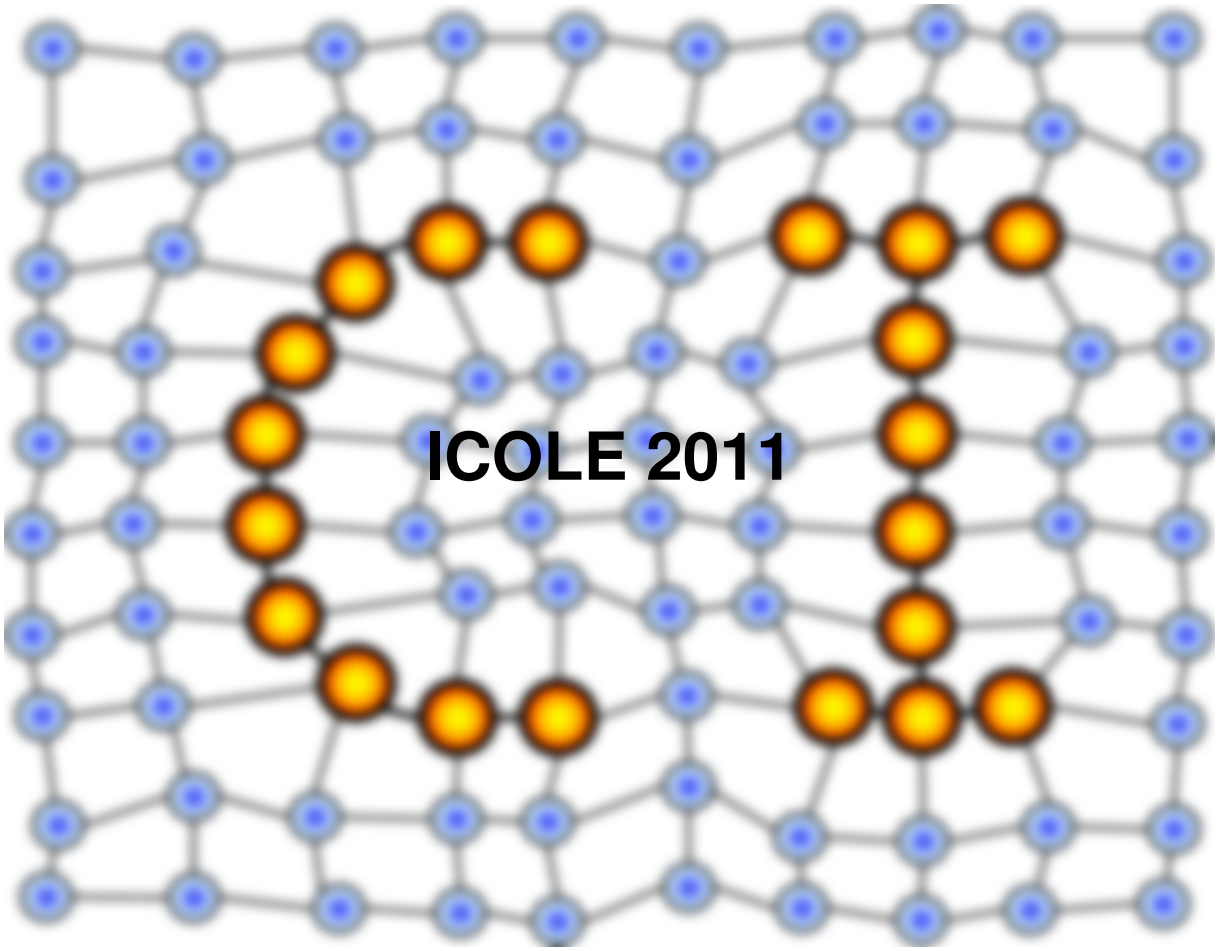
MACHINE LEARNING REPORTS



Report 01/2012
Submitted: 30.11.2011
Published: 31.01.2012

ICOLE 2011, Lessach, Austria

Table of contents

Preview
(J. Blazewicz, K. Ecker, B. Hammer) 4

Structuring gene regulatory networks
(K. Ecker) 6

Protein structure quality assessment and modeling support techniques
(M. Antczak, P. Lukasiak, K. Fidelis, A. Kryshtafovych, J. Blazewicz) 21

Mind the gap: problem definition and properties
(M. Drozdowski, D. Kowalski, J. Mizgajski, D. Mokwa, G. Pawlak) 27

Mind the gap: solutions and algorithms analysis
(M. Drozdowski, D. Kowalski, J. Mizgajski, D. Mokwa, G. Pawlak) …..... 31

