization for layers g_1 and g_2 yields six non-dominated nodes (marked by red points). The other nodes are not in the Pareto front but get a different color based on their dominance rank. Table 6.1 shows a partial list of node rankings of the Pareto front. There are several nodes in the second ranking, and the lowest ranking is 16. In Figure 6.4 all pairwise Pareto fronts and the correlation of the two objectives (layers) are compared with each other. The

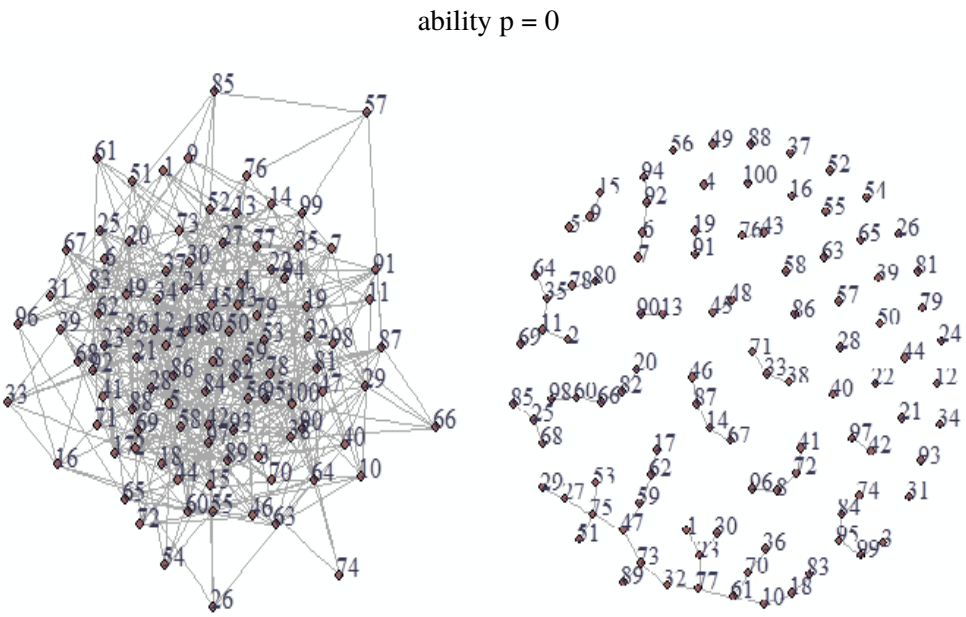


Figure 6.1 Erdős-Rényi random graph example with m = 100. The left network is generated with probability P = 0.1, and the one to the right with probability P = 0.01.

correlation strength of each pair is emphasized by color: Pink pairs have the highest correlation, green is in the middle, and yellow have the lowest correlation. There are no significant differences between the correlations, which can be attributed to the random nature of the network.

6.4.2 · Analysis on Trade Economic Multiplex Networks

The subsequent study shows that real multiplex networks exhibit a very different pattern as observed in random multiplex networks. An analysis for a real network from trade economic data is provided. We use the same data set that was used for multiplex modularity optimization in [41]. The data originates from network economy (trade data) using an import-export commodities network between 207 countries in 2011. (see [39]) The data represents the import-export relationships between some countries of the world, disaggregated for different traded commodities. This network can be defined as a multiplex network composed of many layers, where each layer is given by a different commodity group. The nodes are given by 207 countries. A link between two countries in the $i - th$ layer defined as weight exists if there is trade between

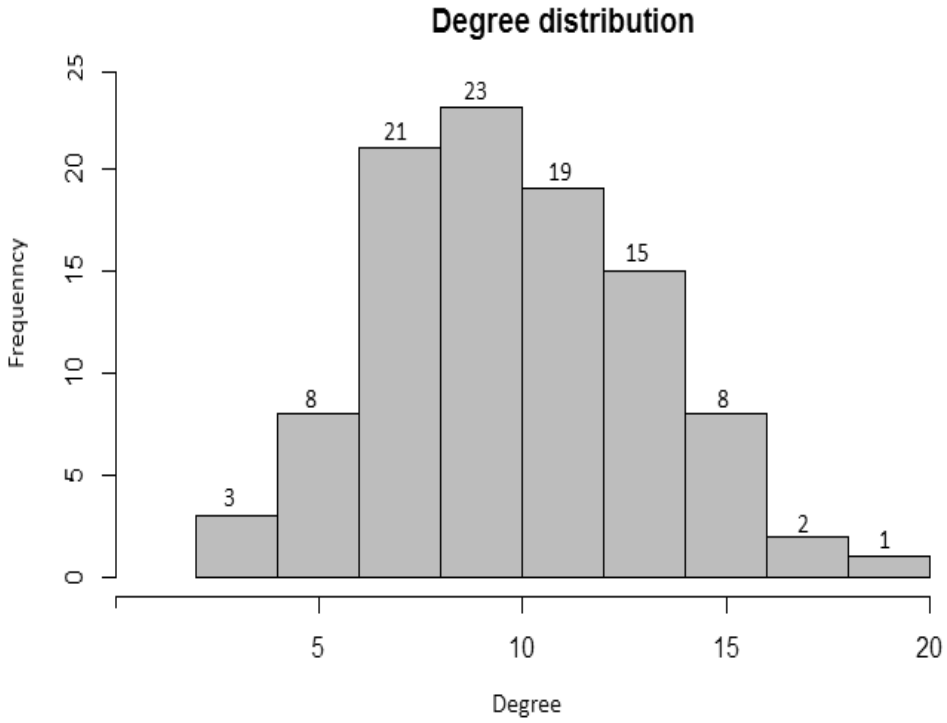


Figure 6.2 Degree distribution of Erdős-Rényi graph model for g_{11} (layer 11).

them in the $i-th$ layer, 6th commodity, for $i \in \{1,...,11\}$, denoting of 11 objective functions f_1 to f_{11}. Data are presented in matrix form: rows and columns represent countries, and the entries of the matrices are the volumes of trade. It is, therefore, a weighted multiplex network. In order to deal with weights, we use the matrix of weights instead of the adjacency matrix, where a weight of zero corresponds to the case the nodes are disconnected and the weights are proportional to the strength of connections. The general classification is based on 96 different commodities. The classification is performed by grouping together similar commodities; this procedure leads to 11 aggregated 'super-commodities'. The data represent the import-export relationships between countries of the world, disaggregated for different traded commodities. We have therefore a multi-layer (multiplex network) composed by many layers, where each layer is given by a different commodity. Each country represents a node of the layer, and a link between two countries in a given layer exists if there is trade between them in that commodity. The data used in the experiment is similar to the data used and

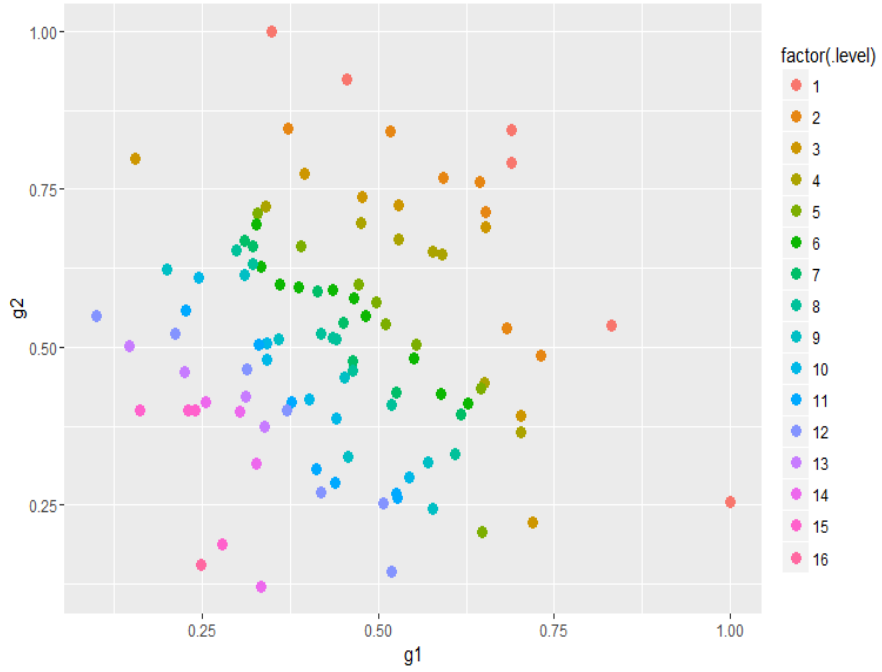


Figure 6.3 Pareto fronts of eigenvector centrality in network layer 1 and layer 2. Non-dominated solutions are shown by points with red colors in the first ranking.

described in the Chapter 5.

Similar to the random network with 11 layers of networks, here the trade multiplex network consist also of 11 layers. The first step is to compute the Pareto front of eigenvector centrality of each layer by full enumeration of 207 node centralities (for each country). As an example, Figure 6.5, shows the Pareto front between objective function f_7 (textiles) and f_{11} (optics and electronics).

Colors of the node centrality on the Pareto Front represent the ranking of the node in the Pareto Front, and the non-dominated solution in the Pareto front is represented by the first ranking from the set of solutions. To give a more clear insight regarding the ranking in the Pareto Front, we can analyze results from Table 6.2. For the non-dominated solution of pairwise optimization of f_7 and f_{11} we can see clearly that there are 5 countries that are non-dominated. They are China, Germany, Italy, England and USA.

From the Figure 6.6 we can see all country rankings in the set of solutions of pairwise optimization.

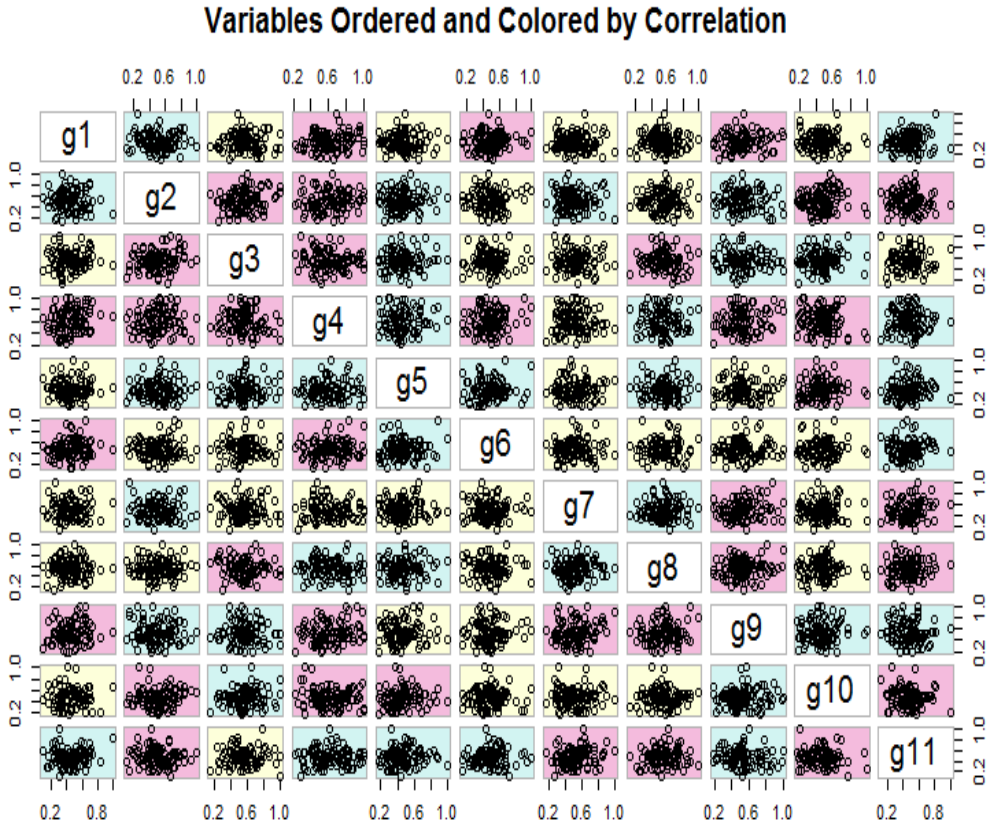


Figure 6.4 Scatter plot matrix for Pareto fronts of Eigenvector centrality with 11 layers of the Erdős-Rényi network model.

It becomes clear that the pairwise optimization for 11 network layers results in a different number of non-dominated solutions, each. There is at least one non-dominated solution and at most 5 non-dominated solutions for the pairwise optimizations. Single non-dominated solutions are USA in f_2 and f_3, France in f_1 and f_4, China in (f_5 and f_6), (f_5 and f_7), and (f_6 and f_7), and UK in f_{10} and f_{11}. There are mostly 2 non-dominated solutions such as China and France in f_4 and f_6, and also there are pairs of objectives with or 3, 4 and 5 non-dominated solutions. The maximal number of 5 non-dominated solutions only occurs in pairwise optimization in f_7 and f_{11}.

Finally, and most importantly, we compute by many-objective optimization the Pareto front for all objective functions (layers) considered together. We present results

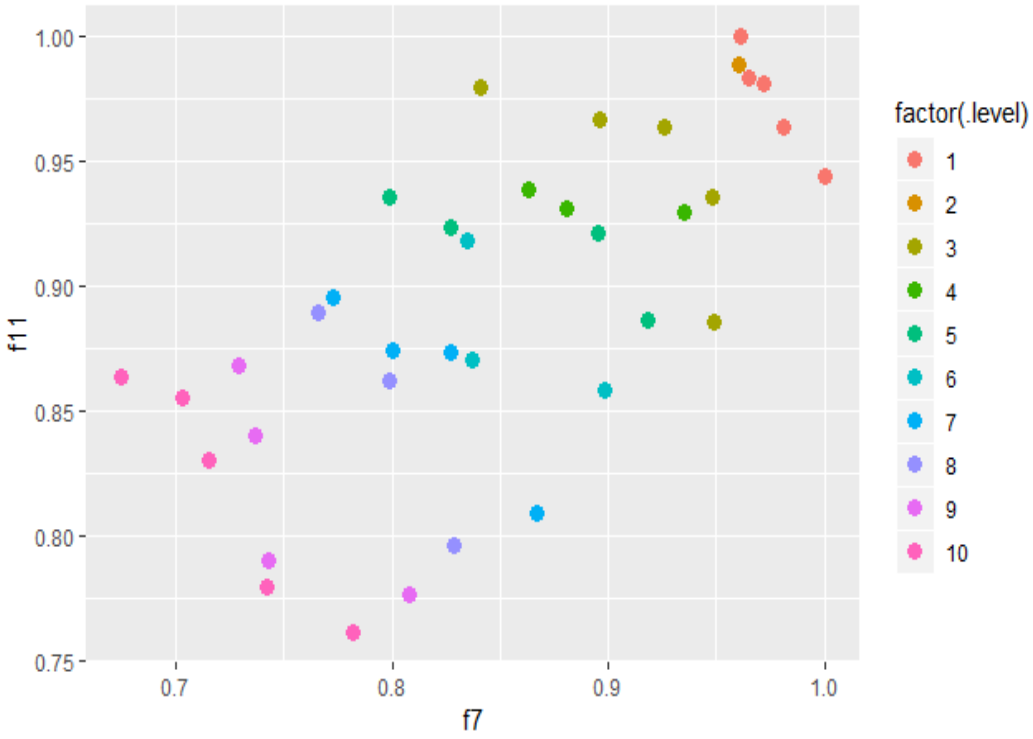


Figure 6.5 Pareto fronts ranking (big 10) of eigenvector centrality in network layer 7 and layer 11 of trade economic data.

in Figure 6.7.

In this figure, from the computation of the Pareto front for all layers together, we found there are 7 countries in the set of non-dominated solutions. They are China, France, Germany, Italy, India, UK, and the USA. Those countries are in the first ranking of the table. The second rank is shared by the countries Belgium-Luxembourg (BLX), Canada, Netherlands, Spain, Switzerland, and Turkey. In total there are 20 ranks and the last three ranks are only single countries Pitcairn Islands (PCN), West Sahara (ESH), Netherlands Antilles (ANT). The result shows that the centrality rank can serve as a rough indicator of the economic power of a country. Note, however, that a particularly strong centrality in one commodity type can be, in principle, the reason for the strong position in the ranking.

Comparing these results by means of pairwise optimization, we find that there are smaller numbers of non-dominated solutions, which is clear because solutions that are

$g1$	$g2$	$Node-label$	$.level$
0.45475356	0.9234941	6	1
0.6897551	0.7932208	19	1
1	0.254959091	43	1
0.6890175	0.8452013	49	1
0.8307032	0.5346712	59	1
0.3492441	1	70	1
0.644231	0.7625479	23	2
0.5920705	0.7692097	24	2
0.51719	0.8429117	13	2
.	.	.	.
.	.	.	.
.	.	.	.
0.3339607	0.1214376	69	14
0.2786245	0.18691	33	15
0.22968245	0.3995689	16	15
0.1624571	0.4001076	26	15
0.2401259	0.3993044	76	15
0.2487949	0.1558329	61	16

Table 6.1 Pareto fronts ranking of Eigenvector centrality in pairwise optimization in layer 1 and layer 2 from Erdős Rényi random graphs

$f7$	$f11$	$Country$	$Node-label$	$.level$
$1.00E+00$	$9.44E-01$	CHN	39	1
$9.65E-01$	$9.83E-01$	DEU	71	1
$9.81E-01$	$9.63E-01$	ITA	90	1
$9.61E-01$	$1.00E+00$	GBR	197	1
$9.72E-01$	$9.81E-01$	USA	199	1
$9.60E-01$	$9.88E-01$	FRA	65	2
$9.26E-01$	$9.63E-01$	BLX	16	3
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
$9.20E-03$	$3.54E-02$	PCN	146	57
$6.77E-03$	$1.51E-02$	ESH	175	58
$2.98E-18$	$2.16E-18$	ANT	125	59

Table 6.2 Pareto fronts ranking of Eigenvector centrality in network layer 7 and layer 11 from trade economic data.

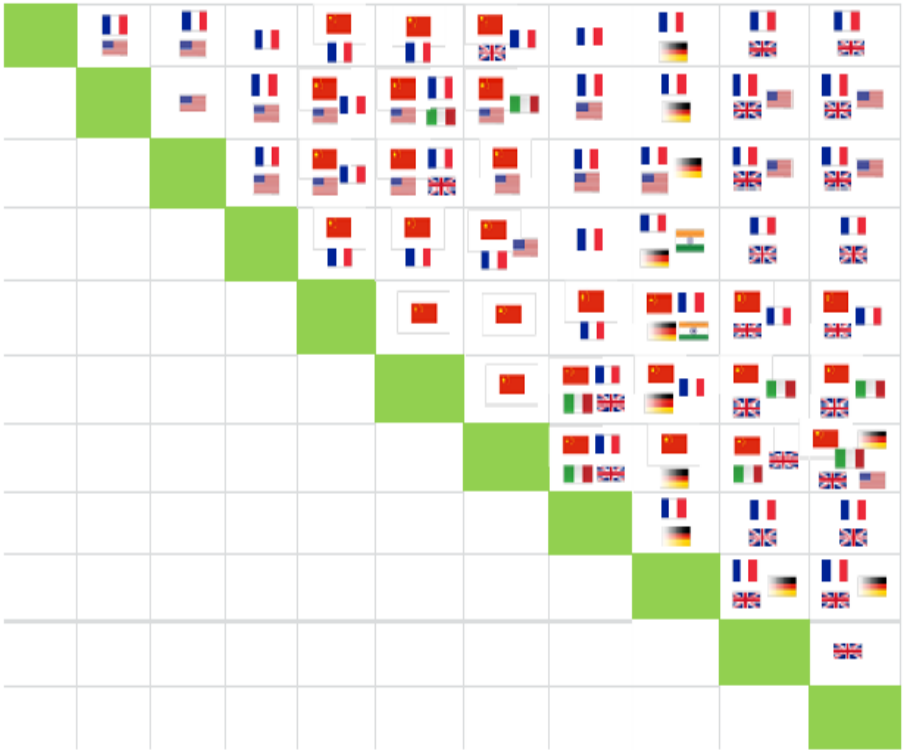


Figure 6.6 Scatter plot matrix for pairwise Pareto fronts of multiplex network trade economic data.Non-dominated solutions are represented by country flags as a vertex of the trade economic network.

non-dominated in only two objectives stay non-dominated when further objectives are added. There are also fewer rankings when all layers are considered. From Table 6.2, we obtain that there are 59 rankings for pairs of objectives (layer 7th and layer 11th), but there are only 20 rankings when optimize all layers together as it is shown in Figure 6.7.

6.5 · Summary

In this chapter, we discussed first results on the computation and analysis of Pareto fronts (set of non-dominated solutions) for eigenvector centrality in multiplex networks for the examples of Erdős Rényi random graphs and economic trade networks. As opposed to the maximization of modularity in previous work [41], the analysis of eigenvector centrality allows for using exact algorithms based on enumeration (all nodes of the networks) and efficient computation of non-dominated sets and dominance ranks

of nodes. We discussed two analysis methods. They reveal different insight into the structure of the dominance relation and the relation between layers. The first method is to compute a Pareto front for every pair of layers. This shows whether similar nodes are central in the two selected layers or whether nodes are positioned very differently. Correlation analysis of the resulting Pareto front matrix can be used to quantify these results. In the analysis of the random graph no significant difference was observed, whereas, in the real-world networks, such as the trade network, the results differ from pair to pair and for some pairs single dominating countries could be identified. Secondly, the non-dominated solutions of the entire network can be computed as well as the dominance rank for all solutions. In the example of the trade multiplex network, the dominance rank is a rough indicator of how important a node is in the global trade network across different commodities. Analysis of the first ranks and last ranks of the networks yield plausible results with respect to this. The total number of non-dominated countries across all 11 commodity groups is, however, relatively small and consists of only 7 countries, all of them in the G20 countries (and 5 of them in the G8)[1]. For all other countries, there exist countries that are better or equal in all commodity centrality values and at least better in one centrality.

[1]India and China are non-dominated but not in the G8.

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	Label	level
0.968	0.959	0.973	0.984	1.000	1.000	1.000	0.934	0.997	0.976	0.944	'CHN'	1
1.000	0.997	0.999	1.000	0.991	0.992	0.960	1.000	0.998	0.999	0.988	'FRA'	1
0.974	0.977	0.977	0.985	0.963	0.979	0.965	0.959	1.000	0.980	0.983	'DEU'	1
0.943	0.970	0.963	0.969	0.956	0.995	0.981	0.964	0.990	0.983	0.963	'ITA'	1
0.831	0.952	0.928	0.995	0.985	0.979	0.948	0.917	1.000	0.965	0.935	'IND'	1
0.932	0.959	0.973	0.977	0.956	0.994	0.961	0.980	0.995	1.000	1.000	'GBR'	1
0.989	1.000	1.000	0.994	0.977	0.986	0.972	0.998	0.993	0.973	0.981	'USA'	1
0.927	0.957	0.964	0.957	0.945	0.953	0.926	0.951	0.978	0.979	0.963	'BLX'	2
0.936	0.957	0.945	0.948	0.915	0.902	0.863	0.962	0.941	0.958	0.938	'CAN'	2
0.946	0.980	0.992	0.968	0.959	0.953	0.896	0.953	0.987	0.996	0.966	'NLD'	2
0.949	0.943	0.952	0.964	0.947	0.953	0.935	0.954	0.978	0.967	0.929	'ESP'	2
0.855	0.877	0.949	0.961	0.920	0.958	0.841	0.953	0.957	0.967	0.979	'CHE'	2
0.797	0.915	0.929	0.930	0.912	0.951	0.949	0.924	0.923	0.918	0.885	'TUR'	2
0.868	0.879	0.920	0.928	0.906	0.921	0.799	0.920	0.942	0.955	0.935	'AUS'	3
0.803	0.846	0.917	0.857	0.891	0.921	0.835	0.917	0.923	0.917	0.918	'AUT'	3
0.760	0.779	0.842	0.845	0.871	0.866	0.837	0.856	0.919	0.941	0.870	'CZE'	3
0.861	0.887	0.881	0.872	0.881	0.899	0.773	0.871	0.890	0.943	0.895	'DNK'	3
0.725	0.803	0.863	0.897	0.908	0.916	0.898	0.879	0.961	0.920	0.858	'IDN'	3
0.831	0.869	0.881	0.924	0.886	0.925	0.881	0.916	0.959	0.935	0.931	'JPN'	3
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
0.029	0.028	0.051	0.055	0.056	0.054	0.039	0.044	0.053	0.084	0.064	'WLF'	17
0.024	0.019	0.008	0.018	0.034	0.017	0.009	0.016	0.048	0.070	0.035	'PCN'	18
0.006	0.000	0.000	0.005	0.005	0.000	0.007	0.012	0.016	0.036	0.015	'ESH'	19
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	'ANT'	20

Figure 6.7 Pareto fronts of Eigenvector centrality 11 layers of the trade economic network. Non-dominated solutions are represented by ranking ("level") one in the the list.