The software development for optimization of production technology for selected energy crops
(A. Gotfryd, G. Pawlak, W. Wojciechowicz) …..... 36

Buffer management in car sequencing problem
(G. Pawlak, W. Wojciechowicz) …..... 41

Advanced multi-item internet shopping
(J. Blazewicz, J. Musial) …..... 48

Genetic algorithm for order completion and delivery problem
(M. Cicheński, M. Jarus, M. Miszkiewicz, M. Sterna, J. Szymczak) …..... 51

RFID – possible applications and challenging problems
(G. Fenrich, M. Sterna) …..... 59

Extensions of learning vector quantization for relational data …..... 67
(X. Zhu, F.-M. Schleif, B. Hammer)

Analyzing motion data by clustering with metric adaptation
(B. Mokbel, M. Heinz, G. Zentgraf)..... 70

Theory of patch clustering for variants of fuzyy c-means, fuzzy neural gas, and fuzzy self-organizing map
(T. Villmann, M. Kästner, M. Lange) …..... 80

Preview

Jacek Blazewicz*, Klaus Ecker†, Barbara Hammer‡

In the frame of the annual Polish-German workshop on Computational Biology, Scheduling, and Machine Learning, ICOLE'2011, twenty-three scientists from Poznan University, Clausthal University of Technology, University of Applied Sciences Mittweida, and Bielefeld University came together in Lessach, Austria, during 18.9.-24-9.2011. In the frame of the workshop, twenty-one talks on different topics including scheduling, bioinformatics, and machine learning were presented, accompanied by vivid discussions, and exchange of novel ideas at the cutting edge of research. This year, for the first time, a best presentation award was given to the joint talk of Mateusz Cicheński and Jarek Szymczak on 'Genetic Algorithms for Order Completion and Delivery Problems'. Further, also for the first time, one contribution of this volume concerns a novel algorithmic proposal developed at the workshop in the frame of patch clustering for fuzzy classification algorithms, as presented by Thomas Villmann, Marika Kästner, and Mandy Lange.

This volume contains twelve extended abstracts accompanying the presentations and ongoing work at the workshop. The first two papers cover challenging problems in computational biology: The contribution 'Structuring Gene Regulatory Networks' gives an overview about gene regulatory network models and algorithms for their discovery, a crucial issue to understand the fundamentals of cell processes. In the article 'Protein structure quality assessment and modeling support techniques', the focus is put on quality assessments for predicted protein structures, which play a fundamental role to explain the function of proteins in biological processes.

Six contributions center around topics in scheduling and discrete optimization: Two papers are connected to a novel problem, 'Mind the gap', the problem of finding a minimum cycle in a transportation system which uses all subway lines. This problem can be formalized as an interesting graph problem with different possibilities to arrive at solution strategies. The project 'The software development for optimization of production technology for selected energy crops' has been conducted in the frame of new European union directives concerning the increase of energy from renewable sources, where optimization techniques can contribute to increase profitability. A problem from car manufacturing is the central issue in the

*Institute of Computing Science, Poznan University of Technology, Poznan, Poland
†Department of Computer Science, Clausthal University of Technology, Clausthal-Zellerfeld, Germany
‡CITEC centre of excellence, Bielefeld University, Bielefeld, Germany

contribution 'Buffer Management in Car Sequencing Problem'. Interestingly, the objective in this setting is not only to reduce the inventory level, but, essentially, to provide stable solutions which allow to fulfill customer demands at the same time. In the contribution 'Advanced multi-item Internet shopping', another novel optimization problem is introduced which has its background in the price policy of internet shops: how to optimum select shops for items if batching can yield discount rates. In the contribution, its connection to the facility layout problem as well as a greedy heuristics are discussed. Another optimization problem connected to trading is tackled in the article 'Genetic Algorithm for Order Completion and Delivery Problem'. It addresses the problem of resellers who aim at an optimization of the purchasing and transportation costs at the same time. A formal description as well as solution strategies based on genetic algorithms are provided.

In the article 'RFID - Possible Applications and Challenging Problems' the novel radio frequency technology RFID is addressed, discussing promises and risks of this development.

Finally, three papers center around topics in machine learning: The article 'Extensions of Learning Vector Quantization for Relational Data' proposes a prototype based method which can directly work with dissimilarity data such as e.g. bioinformatics sequences compared by alignment. An application to motion capture data constitutes the key topic of the contribution 'Analyzing Motion Data by Clustering with Metric Adaptation'. The representation of such data and its capturing using Kinect as well as suitable learning algorithms are discussed in this frame. A novel algorithmic development is presented in the contribution 'Theory of Patch Clustering for Variants of Fuzyy c-Means, Fuzzy Neural Gas, and Fuzzy Self-Organizing Map', where patch clustering, which has already successfully been used in the context of crisp clustering, is transfered to fuzzy clustering algorithms.

Altogether, these contributions demonstrate the lively and fruitful scientific atmosphere caused by the interesting scientific range of the workshop, its international participants, and, last not least, the excellent possibilities offered by Daublebsky's wonderful house in Lessach and its surroundings. As usual, the scientific program has been extended by more social aspects or combinations of science and art such as a world premiere bringing the blues into fuzzy clustering.

Structuring Gene Regulatory Networks

Klaus Ecker, University of Technology, Clausthal, Germany

Abstract

It is generally accepted that the genetic code plays a significant role for the correct course of actions in the cell, whose functions are determined by the cell chemistry and controlled by biochemical conditions and external factors. Elucidating the organization of the gene regulatory network is crucial for understanding the cell behavior in early embryonic and later developmental stages of the organism. The main objective of this article is to give an overview on gene regulatory network models and algorithms. Choosing the right type of model is important because it influences the way of how to tackle the problem of analyzing the structure. In fact, some of the algorithms are based on a precise network model, while others use an experimentally motivated or even implicitly given representation model. Depending on the intended level of details one can choose from several definitions ranging from simple correlation networks, clustering methods, and circuit diagrams (Boolean networks and Petri nets) to more sophisticated models such as hypergraphs. Each gene regulatory network model emphasizes certain structural aspects and provides descriptive insights, allowing model-specific analysis that can better relate predicted behavior and observations.

1 Introduction

During the last few decades huge effort was put in deciphering the genome. Through increasingly faster sequencing methods the genomes from many different species became available in ever shorter time. Yet, besides limited insights, the meaning of the genetic code is largely unknown. It is, however, generally accepted that the code plays a significant role for the correct course of actions in the cell, whose functions are determined by the cell chemistry and controlled by biochemical conditions and external factors. Citing Basso et al. [1], *Cell activity is coordinated by a complex network that regulates the expression of thousands of genes*. Elucidating the organization of the gene regulatory network (GRN) is crucial for understanding the cell behavior in early embryonic and later developmental phases of the organism, and as well as reactions to external effects.

GRN analysis focuses on the network structure by concentrating on the question, which subsets of gene products (i.e., regulatory proteins) enable a gene to be expressed. The purpose of regulatory network analysis is to learn about the principles behind gene regulation, to understand the rules life is being built on, and to develop methods to conquer deceases. The biological mechanisms in the cell and its functions need to be understood, including protein concentrations, diffusion speed, decomposition, binding forces between transcription factor proteins and DNA binding sites, and many more. As biological sciences are yet far from having a complete picture and still there is no generally accepted way available for arriving at such picture, it seems more promising to concentrate on partial problems. “Functional genomics” [2] investigating gene regulatory mechanisms is a sub-problem that found great interest in the last decade because it appears to play a crucial role for understanding many cell functions and allows creating tools for formalizing them.

Networks for regulating cell function can be considered at different detailed levels:

Coarse level deals with conditions for gene activation and inhibition. A coarse-grained model would concentrate on the structure of the gene regulatory network, i.e., (*i*) deal with the question about which genes are regulated by the outcomes of regulatory genes, and (*ii*) what the conditions and cell states are to make this happen. Help can be obtained from comparative genomics focusing on co-regulation to see which genes are commonly expressed and under which conditions. It is known from many experiments that generally co-regulators act together to activate or repress a target gene (see [3, 4]).

Likewise, evolutionary relationships can be revealed from genes with similar expression patterns. Particular objectives of coarse grain GRN analysis are [5, 6]:

- identifying functional modules (kernels), i.e., subsets of genes that regulate each other, but have only few interactions with genes outside the subset, with respect to important cell processes or particular phases of development,
- and studying the behavior in case of perturbations and predicting the response, and directly identifying the affected genes (for example drug response in drug discovery processes).

Detailed level : For modeling the cell behavior a fine-grained system-theoretic approach would determine under which conditions and with what strength a transcription factor (TF) binds to a binding site (TFBS). Transcription factors are proteins usually produced by regulatory genes, which mediate the expression of other genes. A refined model would help to assess the impact of TF concentration and local inhomogeneities, measure the intensity or estimate the probability of resulting gene regulation (expression, repression), consider diffusion speed and disintegration rate of regulatory proteins, and other biochemical details (see, for example, [7]).