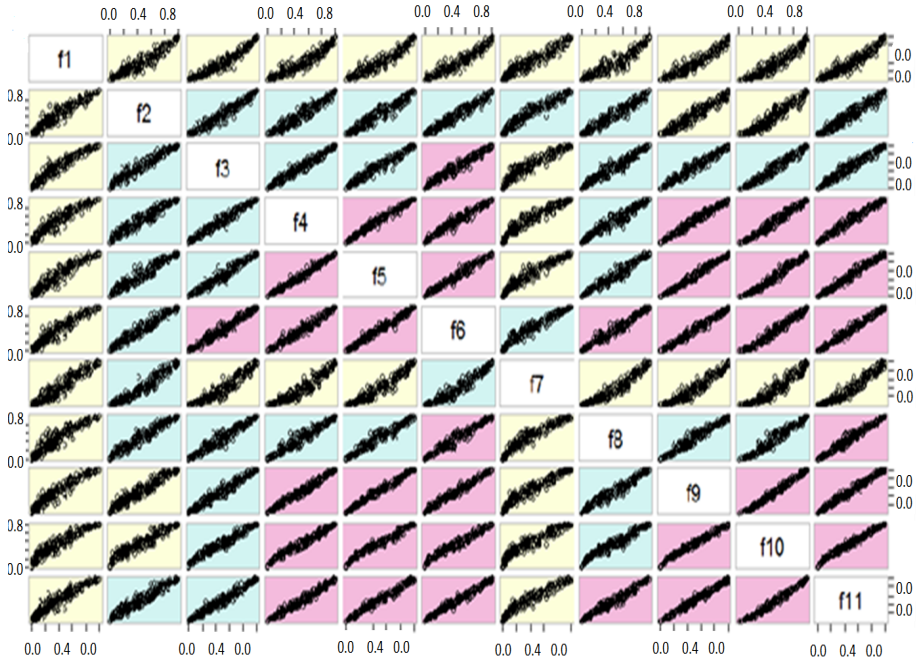


Figure 6.8 Scatter plot matrix of network correlations for the economic network.

7

Immunization of Networks Using Genetic Algorithms and Multi-Objective Metaheuristics

7.1 • Introduction

The study of networks has received increased attention in recent years. The effective control and combating of epidemics, such as Ebola [47] or the Zika virus [37], is one major problem, where the discovery of algorithms for analyzing and controlling networks can make an impact.

This chapter will focus on immunization strategies that achieve a high *eigenvalue drop*. The eigenvalue drop is the drop of the maximum eigenvalue after removal of a subset of nodes from a network, represented as an adjacency matrix. The eigenvalue drop is an effective measure for the impact of an immunization strategy because the maximum eigenvalue is inversely proportional to the epidemic threshold which determines how fast a virus spreads in the network and how long it lingers in the network [14, 15].

The epidemiological model that is considered in this work is the susceptible-infected-susceptible model, in short, SIS model. Here a node in the network can be infected via a direct neighbor and after a time it can recover and is susceptible again. See Figure 7.2 for different epidemiological models. Immunization of nodes can be enforced by measures outside of the network, e.g., by controlling the node or by removing the node from the network. In this work we assume that an immunized node can no longer infect other nodes, nor can it get infected itself.

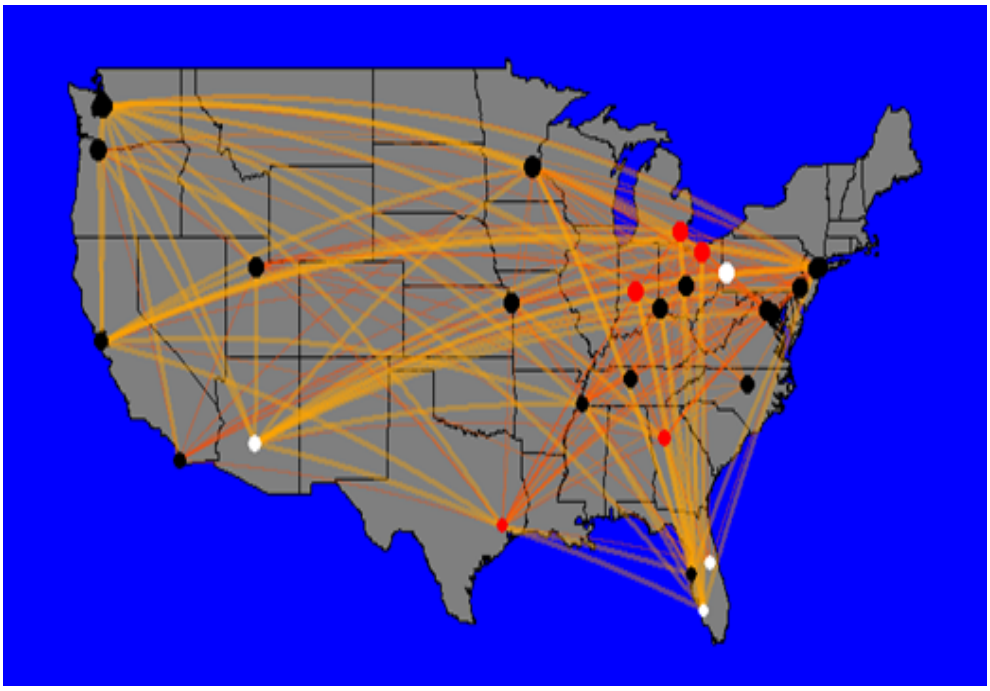


Figure 7.1 US flight network of major airports. The picture shows a snapshot of the spreading of a virus. The black nodes are susceptible, the white nodes are infected, and the red nodes are immunized. `kateto.net/network-visualization`

Consider for instance a network of airports connected by flights, such as the one provided in the US flights dataset shown in Figure 7.1. There might be some nodes already infected and we need to make it difficult for the virus to spread by controlling some major airports, e. g., by special bio-security checks or quarantining.

A network G will be represented as a pair (V, E) where V is a set of nodes $V = \{v_1, \ldots, v_n\}$ and a set of edges $E \subseteq V \times V$. Vertices and edges can have weights and edge weights will be represented by a function $w_E : E \to \mathbb{R}^+$, and node weights by a function $w_V : V \to \mathbb{R}_0^+$. Given this, for a network we can alternatively us the adjacency matrix representation $A(V, E, w_E) \in \mathbb{R}^{n \times n}$ with $a_{ij} = 0$ if $(v_i, v_j) \neq E$ and $w((v_i, v_j))$ otherwise. The first or maximum eigenvalue of the graph will be denoted λ and the corresponding eigenvector with u. The components of this eigenvector, $u_1, ..., u_n$, play a special role in this work and will be called the *eigen-scores* of the matrix.

Definition 7.1 *Given a network G and a network G', where G' is a subgraph of G with some nodes and their adjacent edges removed, the eigenvalue drop $\Delta\lambda$ is defined as*

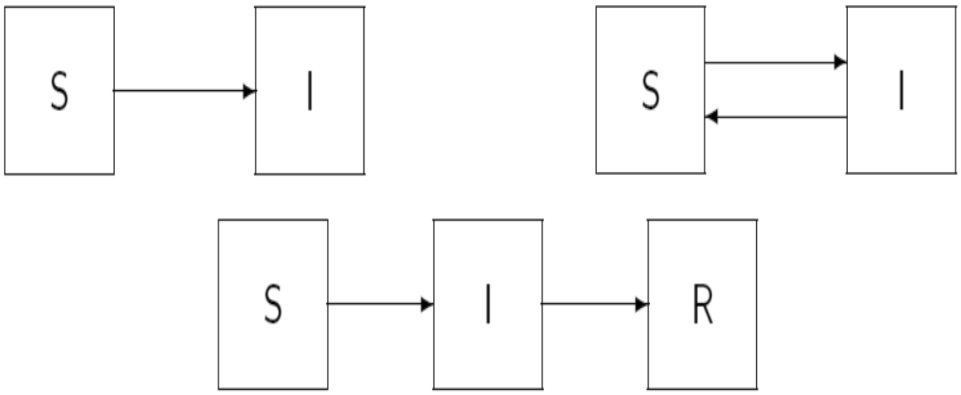


Figure 7.2 Three common models in epidemiology. In the SI model, nodes stay infected, once they got infected. In the SIS model, infected nodes can return into a susceptible state, and in the SIR model nodes are immunized after having recovered and can no longer infect neighboring nodes.

the difference between the maximum eigenvalue of the adjacency matrix of G and the maximum eigenvalue of the adjacency matrix of G′.

Definition 7.2 *The K-Node Immunization problem is the problem of finding a subset of k nodes to be removed from a network with n nodes, such that the eigenvalue drop is maximum.*

It has been shown in [15] that the decision problem that corresponds to the K-Node Immunization problem is NP-complete, and consequently the K-Node Immunization problem is NP-hard. Therefore, heuristic methods have been suggested in [15], most notably the Netshield Plus algorithm. This algorithm does not directly operate on the eigenvalue drop, but uses an approximation of it which is submodular and therefore lends itself to constructing an approximation algorithm. In brief, netshield seeks to maximize the following *Shield value* (Sv) function, which is closely correlated with the eigenvalue drop.

In this thesis, we propose an alternative approach to the k-node immunization problem based on genetic algorithms (Section 7.3.1) and compare results to Netshield Plus (Section 7.3.2). In the problem specific mutation operator, some of the ideas of Netshield will be adopted. Therefore, we will introduce this algorithm and the scoring function used by it briefly in Section 7.2. Moreover, a multi-objective generalization of

the k-node immunization problem is discussed. It introduces a cost term as a second objective (Section 7.4.1). First results on finding the Pareto front of this problem with multi-objective metaheuristics are presented in Section 7.4.2.

7.2 · Netshield Algorithm

Next, we will briefly introduce the Netshield algorithm. Some of the ideas of this algorithm will be useful in the design of the problem specific genetic algorithm. Moreover, the Netshield Plus algorithm, an improved version of the Netshield algorithm, will serve as a baseline algorithm in the benchmarking.

Let $G = (V, E)$ denote the original graph, and $G = (V', E')$ the graph after some nodes have been removed, and we define $S = V \setminus V'$. Moreover, A and A' denote the corresponding adjacency matrices. Then the Shield value (Sv) of S is defined as follows.

$$Sv(S) = \sum_{i \in S} 2\lambda(u_i)^2 - \sum_{i,j \in S} a_{ij} u_i u_j$$

Here, λ denotes the largest eigenvalue, u_i denotes the i-th component of the eigenvector that corresponds to the largest eigenvalue. It is also called the i-th eigen-score. The Shield value rewards dissimilarity between nodes, that is small a_{ij}, and a high eigen-score.

As opposed to the Netshield algorithm, the Netshield Plus algorithm [15] removes nodes in batches of b nodes each. After each batch, the largest eigenvalue and the corresponding eigen-scores are recomputed. This way the algorithm yields more accurate results, but due to multiple eigenvalue computations the computation time increases. Netshield Plus is therefore especially recommended for small or moderate size networks, as we discuss them in this chapter.

7.3 · Problem Specific Genetic Algorithm

7.3.1 · Discussion of the method

In this work we use a standard $(\mu + \mu)$ genetic algorithm (see, e.g., [60]) with scaled proportional selection (mating selection) and truncation selection (environmental selection). The genetic algorithm for the k subset selection problem uses problem specific mutation and crossover operators. The representation of solution candidates

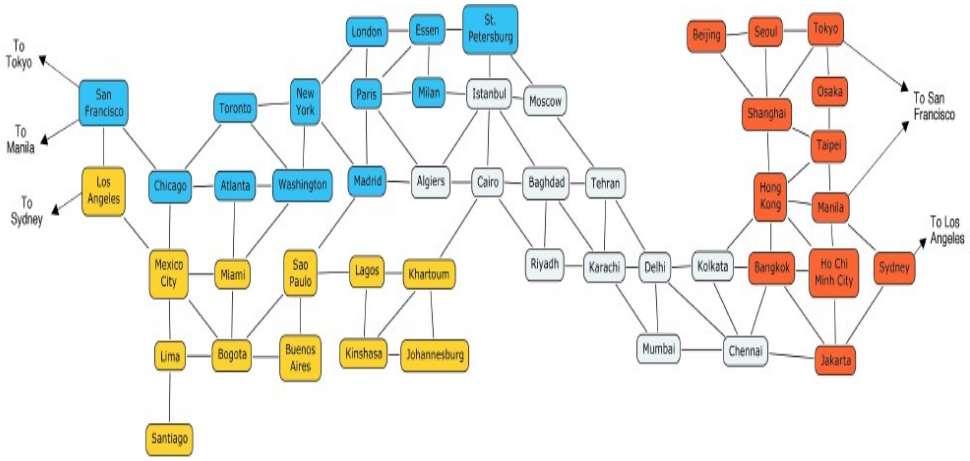


Figure 7.3 Network in the Pandemic game board
(from: http://jhkimrpg.livejournal.com/78787.html).

is not binary, as usual, but a problem specific representation for subset selection as it has been used in other contexts, too [61]. A solution is represented as k non duplicate integers in $[1,n] \subset \mathbb{N}$.

The mutation operator that was designed for this problem relies on two mechanisms:

- Firstly, in each mutation, an integer that is in the array is replaced by an integer in $[1,n]$ that is not in the array.
- Secondly, the algorithm works with two different mutation rates. For nodes with a top-k eigen-score, the probability of mutation is increased by a constant factor ≥ 0, making it more likely to be selected for the set or discarded. This way it is hypothesized that the algorithm spends more time in exploring relevant parts of the graph. The multiplication factor will be denoted with ν.