In this overview we confine ourselves to the coarse level network structure. Depending on the intended purpose and required level of details one can choose from several definitions of GRNs, ranging from simple correlation networks, clustering methods, circuit diagrams (Boolean networks and Petri nets) to more sophisticated models such as hypergraphs. Each GRN model emphasizes certain structural aspects and provides descriptive insights, allowing model-specific analysis which can better relate predicted behavior and observations [8]. There is ample literature available about this topic. Introductory publications to be mentioned are from Diambra [9] who gives a more or less complete picture of the ongoings, from Hecker et al. [10] and Walhout [11] because of the critical discussion about the suitability of representation methods in particular investigations.

Computational genome analysis and the outcomes highly depend on the chosen GRN model. Two objectives are pursued in this review,

- giving an overview of the various representation models,
- and discussing the properties and restrictions of the representation methods and important computational issues for elucidating the network structure.

One of the simpler structures is correlation networks that have been introduced to illustrate simple gene-gene interactions ([12, 13, 14]. Another structuring approach is clustering genes with similar expression patterns [15, 16, 17, 72]. Representation methods avoiding the drawback of undirected relationships are circuit diagrams (Boolean networks, Petri nets, and similar approaches) studied by Steggles et al. [18], and random Boolean networks by Zhao et al. [19]. Petri nets were also used by Steggles et al. [18] and Banks et al. [20]. Hybrid Petri nets for modeling discrete and continuous processes were studied by Matsuno et al. [21], Vasireddy et al. [22] and Chaouiya [23], and Gilbert et al. [24]. Bayesian networks for inferring the GRN structure were used by Wehrli et al. [25] and Needham et al. [26].

The main objective of this article is to describe the algorithmic foundations for getting insights in the complexity of gene regulation. In fact, some of the algorithms are based on a precise GRN model, but others use an experimentally motivated or even implicitly given representation model. This is an important point because the type of model used influences the way of how to tackle the problem of analyzing the GRN structure. It determines the method of analyzing the experimental data and how to interpret the inferred results. In section 2 we start with a short excerpt from experimental methods that should help to bring light into gene regulation. Methods for analyzing the “raw” data from experiments are listed in section 3. Section 4 presents higher structured models of gene regulatory networks that illustrate relationships and dependencies between regulatory genes, and proposes a novel method based on hypergraphs. Section 5 summarizes the presented material and offers an outlook for future developments.

2 Gene Regulation (Preliminaries)

2.1 Experimental methods

In the attempt to decipher the GRN one wants to know how and where in the genome the binding places of a TF are located, what the effect of binding a TF on a gene is, and moreover which TFs contribute to the gene's expression [11]. Well-known experimental methodologies [27, 28, 29, 30] helping to reach this goal are ChIP-chip, ChIP-sequencing, and microarrays including time series experiments.

Fundamentals. Let us start with experimental methods for getting expression information that is fundamental for inferring the structure of the regulatory network. Experiments are expected to show gene expression data in various conditions. In this process its code is transcribed into mRNA, which then changes into a structural protein, an enzyme, or a regulatory protein (TF).

Considering a particular point of time, a certain mixture of regulating chemicals exists in the cell. Depending on which TFs are present, one or more genes may be up- or down-regulated. Particularly for a structural network analysis, finding all sets of transcription factors regulating a gene (gene-centered approach) is a basic sub-step. Experimental methods based on Chromatin ImmunoPrecipitation (ChIP) combined with microarrays (ChIPchip) or ChIP-sequencing identify physical interactions between TFs and DNA [28]. ChIPchip methods account for detecting binding sites of a given transcription factor protein. ChIP-sequencing finds all genes activated by a TF in a condition, where "condition" concerns the availability of other transcription factors and signals, chemical stimulus, or physical conditions such as temperature, etc.. Gene expression data from a microarray experiment give a snapshot of gene activities in a particular condition. A snapshot visualizes the activated genes at a fixed point of time but it does not provide any information about the dynamic behavior of the regulatory system. Besides microarray experiments, computational and experimental discovery methods of regulatory motifs and *cis*-regulatory modules can supplement and consolidate the experimental information [31, 32]. Two types of microarray assays can be distinguished: Time series and steady-state experiments.

Time series experiments. Microarray time series measure the gene expression intensity at different time points, each providing a snapshot of activity levels of the genes implemented on the chip. From the series of snapshots one can see the dynamic progression of expression patterns. This is the source of a pathway analysis that informs about gene activations running like an avalanche through the GRN while controlled by regulatory proteins and the products of the genes regulated by these proteins. Typical application is comparing gene expression time series in unperturbed condition and under additional selective perturbation such as silencing genes (knock out), special chemical treatment or environmental perturbation such as heat shock, chemical stress, compound treatments etc.. Time series are expected to allow predicting time-dependent correlations between genes. On the other hand one has to keep in mind that there may be many hidden intermediate steps between the snapshots. Consequently, when inferring the state of a gene from other states one must make interpolation assumptions which most certainly will lead to poor estimates [33].

Steady-state measurements reveal gene expression profiles in different conditions (without/with perturbation such as silencing genes, applying particular chemical molecules or drugs) thus allowing the detection of similar expression patterns of genes of the same or of different species. Steady state experiments regarding a single time point are good for comparisons of gene expression behavior from which gene correlation can be concluded, but the direction of causality is difficult to capture from purely correlated data [34].

Quality of results. For each experimental method it is important to know about its limitations. In microarray experiments the data is often inaccurate and the quality is affected by measurement noise and systematic errors (bias). Therefore, inferred results of the GNR structure from inaccurate data have to be treated cautiously (see [11]).

2.2 Extracting gene expression data from experiments

Microarray experiments show expression intensities for the set of genes implemented on the chip. A single sample of measurement results in an expression profile in form of a numerical vector representing the expression intensity values of the genes. Time series experiments and experiments with differ-

ent condition settings result in a series of samples (see Figure 1). The corresponding profile vectors can be combined into a gene expression data matrix *S*, where each row represents a gene and each column represents a sample, which is a point in a time series experiment, or an experiment with particular condition settings. Each element, $s_i(j)$, of *S* indicates the expression level of gene g_i in sample *j*. Digitization methods are applied to measure the intensities. Since gene expression data is often noisy, a simple and most secure way is to distinguish between just two different states, 'expressed more than average'/'expressed less than average' [35, 36]. The state of a gene g_i in sample *j*, $s_i(j)$, is +1 if $s_i(j) > \overline{s}_i$ and 0 otherwise, where $\overline{s}_i$ is the average expression level of the i^{th} gene.

For time series experiments, let $s_i(t)$ be the expression level of g_i at time point *t*. The microarray data provide an expression history $s_i(t–L)$, …, $s_i(t–1)$, where *L* is the number of time points. Let $s(\tau) = (s_1(\tau), \ldots, s_n(\tau))$ be the expression profile at the point $\tau \in \{t, \ldots, t–L\}$.

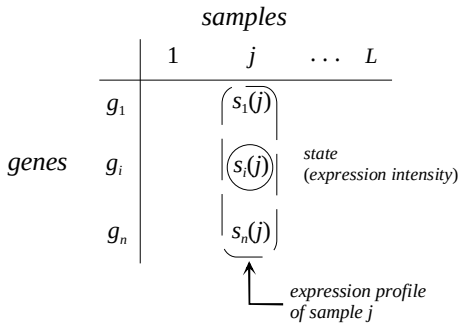


Figure 1: Expression matrix *S*

Given a gene expression matrix *S*, reverse engineering methods intend to infer the mechanisms for regulating gene expression. More specifically, we want to predict [33]:

(*i*) the state of gene g_i in sample *j* from the expression values of other genes in the same sample;

(*ii*) the state of gene g_i in sample *j* from the expression values of genes from the other samples;

(*iii*) the change in the state of gene g_i from the state changes of other genes.

2.3 Gene Regulatory Networks (GRNs)

As to our knowledge there is no generally accepted precise definition of gene regulatory networks available. Genes are usually represented by network nodes. Dependencies between nodes are differently modeled, by edges connecting pairs of nodes, by subsets of nodes or more complex constructs. The GRN is static and can be seen as the hardwired code for regulating the expression of genes, also referred to as "regulatory skeleton" [15]. It determines the dynamic behavior caused by the regulatory machinery interpreting and executing the genetic code. It is important to note that the GRN is independent of time and conditions, and of variable quantities such as molecular concentrations and kinetic reaction rates [37].

GRN analysis is the process of inferring the static structure from the dynamical behavior as obtained from experiments. The primary purpose is to gain a comprehensible view of the regulatory mechanism. On the other hand, if the conditions of a time series experiment are initially known it should be possible to verify the same pathway by simulation. In the future GRNs may also serve as a tool for prognosticating pathways under given or assumed conditions. For example, responses to drug application or the impact of other stimuli of chemical or physical nature would eventually be amenable by simulation.