Mutation is applied to each offspring individual. First, an integer in the array is selected proportionally to the mutation probabilities. Then an integer outside the array is selected proportional to the mutation probabilities. And then the node inside the array is replaced by the node outside the array. The genetic algorithm does not feature crossover, but we might consider the development of a problem specific crossover for future research.

7.3.2 · Comparison to Netshield Plus

For the empirical comparison of algorithms we will use five data sets on networks:

- Karate: A social network of friendships between 34 members of a karate club at a US university in the 1970s [62].
- Dolphins: It is a social network consisting of an undirected network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand. [38]
- US Flights: This is a list of the most important Airports in the United States connected to other based on the exist are of connecting flights (edge) from one airport to the other airports.
- Pandemic: A cooperative board game with the goal to fight the outbreak of the virus. We used the graph that connects cities in the world as an example data set [35]. A picture of the Pandemic board is shown in Figure 7.3
- Conference Day 1: The social interactions of members of a conference on the first day. Taken from http://www.sociopatterns.org/datasets/infectious-sociopatterns.
- Conference Day 3: From the same data set as above, but for the third day.

The data sets US flights and Pandemic are most representative for the problem class. The other networks are added to gain more general insights into the algorithm behavior and reliability. Note that social interaction networks are also relevant in the spread of the virus, albeit control is less straightforward as compared to networks where nodes are assigned to places, such as US flights and the Pandemic board game network.

For the k-node immunization problem we used the Netshield Plus algorithm and parameters as described in [15]. For the genetic algorithm tests the following setting was applied: The number of function evaluations was 30000. Different mutation parameters were tested, with a value of $\nu \in \{1/n, 2/n, 3/n, 6/n, 1\}$, that is the mutation rate for the k components of u with the highest eigen-score. For all other nodes, the mutation probability was set to $1/n$, which is a recommended rate according to Bäck [4].

Regarding the single objective genetic algorithms, they were executed 20 times each, for $k = 3, 5$ and 10 on the Karate, Dolphins, US Flights, Pandemic, Conference Day 1 and Conference Day 3 networks. Table 7.1 shows results for single objective optimization of the eigenvalue drop. For assessing statistical significance we also provide box plots of our results in Figure 7.5, 7.6, 7.7 and Figure 7.6. We observe that GA_5, which represents the $(\mu + \mu)$ genetic algorithm that introduces a mass of

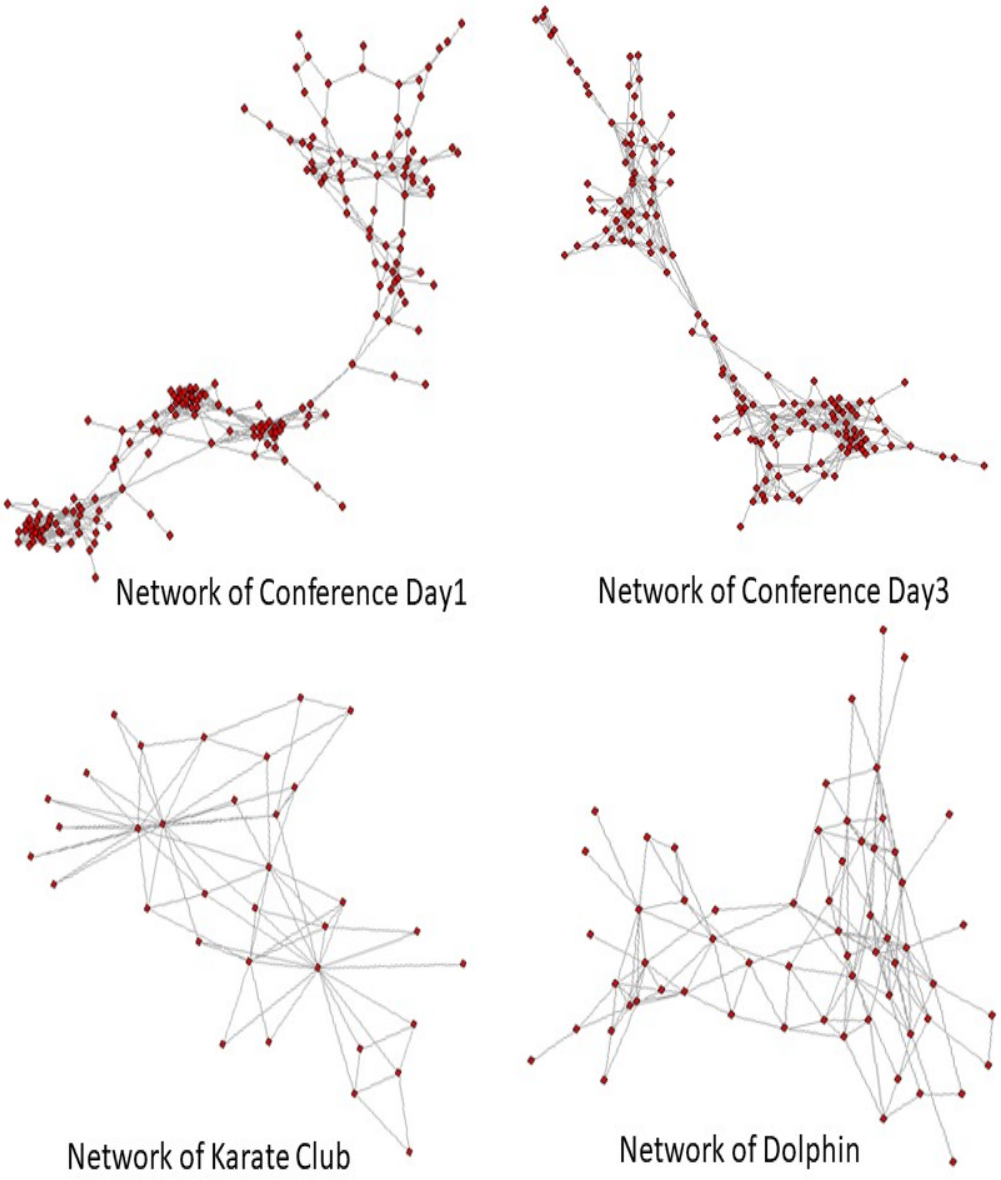


Figure 7.4 Depiction of network that we consider for our experiments consist of network of Conference Day1, Conference Day3, Karate Club and network of Dolphin

	Network	*GA_1*	*GA_2*	*GA_3*	*GA_4*	*GA_5*	*Netshield+*
$K = 5$	*karate*	4.1068	4.1068	4.1068	4.1068	4.1068	4.1068
	Dolphins	2.0812	2.0769	2.0807	2.0978	2.0978	2.0817
	USA	7.2043	7.2043	7.2043	7.2043	7.2043	7.2043
	Pandemic	0.9243	0.9419	0.9502	0.9502	0.9133	0.9556
	$Conf.day1$	3.0109	2.9583	3.0289	3.0455	3.0391	3.0638
	$Conf.day3$	17.670	17.671	17.669	17.669	17.610	3.8542
$K = 10$	*karate*	5.1077	5.1077	5.1077	5.1077	5.3115	5.3115
	Dolphins	2.9077	2.9230	2.9685	3.1575	3.2862	3.3997
	USA	11.690	11.809	11.922	12.177	12.608	12.608
	Pandemic	1.4201	1.4299	1.4490	1.5114	1.5215	1.4442
	$Conf.day1$	4.3853	4.3831	4.4207	4.6697	19.237	4.9121
	$Conf.day3$	17.659	17.664	17.658	17.659	17.658	5.6483

Table 7.1 Results of genetic algorithm and Netshield Plus comparisons.

5 to the k-highest eigen-score nodes, to be the best candidate. Although there is not a unanimously best genetic algorithm for the task, we consider our genetic algorithms to be a supplementary tool to Netshield/Netshield Plus, for medium-sized networks (≤ 200 nodes).

7.4 · Multi-Objective Node Immunization

In real-world scenarios, it is likely that multiple nodes need to be controlled or immunized, but it is typically not the case that the value of k is given a priori. Rather it is the case that the immunization of a node comes with a cost, which can differ from node to node. If a larger number of nodes is immunized the total cost would be approximately proportional to the cumulated cost of immunizing the single nodes. Let S denote the set of indexes of the immunized nodes and c_i denote the cost of immunization of node i, defined a priori. Then the *immunization cost* objective function can be defined as

$$C(S) = \sum_{i \in S} c_i \rightarrow \min$$

In multi-objective optimization, problems with two or more objectives are solved. In case of the node immunization problem the problem formulation reads as follows:

ID	*City*	*Population*	*ID*	*City*	*Population*
1	*SanFrancisco*	723724	25	*Beijing*	7602069
2	*Chicago*	2830144	26	*Seoul*	9860000
3	*Montreal*	3280123	27	*Tokyo*	8372440
4	*NewYork*	8124427	28	*Shanghai*	15017783
5	*Washington*	548359	29	*HongKong*	7347000
6	*Atlanta*	424096	30	*Taipei*	2491662
7	*Madrid*	3146804	31	*Osaka*	2590815
8	*London*	7489022	32	*Bangkok*	4935988
9	*Paris*	2141839	33	*HoChiMinhCity*	3496586
10	*Essen*	596204	34	*Manila*	10546511
11	*Milan*	1316218	35	*Jakarta*	8556798
12	*St.Petersburg*	4991000	36	*Sydney*	4444513
13	*Algiers*	2029936	37	*Khartoum*	2090001
14	*Istanbul*	10034830	38	*Johannesburg*	2091491
15	*Moscow*	10472629	39	*Kinshasa*	9464000
16	*Cairo*	7836243	40	*Lagos*	9020089
17	*Baghdad*	5753612	41	*SaoPaulo*	10059502
18	*Tehran*	7160094	42	*BuenosAires*	11595183
19	*Delhi*	11215130	43	*Santiago*	4893495
20	*Karachi*	11969284	44	*Lima*	7857121
21	*Riyadh*	4328067	45	*Bogota*	7235084
22	*Mumbai*	18410000	46	*MexicoCity*	8659409
23	*Chennai*	7088000	47	*LosAngeles*	3911500
24	*Kolkata*	4497000	48	*Miami*	386740

Table 7.2 Cost values for Pandemic network (proportional to city size).

Label	*Airport*	*Visits*
42	*Cincinnati/northernKentucky*	117
51	*DetroitMetropolitanWayneCounty*	126
71	*GeorgeBushIntercontinental*	90
81	*Hartsfield − jacksonAtlantaInternational*	102
85	*HopkinsInternational*	123
88	*IndianapolisInternational*	120
106	*KansasCityInternationalAirport*	117
110	*LaGuardia*	123
131	*MemphisInternational*	105
137	*Minneapolis − St.PaulIntl*	135
153	*NashvilleInternational*	108
155	*NewarkLibertyInternational*	123
164	*OrlandoInternational*	84
169	*PhiladelphiaInternational*	120
172	*PittsburghInternational*	120
173	*PortColumbusIntl*	120
174	*PortlandInternational*	138
177	*Raleigh − durhamInternationalAirport*	108
190	*RonaldReaganWashingtonNationalAirport*	117
193	*SaltLakeCityInternational*	123
195	*SanDiegoInternationalAirport*	99
196	*SanFranciscoInternational*	114
201	*Seattle − TacomaInternational*	141
204	*SkyHarborIntl*	99
206	*SouthwestFloridaReg*	81
214	*TampaInternational*	84
224	*WashingtonDullesInternational*	117

Table 7.3 Cost values for Pandemic network (proportional to city size).

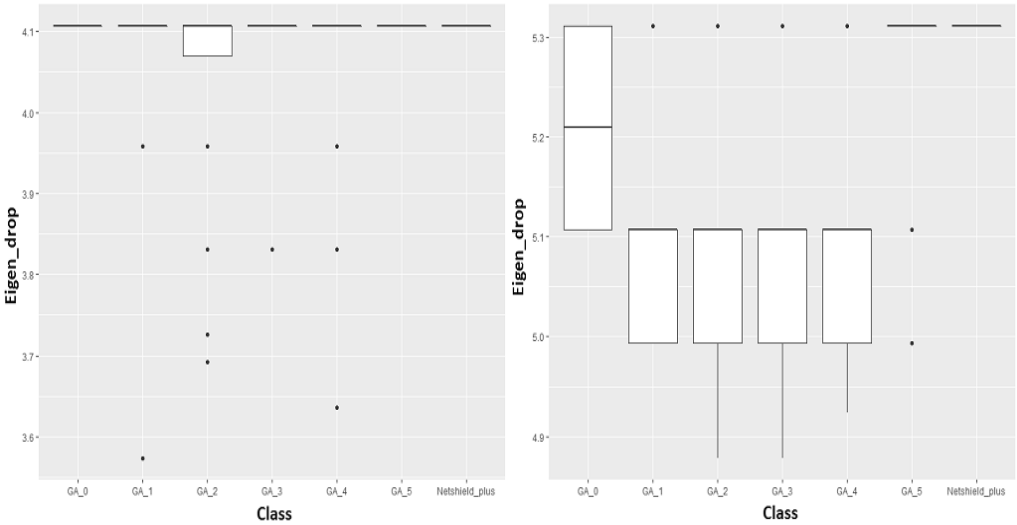


Figure 7.5 Results of genetic algorithm and Netshield Plus comparisons for the Karate network.