3 Analysis of Microarray Data

3.1 Introductory Remarks

Analysis methods for gene regulatory networks follow the hypothesis that regulatory genes affect the expression of (other) genes, that themselves have regulatory function. If the knowledge of gene g_i's expression level yields information about another gene g_j, the genes are said to be correlated. To this purpose the correlation matrix W is defined as an $n \times n$ matrix, where n is the number of genes in the experiment. The matrix elements w_{ij} are often normalized to the interval [–1 , 1] with $w_{ij} < 0$ if regulatory gene g_i acts silencing on g_j , and $w_{ij} > 0$ if g_i produces a TF that encourages expression of g_j. The absolute value $|w_{ij}|$ measures the intensity of influence. The correlation matrix is an intermediate step toward a better understanding of the regulatory functions. However, as the number of genes in an experiment assay is usually much bigger than the number of samples, the correlation matrix has many more elements than the expression matrix, and hence cannot be uniquely determined from the observed data. Proposed methods to escape this shortage are (see [38])

- artificially enlarging the number of measurements by interpolation [39],
- imposing additional mathematical constraints [40, 41],
- finding sparse interaction matrices by combinatorial search strategies [12, 41, 42],
- reducing of the number of genes by eliminating redundant information [15],
- applying a clustering algorithm and restrict to cluster representative genes for which additional biological knowledge is available [38].

Computational methods revealing correlation data from time series experiments try to explain expression intensities from earlier intensities.

3.2 Static Systems Approaches

As cause should precede its effect, one expects that an expression profile $s(t)$ can generally be approximated by its expression history, for example by the linear regression

$$s_i(t) \approx \sum_{l=1}^{L} a_l s_i(t-l)$$

with properly chosen weight parameters a_l. A more accurate solution is found by not only considering the g_i's own history, but also by including expression histories from other genes. A time series of g_j is said to "Granger cause" the time series of g_i if

$$s_i(t) \approx \sum_{l=1}^{L} a_l s_i(t-l) + \sum_{l=1}^{L} b_l s_j(t-l)$$

[43, 44]. Granger causality has been proven to be more accurate than the first relation [44]. The constant parameters a_l and b_l must be accordingly adjusted from the expression intensities. Granger cause is used to decide if g_i is up- or down-regulated by g_j, or if both genes operate independently. Some authors also propose an extension of Granger causality to more than two genes, see Lozano et al. [45] for details.

Rangel et al. and Hirose et al. [46, 47] assume that the dynamical behavior of the observed profiles $s(\tau)$ can be modeled from the development of a small number of k hidden state variables. A set of hidden state variable vectors $x(t)$, …, $x(t–L)$ of dimension k defines a first-order Markov process

$$s(t) = Hx(t-1) + As(t-1) + \varepsilon_s(t-1)$$

$$x(t) = Fx(t-1) + Bs(t-1) + \varepsilon_x(t-1) .$$

Matrix H describes the influence of the hidden state variables to the current state vector; Matrix F describes the current changes of the hidden state variables; Matrix A captures the causal relationships of genes to the current state vector; Matrix B captures the influences of the previous gene expressions; The functions ε_s and ε_x describe uncorrelated white noise.

3.3 Dynamic Approaches

Measuring the gene expression levels over a series of time steps, the rate of expression change of a gene g_i can be described as a function of the expression levels of all genes,

$$\frac{ds_i}{dt} = f_i(s_1 \ldots s_n) \, , \; i = 1, \ldots, n$$

(see, for example, [48]). The functions f_i are supposed to capture all kinds of influence on the expression of g_i, including transcription factors and other chemical signals. The initially unknown functions f_i need to be guessed or approximated from the observed expression data. As such they should be simple to enable easy and fast calculations.

Linear differential equations. Choosing linear functions turn the above relation into a set of linear differential equations, in which undefined parameters are to be fitted to the given data,

$$\frac{ds_i}{dt} = \sum_{j=1}^{n} w_{ij} \, s_j(t) + \varepsilon_i(t) \, .$$

$s_j(t)$ is the expression intensity of gene g_i at time t, n is the number of genes, and $\varepsilon_i(t)$ is a stochastic variable capturing external perturbation in g_i. The regulatory weights w_{ij} are approximated from the time-course gene expression data by minimizing the squared error [49, 50].

Modeling by linear differential equation is a simplified approach that is merely able to capture the main features of a network. Though only an approximation, due to its simplicity it is good as a starting point for further investigations. To make it even simpler, the network structure can be made sparse by applying a LASSO-constraint [51] of the form

$$\sum_{j=1}^{n} | \, w_{ij} \, | \leq \mu_i$$

[52, 53], where μ_i is the Lasso constraint parameter for regulating the sparseness of the weight matrix.

Various other methods were discussed by Guthke et al. [38] and Bansal et al. [54]. Chen et al. [55] proposed a more detailed approach for modeling the dynamics of gene expression. The expression level (s_i) and transcription rate (G_i) of each gene are extracted from the microarray data, and are then used to model dynamics of gene expression by the differential equation

$$\frac{ds_i}{dt} = G_i(t) - \lambda_i s_i(t) + \xi_i(t) \, .$$

Here λ_i is the self-degradation rate of the produced TF or the inverse of the time delay from input $G_i(t)$ to output $s_i(t)$, and ξ_i is the noise capturing data uncertainty and model residuals.