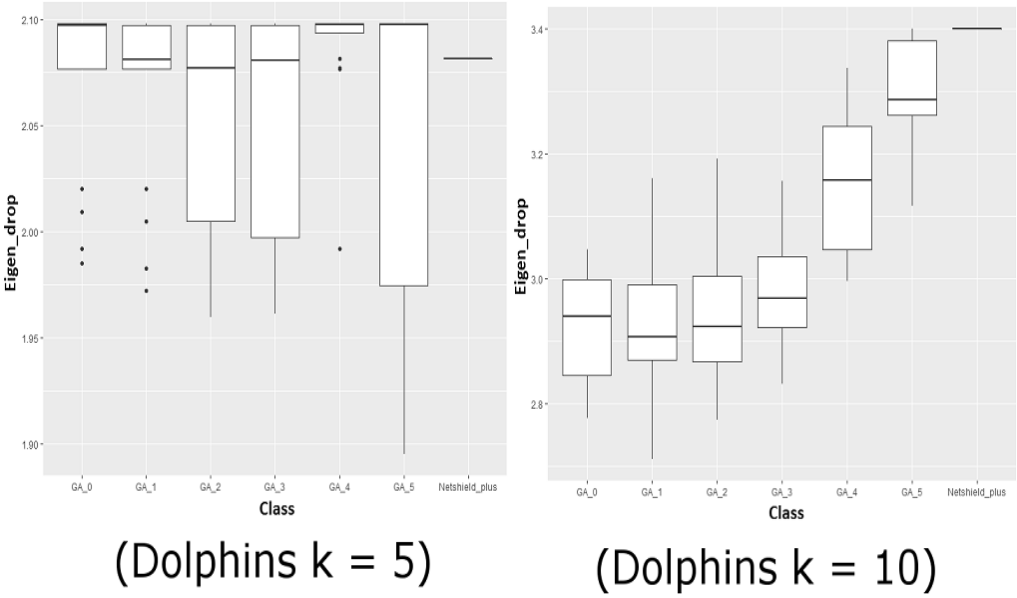


Figure 7.6 Results of genetic algorithm and Netshield Plus comparisons for the Dolphin network.

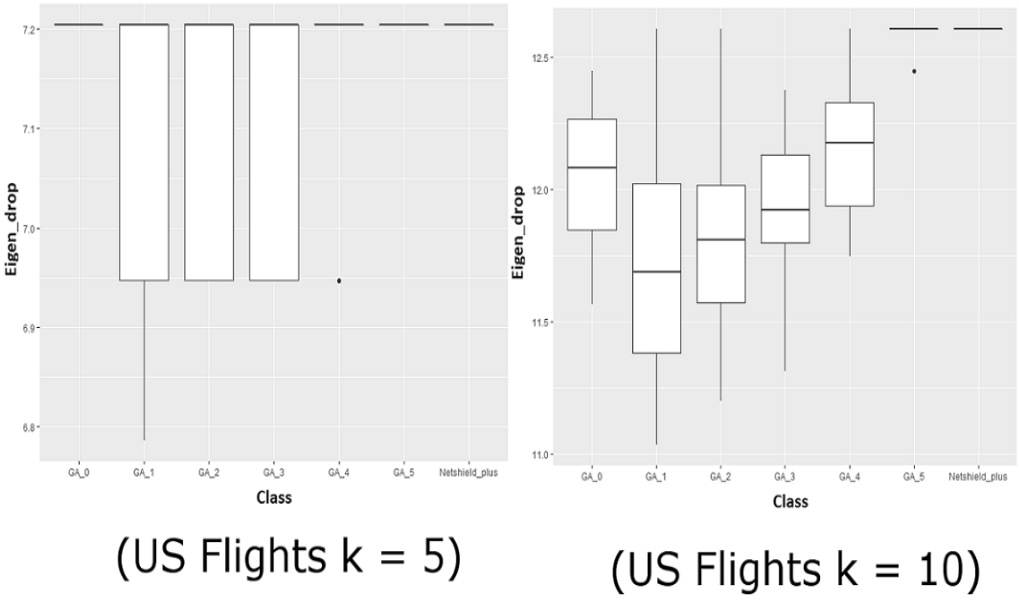


Figure 7.7 Results of genetic algorithm and Netshield Plus comparisons for the US flight network.

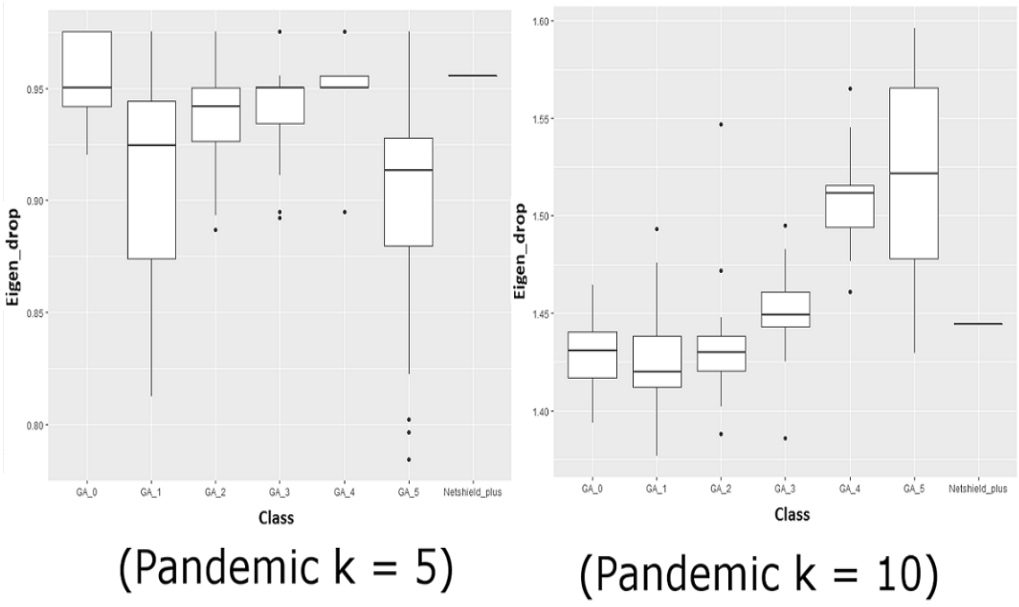


Figure 7.8 Results of genetic algorithm and Netshield Plus comparisons for the Pandemic network.

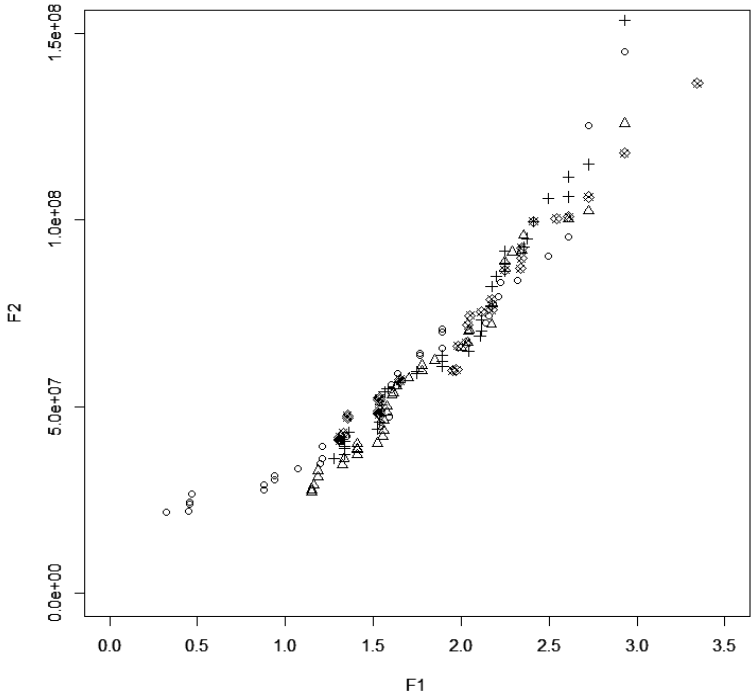


Figure 7.9 Pareto Front for the Pandemic Network found by the NSGA-II algorithm with 5 experiments.

$$f_1(S) = \lambda(S) \rightarrow \max \tag{7.1}$$

$$f_2(S) = C(S) \rightarrow \min \tag{7.2}$$

$$S \subseteq \{1, \ldots, n\} \tag{7.3}$$

We are interested in the efficient set of this problem, that is the set: $\mathcal{S}_E = \{S \in \{1,...,n\} | \nexists S' \subseteq \{1,\ldots,n\} : f_1(S') \geq f_1(S) \wedge f_2(S') < f_2(S) \vee f_1(S') > f_1(S) \wedge f_2(S') \leq f_2(S)\}$ and the Pareto front $\{(f_1(S), f_2(S))^T \mid S \in \mathcal{S}_E\}$.

7.4.1 · Multi-objective Metaheuristics

Two multi-objective evolutionary algorithms (MOEA, or EMOA) are considered as solvers: The first one is the non-dominated sorting genetic algorithm (NSGA-II) [17] and the second one is the S-metric selection algorithm (SMS-EMOA) [8].

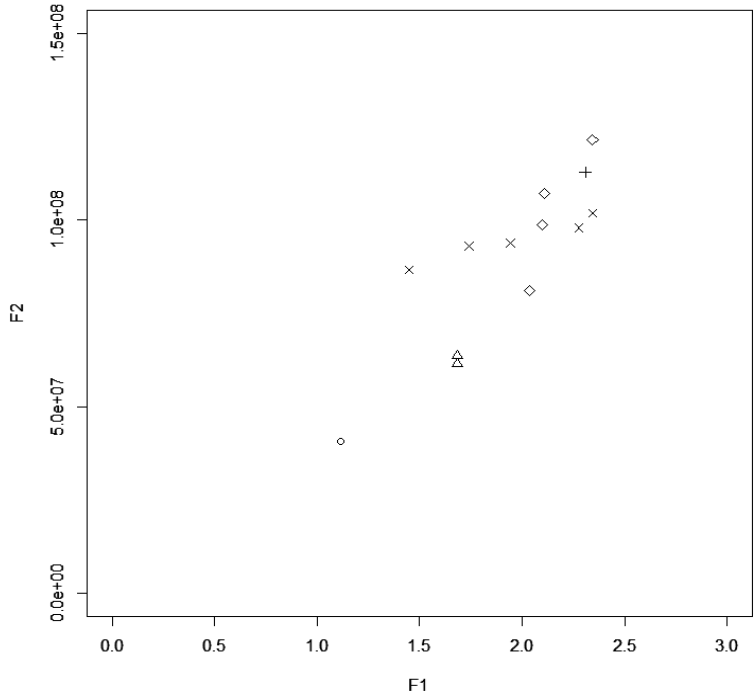


Figure 7.10 Pareto Front for the Pandemic Network found by the SMS-EMOA algorithm with five experiments.

The implementations of SMS-EMOA and NSGA-II in R featured by Bossek's `ecr` package was used in this work. The representation of a subset is chosen to be a bit vector b in $\mathbb{B}^n$, where $b_i = 1$ means the node is selected to be removed/quarantined and $b_i = 0$ means the node is not selected, for $i = 1, \ldots, n$. As recombination operator, one point crossover is used. For all bits, we used $p_m = 1/n$ as the mutation probability. The reason for this mutation rate is that, in contrast to the single objective genetic algorithms we discussed, here we do not know a priori the number of nodes to remove/quarantine. That is, we do not specify a subset cardinality. As a consequence, the algorithm should not try to explore a particular direction of the search space (bias introduced from the mutation operator), but rather present to the decision makers a complete picture of their possible choices. For example, quarantining 10 less-important (in terms of eigen-score) airports could be more beneficial in terms of cost, than quarantining 1 important (in terms of eigen-score) airport.