Discretization of differential equation leads to difference equation models. The expression level of a target gene g_i at time $t + \Delta t$ can be derived from the expression levels of the regulators $g_i(t)$ at time t and the regulating weights of the genes controlling the target gene, see Hecker et al. [10],

$$(s_i(t + \Delta t) - s_i(t))/\Delta t = \sum_{i=1}^{n} w_{ij} \, s_j(t) + b_i \, u \qquad (i = 1, \ldots, n) \, .$$

b_i is the impact of perturbation on the expression of g_i and u is a perturbation signal.

Non-linear approaches. Other approaches use polynomials of degree 2 or higher [48]. Dynamic Bayesian networks based on a deterministic inertial model [56, 57] use a second order differential equation,

$$\frac{d^2 s_i(t)}{dt^2} + 2\lambda_i \omega_i \frac{ds_i(t)}{dt} + \omega_i^2 s_i(t) = \sum_j w_{ij} \, s_j(t) \quad ,$$

where $s_i(t)$ is the expression level of gene g_i (the quantity of mRNA produced by this gene) at time t, λ_i is an absorption coefficient characteristic for g_i , and ω_i is a natural frequency of g_i [56].

A non-linear model presented by Vu et al. [58] is derived by assuming recursive action of regulators on the target over time. The model uses a sigmoid function for the control of target gene expression s_i,

$$\frac{ds_i}{dt} = k_1 \frac{1}{1 + e^{-s_i}} - k_2 s_i \, .$$

The constant k_1 is the maximal rate of expression of the target gene g_i , and constant k_2 represents the degradation rate of the gene product.

Another non-linear approach is an improved S-system model [59], see also [60] where the gene expression rate is modeled by excitatory and inhibitory components (α_i, β_i):

$$\frac{ds_i}{dt} = \alpha_i \sum_{j=1}^{n} s_j^{g_{ij}} - \beta_i \sum_{j=1}^{n} s_j^{h_{ij}} \, .$$

g_{ij} and h_{ij} are kinetic components to adjust interactive parameters for modeling increasing and decreasing effects of g_j to g_i.

Inference by stochastic analyses, Bayesian networks. Stochastic differential equations are variations of the above concepts making use of the irregularity of gene expression which becomes apparent if, e.g., the number of TF molecules is low [61, 62]. One step further are Bayesian networks (BN), reflecting the stochastic nature of gene regulation. Gene expression values are described by random variables following probability distributions [25, 26]. Bayesian network as an inference tool for analyzing gene regulatory networks were early proposed because the statistical foundations for learning Bayesian networks from observations are well understood, they can capture complex stochastic relationships, accommodate noise due to their probabilistic nature, and are able to consider indirect influences through unobserved components [63, 64, 65]. A Bayesian network is defined by the joint distribution of random variables representing the independent observation of the expression levels of the genes and possibly other random variables for experimental conditions and noise.

So far a short overview on mathematical methods to infer expression correlations between pairs of genes. The correlation matrix is the jumping-off point for further structural analysis of the GRN which can be classified as:

- gene clustering methods,
- correlation networks and activation/inhibition graphs,
- circuit diagrams (Boolean networks and Petri nets),
- hypergraphs.

Next we give brief insights to these models and discuss corresponding inference approaches.

4 Modeling of Gene Regulatory Networks

4.1 Gene Clustering

The objective of clustering is to group genes with similar expression patterns. A co-expression cluster, also called transcription regulatory module [18], is a group of genes that are regulated by a common set of transcription factors. Microarray data showing gene subsets with the same expression level can be a useful hint for an initial assortment of genes possibly co-regulated under the experimental condition. Genes expressed with similar patterns in time series experiments may even give more detailed insights. In time series experiments and as well in steady-state experiments with changing conditions it is expected that genes still retain certain common regulatory behavior [17]. Genes sharing the same expression pattern are likely to be involved in the same regulatory process. If they share the same or similar regulatory sub-network, correlations between cis-elements and expression profiles are expected to exist [15, 66].