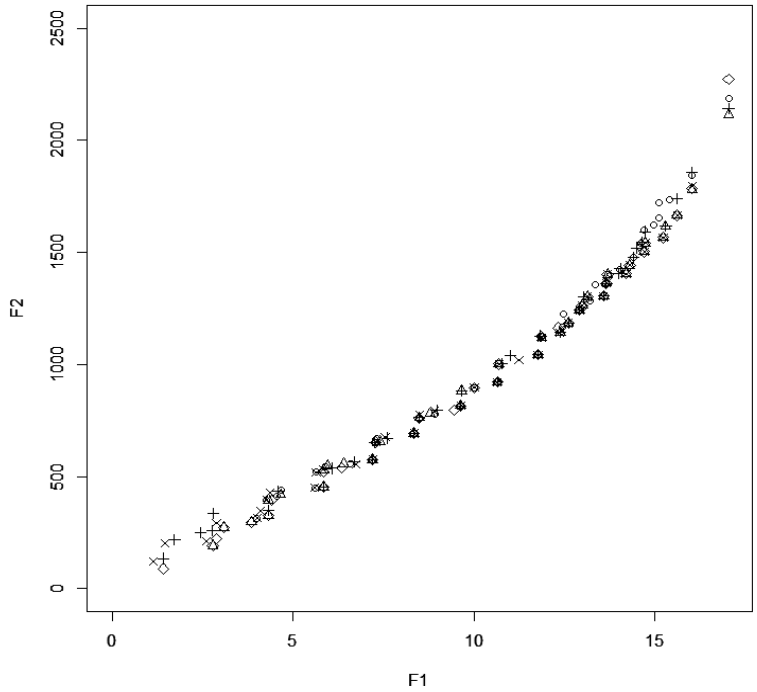


Figure 7.11 Pareto Front for the USA Flight Network found by the NSGA-II algorithm with 5 experiments.

7.4.2 · Empirical Results

The Pandemic and the US flights networks serve as examples for computing the Pareto fronts and efficient sets. In case of the Pandemic network, the size of the cities was used as a cost, assuming that it is more difficult to immunize larger cities. In case of the US flights network, the size of the airport (number of visits) was taken into account. The cost values are tabulated in Table 7.2 (Pandemic) and Table 7.3 (US flight). While we aimed for realistic problem settings, we would like to note that in order to plan effective real-world immunization more modeling is needed, including social interactions, geographic environment, and various other factors. Here, we merely focus on the network aspects of the problem. Each algorithm for the multi-objective optimization was run 5 times, producing 5 Pareto front approximations. Results for the Pandemic Network are shown in Figure 7.9 and Figure 7.10. Results for the Pandemic are shown in Figure 7.11 and Figure 7.12. The Pareto front looks near linear. Overall the

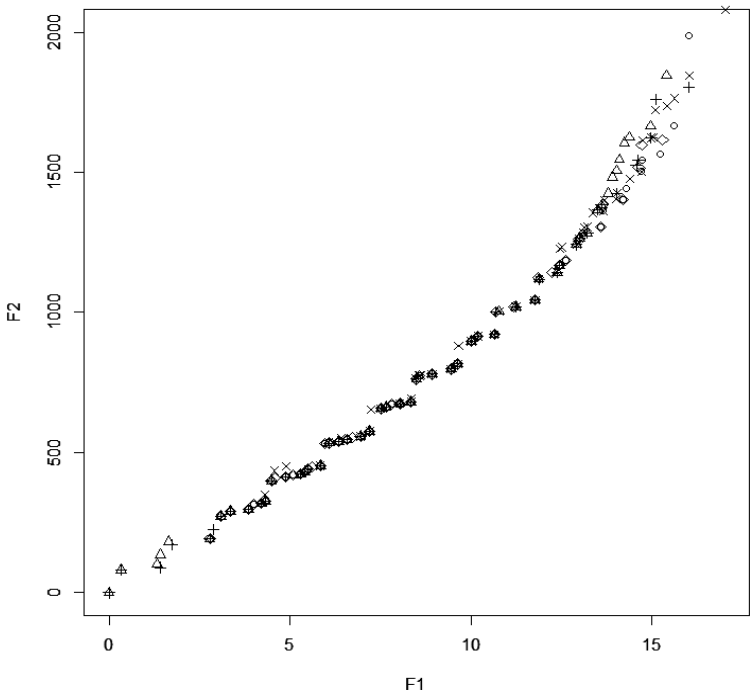


Figure 7.12 Pareto Front for USA Flight Network found by the SMS-EMOA algorithm with 5 experiments.

NSGA-II algorithm obtained better results and displayed a more robust performance than the indicator based SMS-EMOA on this problem. It is also interesting to note that the Pareto front looks near linear, which might be explained by the fact that big nodes (larger cities or, respectively, airports) are at the same time costly and important for immunization. For the US Flights network, a knee region can be identified.

7.5 · Summary

This chapter discuses network immunization techniques based on a heuristic method using genetic algorithms technique. Compared to Netshield Plus, the results show that the genetic algorithm often performs better, sometimes significantly better, in solving the k-node immunization problem. Netshield Plus is a fast heuristic and produces in many cases good results. Based on our findings, a strategy could be recommended that, if time is available, uses not only Netshield Plus but also a problem specific genetic algorithm to make it more likely that the best solution for the edge drop objective is

not overlooked.

In order to achieve good results, specific adaptations turned out to be very useful. An idea that works well is to use eigen-score values in order to adjust the mutation probabilities. This way the search is more focused on the part of the search space that is more likely to be relevant to solving the problem. We should also emphasize here that the supplementary use of a problem specific genetic algorithm has the advantage of calculating the actual eigen-drop, rather than an approximation of it. This can be useful for moderately sized networks. However, in large networks, the computational cost increases, since the algorithm eigen decomposes larger adjacency matrices.

First results were also obtained on a multi-objective formulation of the node immunization problem. We discuss the formulation where the total cost of immunization is one objective and the drop of the eigenvalue is the second objective. Two different meta-heuristics are applied to solve this problem and they widely agree with the results and show robust performance.[1]

[1]We remark that the source code of the algorithms and the network datasets are available by the authors on the research groups web page http://moda.liacs.nl.

8

Conclusions and Outlook

This thesis focuses on multi-objective and many-objective optimization for complex network analysis and vice versa on using a technique of network analysis for the purpose of many-objective optimization. As the result, it has been shown that both two different research topics have resulted in interesting and useful methods for mining and discovering valuable information, and for decision making/analysis as well. The main contributions of this thesis to the research area are:

- Using a network analysis technique (Community Detection) in reducing the complexity of many-objective optimization.
- A novel network analysis technique to study the complexity theory transition in interactive networks.
- Applying many-objective optimization for community detection in analyzing multiplex networks.
- Utilize many-objective optimization for finding a set of key players in multiplex network analysis.
- Utilize a multi-objective meta-heuristic for network immunizations.

In Section 8.1 we will describe these result in more detail, followed by an outlook of the result in section 8.2

8.1 · Conclusions

This thesis combines two different research topics, many-objective optimization, and complex network analysis. Most of the algorithms we applied are metaheuristics, based

on the paradigm of evolutionary multi-criterion optimization. On the algorithm side, we applied NSGA-II (Non-dominated Sort Genetic Algorithm II), MOEA/D (Multi-Objective Evolutionary Algorithm using Decomposition), and SMS-EMOA (S-Metric Selection - Evolutionary Multi-Objective Optimization).

As the thesis consist of two parts, in the first part, Chapters 3 and 4 explain the approach for understanding and reducing the complexity of the optimization problem.

1. In chapter 3, the result showed the workflow called Community Detection for Many-objective Optimization (CoDeMO) was discussed that uses graph-theoretic community detection reveal the structure of many-objective black-box optimization problems. It interprets objective functions as actors in a social network (complimentary), which might be friends, enemies (conflicting) or neutral with respect to each other. The proof of concept study shows that for problems with a relatively simple underlying structure this approach works both to reveal the structure and to exploit it by providing more interpretable and also more accurate optimization results. The community detection works well for many-objective optimization and was tested with up to 50 objective functions. In addition, we pointed to some limitations of the approach. We assume that in many cases the result of the community structure reveals whether or not a decomposition is possible. By pointing at the possibility of higher order interactions we also show that in some non-linear problems, apparently simple structured problems can have complex interactions.

2. Chapter 4 shows for NK Landscapes, that community structure that is detected for the 'correlation graph' does not correspond with the community structure of the epistatic link network which has many components for small values of k and only one big component for $k = N - 1$ (every gene is linked to every other gene). Instead, the correlation network has the lowest number of components for $k = 2$. For lower and higher values the number of communities clearly grows. As the critical transition from polynomial time, solvable maximization problems to NP-complete maximization problems appears at the transition from $k = 1$ to $k = 2$ (for random networks) we suspect that these findings might be not coincidental. We show also that the average squared correlation reaches a sharp peak near this value of k. This peak is less pronounced for adjacent epistatic genes which do not undergo a critical transition but a gradual transition in terms

of complexity. So far we have only studied the case $N = 10$ and studies on larger networks are required in the future to improve the generality of the findings. A problem that needs to be solved for such studies is how to tame the 'explosion' in the size of the random number tables needed to generate the NK-landscapes. A useful proposal has been made by Altenberg [2], who suggested to re-generate the random numbers on-the-fly when needed and provided a function that can be used for this.

The second part that consists of chapters 5, 6, and 7 deals with the analysis of complex networks using multi- and many-objective optimization. The main results are the following,

1. Chapter 5 showed how to apply many-objective optimization for the analysis of multiplex networks. The results are analyzed using three tools suggested here: Correlation heatmap, the community of objectives analysis, and the Pareto-front plot matrix. These were computed for an economic trade network with 11 groups of commodities. Clearly, a grouping emerges in terms of complementarity and/or in terms of indifference. NSGA-II, SMS-EMOA, and single-objective genetic algorithms can be used as a search engine.

2. Chapter 6 discussed the results of the computation and analysis of Pareto fronts (set of non-dominated solutions) for eigenvector centrality in multiplex networks for the examples of Erdős Rényi random graphs and economic trade networks. As opposed to the maximization of modularity in previous work [41], the analysis of eigenvector centrality allows for using exact algorithms based on enumeration (all nodes of the networks) and efficient computation of non-dominated sets and dominance ranks of nodes. The experiment using trade network data for 11 groups of commodities between countries around the world, and the analysis of the first ranks and last ranks of the networks yield plausible results with respect to this. The total number of non-dominated countries across all 11 commodity groups is however relatively small and consists of only 7 countries out of 207 countries, all of them in the G20 countries and 5 of them in the G8.

3. The last chapter of the second part concerns network immunization by solving the k-node immunization problem. We formulate the node immunization problem as a multi-objective problem. The first objective is to maximize the eigenvalue

drop and the second objective is the cost of immunization itself. The eigenvalue drop is the drop of the maximum eigenvalue after removal of a subset of nodes from a network, represented as an adjacency matrix. First results are presented on biobjective optimization using multi-objective genetic algorithms as solver. We emphasize here that the supplementary use of a problem specific genetic algorithm has the advantage of calculating the actual eigen-drop, rather than an approximation of it.

8.1.1 · Outlook

There is a lot of interesting future work related to the topic of the thesis out there, but we list some specific future work based on our findings as follows:

1. To extend the proof of concept results by additional benchmarking, more in-depth analysis of an extended benchmark will be conducted. For reducing the complexity of many-objective optimization, it will become a significant result if it is applied to a real-world problem.

2. The results on NK landscapes focused on the study of interactions between traits and a better understanding of complexity transitions. In future work, NK landscapes with different community structure could provide an interesting test problem for many-objective optimization and complexity reduction techniques, such as those suggested in Chapter 3. The different context of multi-objective and many-objective optimization [27], the problem of maximizing the components of an NK landscape could yield an interesting test case for many-objective optimization with a tunable degree of correlation between the objective functions. To this end, first results on how to exploit community structure for more effective maximization have recently been shown on a different optimization problem.

3. Extending the analysis to more networks within larger networks could be tested. Moreover, application on a dynamic network also will be very promising.

4. Related to network centrality, using different measures of centrality and also revealing the trade-off between them would be an interesting extension of our work.

5. Based on our experience, an application to network immunization would require further adaptations to the Genetic Algorithm. We believe that a promising path to

accomplish this is to further hybridize the GA with Netshield Plus, for instance by using the latter in constructing initial solutions. Moreover, the development of problem-specific crossover operators could be beneficial.

Bibliography

[1] Pieter Adriaans. Facticity as the amount of self-descriptive information in a data set. arXiv preprint arXiv:1203.2245, 2012.
Cited on page 44.

[2] Lee Altenberg. NK landscapes. In Z. Michalewicz T. Bäck, D Fogel, editor, Handbook of Evolutionary Computation. Oxford University Press, 1997.
Cited on pages 39, 47, and 103.

[3] Philip Anderson. Perspective: Complexity theory and organization science. Organization science, 10(3):216–232, 1999.
Cited on page 37.

[4] Thomas Bäck. Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford University Press, 1996.
Cited on page 88.

[5] Johannes Bader and Eckart Zitzler. Hype: An algorithm for fast hypervolume-based many-objective optimization. Evolutionary Computation, 19(1):45–76, 2011.
Cited on page 21.

[6] Marc. Barthelemy. Betweenness centrality in large complex networks. The European Physical Journal B-Condensed Matter, 38(2):163–168, 2004.
Cited on page 12.

[7] E. T. Bell. Exponential numbers. The American Mathematical Monthly, 41(7):411–419, 1934.
Cited on page 58.

[8] Nicola Beume, Boris Naujoks, and Michael Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research, 181(3):1653–1669, 2007.
Cited on pages 21, 57, 70, and 95.

[9] Boccaletti, Stefano, Ginestra Bianconi, Regino Criado, Charo I. Del Genio, Gómez-Gardenes J, Miguel Romance, Irene Sendina-Nadal, Zhen Wang, and Massimiliano Zanin. The structure and dynamics of multilayer networks. Physics Reports, 544(1):1–122, 2014.
Cited on page 70.

[10] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. Journal of Mathematical Sociology, 2(1):113–120, 1972.
Cited on page 12.

[11] P. Bonacich and P. Lloyd. Eigenvector-like measures of centrality for asymmetric relations. Social Networks, 23(3):191–201, 2001.
Cited on page 12.

[12] S. P. Borgatti and M. G. Everett. A graph-theoretic perspective on centrality. Social Networks, 28(4):466–484, 2006.
Cited on page 12.

[13] Anatoly S. Buzdalov, M. A provably asymptotically fast version of the generalized jensen algorithm for non-dominated sorting. In International Conference on Parallel Problem Solving from Nature, volume 1, pages 528–537. Springer International Publishing, 2014.
Cited on page 71.

[14] Deepayan Chakrabarti, Yang Wang, Chenxi Wang, Jurij Leskovec, and Christos Faloutsos. Epidemic thresholds in real networks. ACM Transactions on Information and System Security (TISSEC), 10(4):1, 2008.
Cited on page 83.

[15] Chen Chen, Hanghang Tong, B Aditya Prakash, Charalampos E Tsourakakis, Tina Eliassi-Rad, Christos Faloutsos, and Duen Horng Chau. Node immunization on large graphs: Theory and algorithms. IEEE Transactions on Knowledge and Data Engineering, 28(1):113–126, 2016.
Cited on pages 83, 85, 86, and 88.

[16] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. A. M. T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2):182–197, 2002.
Cited on pages 57 and 70.

[17] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T.A.M.T Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2):182–197, 2002.
Cited on pages 21 and 95.

[18] G. Didier, C. Brun, and A. Baudot. Identifying communities from multiplex biological networks. PeerJ 3, e1525, 2015.
Cited on page 70.

[19] Peter. J. Fleming, Robin C. Purshouse., and R. J. Lygoe. Many-objective optimization: An engineering design perspective. Evolutionary multi-criterion optimization, pages 14–32, 2005.
Cited on page 11.

[20] Galton Francis. Regression towards mediocrity in hereditary stature. Journal of the Anthropological Institute of Great Britain and Ireland, 15:246–263, 1886.
Cited on page 14.

[21] L. Freeman. Centrality in social networks conceptual clarification. Social Networks, 1(3):215–239, 1979.
Cited on page 12.

[22] Koen Frenken. A complexity approach to innovation networks. the case of the aircraft industry (1909–1997). Research Policy, 29(2):257–272, 2000.
Cited on page 37.

[23] A.P. Giotis, KC Giannakoglou, and J. Périaux. A reduced-cost multi-objective optimization method based on the Pareto front technique, neural networks and pvm. In Proceedings of the ECCOMAS, 2000.
Cited on page 9.

[24] Sergio Gómez, Pablo Jensen, and Alex Arenas. Analysis of community structure in networks of correlated data. Phys. Rev. E, 80:016114, Jul 2009.
Cited on page 24.

[25] H. Hu, Y. van Gennip, B. Hunter, A. L. Bertozzi, and M. A. Porter. Multislice modularity optimization in community detection and image segmentation. In IEEE,12th International Conference on Data Mining Workshops, pages 934–936. IEEE, 2012.
Cited on page 70.

[26] Hisao Ishibuchi, Naoya Akedo, Hiroyuki Ohyanagi, and Yusuke Nojima. Behavior of EMO algorithms on many-objective optimization problems with correlated objectives. In Evolutionary Computation (CEC), 2011 IEEE Congress on, pages 1465–1472. IEEE, 2011.
Cited on page 21.

[27] Hisao Ishibuchi, Noritaka Tsukamoto, and Yusuke Nojima. Evolutionary many-objective optimization: A short review. In IEEE congress on Evolutionary Computation, pages 2419–2426. Citeseer, 2008.
Cited on page 104.

[28] Antonio López Jaimes, Carlos A Coello Coello, and Jesús E Urías Barrientos. Online objective reduction to deal with many-objective problems. In Evolutionary Multi-Criterion Optimization, pages 423–437. Springer, 2009.
Cited on pages 21 and 22.

[29] Renaud Lambiotte Jean-Loup Guillaume. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008.
Cited on page 24.

[30] Stuart Kauffman and Simon Levin. Towards a general theory of adaptive walks on rugged landscapes. Journal of theoretical Biology, 128(1):11–45, 1987.
Cited on pages 37 and 39.

[31] L. Ke, Q. Zhang, and R. Battiti. A multiobjective evolutionary algorithm using decomposition and ant colony. IEEE transactions on cybernetics, 43(6):1845–1859, 2002.
Cited on pages 57 and 70.

[32] M. Kivelä, A. Arenas, M. Barthelemy, J.P. Gleeson, Moreno, Y., and M.A. Porter. Multilayer networks. Journal of complex networks, 2(3):203–271, 2014.
Cited on page 70.

[33] J. Knowles and D. Corne. Quantifying the effects of objective space dimension in evolutionary multiobjective optimization. In Evolutionary multi-criterion optimization, pages 757–771, 2007.
Cited on pages 10 and 56.

[34] Hsiang-Tsung Kung, Fabrizio Luccio, and Franco P. Preparata. On finding the maxima of a set of vectors. IEEE transactions on Cybernetics, 22(4):469–476, 1975.
Cited on page 71.

[35] M Leacock. Pandemic.[board game]. Z-Man Games: Mahopac, NY, 2008.
Cited on page 88.

[36] Y. Li, Y. Wang, J. Chen, L. Jiao, and R. Shang. Overlapping community detection through an improved multi-objective quantum behaved particle swarm optimization. Journal of Heuristics, 21(4):549–575, 2015.
Cited on page 71.

[37] Daniel R Lucey and Lawrence O Gostin. The emerging zika pandemic: Enhancing preparedness. Jama, 315(9):865–866, 2016.
Cited on page 83.

[38] David Lusseau, Karsten Schneider, Oliver J Boisseau, Patti Haase, Elisabeth Slooten, and Steve M Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology, 54(4):396–405, 2003.
Cited on page 88.

[39] Barigozzi Matteo, Giorgio Fagiolo, and Diego Garlaschelli. Multinetwork of international trade: A commodity-specific analysis. Physical Review, 81(4), 2010.
Cited on pages 60, 70, and 73.

[40] A. Maulana, Z. Jiang, J. Liu, T. Bäck, and M. Emmerich. Reducing complexity in many objective optimization using community detection. In Proceedings of IEEE Congress on Evolutionary Computation (CEC), pages 3140–3147. IEEE, 2015.
Cited on pages 10, 56, 58, and 61.

[41] Asep Maulana, Valerio Gemmetto, Diego Garlaschelli, Iryna Yevesyeva, and Michael Emmerich. Modularities maximization in multiplex network analysis using many-objective optimization. In IEEE Symposium Series on Computational Intelligence (SSCI), pages 1–8. IEEE, 2016.
Cited on pages 73, 79, and 103.

[42] M. E. Newman. Modularity and community structure in networks. In Proceedings of the national academy of sciences, pages 8577–8582, 2006.
Cited on page 11.

[43] K. Okamoto, W. Chen, and X.-Y. Li. Ranking of closeness centrality for large-scale social networks. Springer, Frontiers in Algorithmics, 186195, 2008.
Cited on page 12.

[44] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Stanford InfoLab, 1999.
Cited on page 12.

[45] Pearson. SPSS tutorials: Pearson correlation. Statistical Science, 1(1), 1989.
Cited on page 14.

[46] Karl Pearson. Notes on regression and inheritance in the case of two parents. Proceedings of the Royal Society of London, 58:240–242, 1895.
Cited on page 14.

[47] Aske Plaat. Data science and ebola. arXiv preprint arXiv:1504.02878, 2015.
Cited on page 83.

[48] Robin C Purshouse and Peter J Fleming. Evolutionary many-objective optimisation: An exploratory analysis. In Evolutionary Computation, 2003. CEC'03. The 2003 Congress on, volume 3, pages 2066–2073. IEEE, 2003.
Cited on page 21.

[49] B. Ruhnau. Eigenvector-centralitya - a node-centrality? Journal of Heuristics, 22(4):357–365, 2000.
Cited on page 12.

[50] D. K. Saxena, A. Duro, J. A.and Tiwari, K. Deb, and Q Zhang. Objective reduction in many-objective optimization: Linear and nonlinear algorithms. evolutionary computation. IEEE Transactions on, 17(1):77–99, 2013.
Cited on pages 10 and 56.

[51] Dhish Kumar Saxena, Joao A Duro, Ashutosh Tiwari, Kalyanmoy Deb, and Qingfu Zhang. Objective reduction in many-objective optimization: Linear and nonlinear algorithms. Evolutionary Computation, IEEE Transactions on, 17(1):77–99, 2013.
Cited on pages 21 and 22.

[52] John R. Seeley. The net of reciprocal influence. a problem in treating sociometric data. anadian Journal of Experimental Psychology, 3(4):234, 1949.
Cited on page 12.

[53] Stephen M. Stigler. "Francis Galton's account of the invention of correlation. Statistical Science, 4(2):73–79, 1989.
Cited on page 14.

[54] Andrej Mrvar Vladimir Batagelj. Pajek - program for analysis and visualization of large networks. Timeshift - the world in twenty - fife years : ARS Electronica, 2004.
Cited on page 24.

[55] Tobias Wagner, Nicola Beume, and Boris Naujoks. Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In Evolutionary multi-criterion optimization, pages 742–756. Springer, 2007.
Cited on page 21.

[56] Tobias Wagner, Nicola Beume, and Boris Naujoks. Pareto, aggregation, and indicator-based methods in many-objective optimization. In International conference on evolutionary multi-criterion optimization, 2007.
Cited on pages 57 and 70.

[57] Handing Wang and Xin Yao. Objective reduction based on nonlinear correlation information entropy. Soft Computing, 20(6):2393–2407, 2016.
Cited on pages 10 and 56.

[58] Weinberger and D Edward. Local properties of Kauffman's N-K model: A tunably rugged energy landscape. Physical Review A, 44(10):6399, 1991.
Cited on page 43.

[59] X. Wen, N. Chen, Y. Lin, T. Gu, H. Zhang, Y. Li, and J. Zhang. A maximal clique based multi-objective evolutionary algorithm for overlapping community detection. Journal of IEEE Transactions on Evolutionary Computation, 21(3):363–377, 2017.
Cited on page 70.

[60] Darrell Whitley. A genetic algorithm tutorial. Statistics and computing, 4(2):65–85, 1994.
Cited on page 86.

[61] Jihoon Yang and Vasant Honavar. Feature subset selection using a genetic algorithm. IEEE Intelligent Systems and their Applications, 13(2):44–49, 1998.
Cited on page 87.

[62] Wayne W Zachary. An information flow model for conflict and fission in small groups. Journal of anthropological research, 33(4):452–473, 1977.
Cited on page 88.

[63] Y. Zeng and J. Liu. Community detection from signed social networks using a multi-objective evolutionary algorithm. In Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, pages 259–270. Springer, 2015.
Cited on page 71.

[64] Qingfu Zhang and Hui Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. Evolutionary Computation, IEEE Transactions on, 11(6):712–731, 2007.
Cited on page 21.

[65] Eckart Zitzler, Marco Laumanns, Lothar Thiele, Eckart Zitzler, Eckart Zitzler, Lothar Thiele, and Lothar Thiele. SPEA2: Improving the strength pareto evolutionary algorithm, tik report 103, 2001.
Cited on page 21.

Samenvatting

Dit proefschrift combineert twee gebieden van de informatica: Complexe netwerken, multicriteria optimization en complexe netwerkanalyse. Enerzijds onderzoekt het methoden uit de analyse van complexe netwerken teneinde conflicten en afhankelijkheden tussen objective functions te begrijpen wanneer er een groot aantal criteria is. Andersom stellen we voor om multicriteria optimization te gebruiken om een aantal problemen in multicriteria optimization te analyseren en op te lossen. Onze methoden vinden toepassingen in het vinden van een goede locatie voor een faciliteit, in de analyse van economische en biologische netwerken, en in epedemiologie.

Veel optimalisatieproblemen bevinden zich in een situatie waar meerdere groepen verschillende belangen hebben. Er is bovendien een dermate groot aantal opties waartussen gekozen moet worden dat zoekalgoritmes nodig zijn om die ruimte te verkennen. Het plannen van een stadsinrichting is een voorbeeld van zo'n situatie: er moeten bijvoorbeeld optimale locaties gevonden worden voor scholen en ziekenhuizen. Idealiter is elke burger tevreden en zijn zowel de economische kosten als de lasten op het mileu minimaal. Als het aantal criteria erg groot is, wordt het voor menselijke beleidsmakers moeilijk om het probleem te begrijpen en te bediscussiëren. Zelfs bestaande optimalisatiealgoritmes hebben moeite met het verwerken van dergelijke problemen met een groot aantal beoordelingscriteria. In dit proefschrift wordt het Community-Detection for Many Objective Optimization (CoDeMo, of Groependetectie voor Optimaliseren van Grote Aantallen van Criteria) ontwikkeld. Voor een probleem met veel beoordelingscriteria vindt het correlaties tussen de doelfuncties. Op basis van de correlatiematrix bouwen we een netwerk waarin we groepen van complementaire criteria identificeren. Zo ontbinden we het probleem in onafhankelijke deelproblemen die we oplossen met state-of-the-art technieken in multicriteria optimization, waarna we de oplossingen weer samenvoegen. CoDeMo is succesvol toegepast op problemen met 30 en met 50 doelcriteria voor het vinden van een goede locatie van een faciliteit, en op de constructie van een bepaald genregulatienetwerk (een NK-landschap) met 10 doelcriteria.

Een grote uitdaging in de analyse van complexe netwerken is om in multiplex netwerken groepen te detecteren, en om de centraalheid van knopen te bepalen. Een

netwerk is multiplex wanneer een bepaalde verzameling knopen verbonden is met meerdere typen takken. Zo vormen de verschillende takken verschillende lagen in het netwerk. Een voorbeeld is een netwerk waar de knopen landen zijn, en een tak tussen landen hun handel in een bepaald product voorstelt. Het is over het algemeen moeilijker om in multiplex netwerken groepen te detecteren dan in gewone netwerken met maar één type tak, omdat in dit voorbeeld landen die een cluster vormen in één type handel (bijvoorbeeld olie) niet noodzakelijkerwijs ook een cluster vormen in een ander type handel (bijvoorbeeld koffie). Zo kan het voorkomen dat een land dat centraal ligt in de oliehandel niet centraal ligt in de koffiehandel.

In dit proefschrift is de nieuwe tool nieuwe Pareto Front Modularity for Multiplex Networks (PaMoPlex) ontwikkeld om in multiplex netwerken groepen te detecteren en vast te stellen in hoeverre groepen in verschillende lagen overlappen of juist verschillen. Hiervoor is multiobjective combinatorial optimization een essentieel ingrediënt. Het geeft ons namelijk een matrix van Pareto fronts die we kunnen gebruiken om de knopen in het netwerk in groepen in te delen op basis van in welke lagen ze overeenkomen in hun groepsstructuur. PaMoPlex vertelt ook hoeveel onafhankelijkheid en conflict er is tussen de gevonden groepen. Een soortgelijke aanpak is gebruikt om de centraliteit van knopen te bepalen. Bestaande opsommingsalgoritmes kunnen daarna een lijst maken van alle Pareto-optimale oplossingen. We passen PaMoPlex toe op het netwerk van wereldhandel in elf goederen, en vinden dat er slechts zeven Pareto-optimale landen zijn, waarvan er destijds vijf in de G8 zaten.

Als laatste onderzoeken we het probleem van immunisatie in een complex netwerk. Dit probleem heeft recentelijk veel aandacht verdiend vanwege het verhoogde risico op een epedemie in een globaliserende wereld. De Ebola- en Zikauitbraken zijn twee voorbeelden, maar soortgelijke verspreiding doet zich voor bij computervirussen. Het immunisatieprobleem is om het netwerk zo aan te passen dat de verspreiding van het virus wordt voorkomen of afgeremd met minimale kosten. Het idee is om een bepaalde verzameling van de knopen in het netwerk in quarantaine te stellen. Het aantal mogelijke keuzes is echter erg groot en het optimalisatieprobleem, bekend als het subset selection problem, is NP-Hard, dus voor goede oplossingen zijn heuristische zoekmethoden nodig. We ontwikkelen hiervoor een genetisch algoritme dat het probleem in sommige gevallen beter oplost dan het state-of-the-art Netshield Plus algoritme, dat gebaseerd is op het "gretig"verwijderen van knopen. We stellen ook voor om met twee doelcriteria te werken, namelijk de kosten en de baten van de immunisatiestrategie, om zo alle Pareto-

optimale strategieen te vinden. Deze methode is geëvalueerd op het vliegverkeer in de VS, het bordspel Pandemic, en benchmarks uit de literatuur.

Summary

This thesis is combining two fields of computer science: Multicriteria optimization and complex network analysis. On the one hand, it investigates methods from complex network analysis to understand conflicts and dependencies between objective functions in many objective optimization, that is multicriteria optimization with a large number of criteria. On the other hand, it proposes the use of multicriteria optimization to analyze and solve problems in complex networks. In particular, problems to modularity and centrality in multiplex networks are addressed. Applications of the proposed methods are found in facility location problems, economic and biological network analysis, as well as in epidemiology.

Many objective optimization problems occur in settings where various interest groups and a diverse set of criteria have to be considered in order to find an optimal decision. Moreover, the number of possible decisions is very high and search algorithms are needed to explore the decision space. An example could be an urban planning problem, such as finding optimal locations of hospitals and schools. Ideally each citizen of the town should be satisfied and costs (both economic and environmental) should be minimized. If the number of criteria is very big, it is very difficult for human decision makers to understand and to discuss the problem. Moreover, optimization algorithms have difficulties to process large number of objectives. The Community Detection for Many-Objective Optimization (CoDeMo) technique is developed in this thesis for structuring: First we have a problem with many objectives, then evaluate this problem to determine correlations between objective functions. Based on the correlation matrix we can construct a network and detect the communities (clusters) of the network. Then we decompose the problem into independent subproblems and aggregate clusters with complementary objectives. Then the lower dimensional subproblems can be solved with state-of-the-art techniques in multi-objective optimization, and the solutions of the subproblems are merged again. The CoDeMo process has been successfully exemplified on a problem with 30 and a problem with 50 objectives in facility location, and for a model problem on gene regulatory network synthesis (NK- landscapes) with 10 objectives.

A challenging problem in complex network analysis is the detection of communities

and the assessment of centrality of nodes in multiplex networks. A multiplex network is a network with a fixed set of nodes, but different sets of edges (links). The different sets of edges form different layers of the multiplex network. For instance, in an economical networks one could consider different countries as nodes, and trade networks in different commodities, where for each type of commodity (e.g. oil and coffee) corresponds to one layer. It is in general more difficult to detects communities or clusters in these networks, because countries that form a trade cluster concerning one layer (e.g. oil trade) may not form a cluster in another layer (e.g. coffee trade). More important, the importance (centrality) of countries or nodes might differ from layer to layer, that is a country that is important in oil trade might not be important in coffee trade. The novel Pareto Front Modularity for Multiplex Networks (PaMoPlex) has been developed in the thesis as a tool to analyze the community structures Multiplex networks and to which extent they overlap between layers or differ. For this multi-objective combinatorial optimization is an essential ingredient. It yields a matrix of Pareto fronts that can then be grouped based on layers that are similar in their community structure. Also independence and the degree of conflict between communities can be assessed by means of PaMoPlex. It is applied to the world trading economy network layered into 11 commodity groups. Moreover, a similar approach was used to analyse the centrality of nodes. Here, enumeration algorithms can be used to find all Pareto optimal solutions, that is nodes that when compared to any other node are more important at least in one layer with respect to that node. It was found for the world economic trade network that there were seven Pareto optimal countries, of which five of them were in the G8 at this time.

Finally, the problem of immunization of complex networks was investigated. This problem has recently achieved a lot of attention, due to the increased risk of pandemia in a globalized world. The outbreak of Ebola and Zika are two examples. Moreover, similar spread can be observed when a computer virus spreads in the internet.

The problem of immunization is to modify networks in such a way, that the spreading of a virus is inhibited. Moreover, the cost of implementing the modification of the network should be minimized. The idea is to immunize or quarantine a subset of nodes in the network in order to decelerate or prevent the spread of a virus. However the number of possible subsets of nodes in a network is very large and heuristical combinatorial search needs to be applied, also due to the proven NP hardness of the resulting subset selection problem.

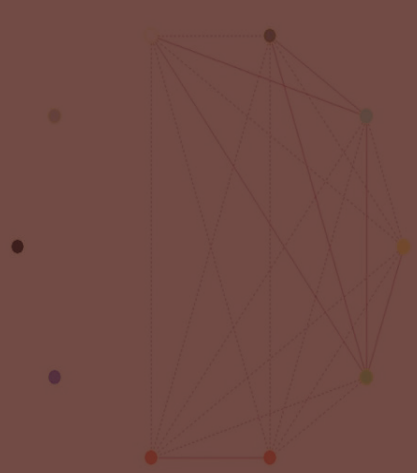
We developed a problem specific genetic algorithms that can solve the problem in some cases much better than the Netshield Plus algorithm, that so far was the state-of-the-art approach and based on a greedy node removal strategy. In this thesis it was also suggested for the first time to use bi-objective combinatorial optimization to solve this problem, that is, to find all Pareto optimal solutions for it, trading-off the cost of a immunization strategy and its effect. The approaches were tested on example networks, including the US flight network, pandemics board game network, and networks from the common benchmark networks used in literature.

About the Author

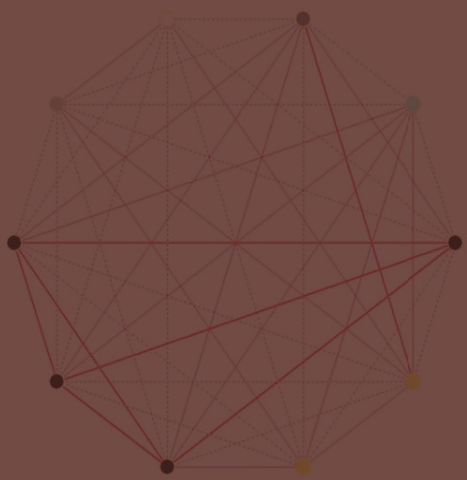
Asep Maulana was born on March 3rd, 1974 in Bandung, Indonesia. He completed his bachelor studies in Computer Science at the Faculty of Mathematics and Natural Science Padjadjaran University Bandung, Indonesia, in 2002. From 2003 - 2005 he starts to worked as a teacher at Senior High School level for mathematics.

In 2005, he then moved to Sweden to start his Master study in Computer Engineering with a specialty in Intelligent Systems. He finished his Master in 2007 with the Master thesis on Bus Time Table optimization with J2ME (Java for Mobile Application). After returning back to Indonesia, he come back to the previous job to serve as a teacher in Mathematics. In 2009 he started to work as a teacher at Telkom University Bandung, teaching for courses at the Bachelor level such as Algorithms and Programming, Mathematics Discrete and Artificial Intelligence.

In February 2014, he moved to Leiden, The Netherlands to start his Ph.D. studies within the research group headed by Prof. Thomas Bäck at the Leiden Institute of Advanced Computer Science (LIACS). He was supervised by Prof. Thomas Bäck and Dr. Michael Emmerich. His Ph.D. research studies on many-objective optimization and complex network analysis were funded by the Indonesian Endowment Fund for Education (LPDP). During his Ph.D. research, he collaborated with researchers in LCN2 (Leiden Complex Networks Network) and LCDS (Leiden Center of Data Science).

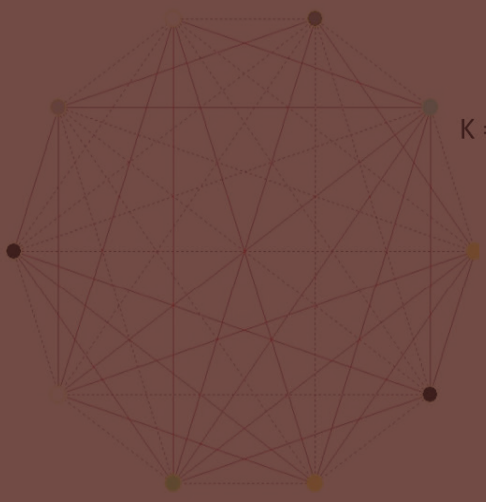


K = 0

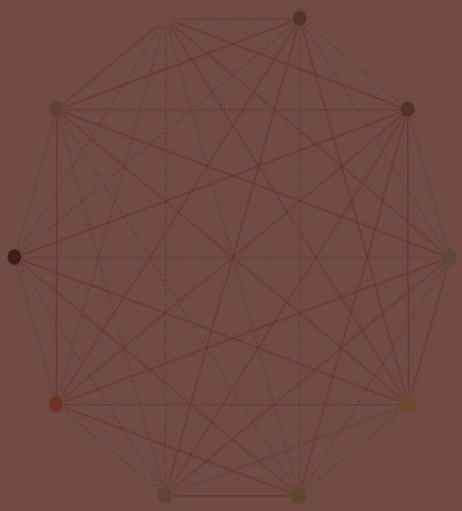


K = 1

K



K = 3



K = 5

K = 6

978-94-6375-227-5