The Pearson correlation coefficient provides a simple way to measure the correlation of genes g_i and g_j arising from a series of samples [6]:

$$r_{ij} = \frac{\sum_{k=1}^{L} \left(s_i(k)\, s_j(k)\right)}{\sqrt{\left(\sum_{k=1}^{L} s_i^2(k) \sum_{k=1}^{M} s_j^2(k)\right)}} \ .$$

Clustering genes of similar expression profiles is symmetric: Clusters only describe correlations between genes, without showing any direction. Hence, clustering alone cannot resolve gene regulation dependencies [53]. This representation deficiency can be partially overcome by taping additional information sources such as specific experiments, analyzing promoter regions for common binding motifs, or *cis*-regulatory modules [32]. A structural solution proposed by Bolouri et al. [16] suggests modeling GRNs as hierarchical constructs of modular building blocks with distinct functionality, starting from minimal building blocks which are defined as basic transcriptional control processes executed by a few functionally linked genes [16].

Co-regulation and cross-species analysis. Genes with similar expression profiles in the same species and structurally similar genes in the same or different species are *key components of the biological response* [67], allowing deductions of high reliability.

Motivation for *co-regulation analysis*: At different places of the DNA similar regulatory structures can be found. These seem to be mostly due to copying processes that took place in the past. The related genes are expected to be similar, again due to copying processes. Interestingly enough, situations nevertheless have been found where the co-regulated genes show great diversity. In co-regulatory analysis, the gene expression profiles obtained from single chip experiments can be compared against itself. In time series assays the expression profiles of different time points can be compared.

Motivation for *cross-species analysis*: Two species sharing similar regulatory structures and regulating similar genes are expected to have a common ancestor with a regulatory structure from which the present structures were derived [3]. Comparing time-series experiments with DNA from different species can be difficult because conditions and reaction times may diverge from each other even for similar genes [67]. Depending on the evolutionary distance it is known that similar genes across different species often perform similar function and are mediated by similar regulatory mechanisms [13, 44, 66]. It has been pointed out by Lu Y [67] that successful application of cross-species analysis leads to insights that cannot be obtained when analyzing data from a single species.

4.2 Correlation Network

One of the simplest modeling methods of GRNs is the correlation network, which is an undirected graph with nodes representing genes and undirected edges connecting pairs of genes that influence each other [13]. Edges are weighted by a numerical coefficient expressing the intensity of the correlation. By imposing a correlation threshold, sparingness of the graph can be controlled.

Activation/Inhibition networks [12, 14] are refined correlation networks, where the edges are oriented to inform about the direction of influence, as described by the correlation matrix *W*. A positively weighted edge from g_i to g_j is drawn as $g_i \rightarrow g_j$, negatively weighted edges are drawn as $g_i \dashv g_j$ (see figure 2). The absolute weight value can be interpreted as the intensity of influence.

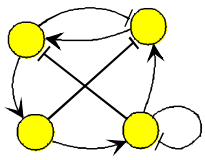


Figure 2: An example activation/inhibition graph

The activation/inhibition network is the basis for further investigations. One immediate question concerns the modules of regulatory genes (or TFs) mediating the expression of a given gene g_i. A simple strategy is to investigate the set of predecessors of g_i. Accordingly, the notion of a parent set of g_i coined by Barker et al. [34] is the set of genes whose product proteins regulate g_i. For finding the parent sets from a local analysis one can chose from different algorithms. Learning strategy with feedback declare the best scoring sets as parents. Heuristics like hill-climbing, simulated annealing or genetic algorithms can be applied in this step. The child's own level is used to further help determine a child's parent genes. The method considers sequential points of a time series experiment and determines the percentage of time a child gene raises its expression level in a given configuration. The method can potentially better determine activation and repression behavior in cases where there is tight feedback in the network.

The assumption that all input (predecessor) genes of g_i belong to a single module is perhaps too simplistic because there may be several different occasions at which g_i with high expression value is activated, and the predecessor set may hence cover more than one module. A serious drawback of activation/inhibition network regards the fact that for a node with two or more incoming edges, it is not clear if the corresponding genes commonly or independently regulate the target gene. As we have seen from clustering methods, a gene is often regulated by a set of genes rather than by a single gene. Such situation cannot be captured by correlation networks. Circuit diagrams reviewed below are more specific about this point.

Another simple approach is to search for groups of genes with certain common local properties such as hubs. Genes with high regulatory influence on other genes can be isolated from the activation/inhibition graph. These are particularly genes having several out-going connections. Genes with high out-degree are assumed to be members of a common biological process. It is known that modules of commonly regulated genes are in more stable relation than expected [52, 53].

4.3 Circuit diagrams

In contrast to correlation networks which expose pair-wise relationships of genes, circuit diagrams aim to model flows of actions in gene regulatory networks. Again, network nodes represent genes, with weights specifying their current expression status. The status of a node is defined by a function of the status of parent nodes. Update functions may be applied synchronously on all genes, or update delays may be individually specified for the genes for modeling a – perhaps more realistic – asynchronous network behavior.

A simple "classifier" language for describing up- and downregulation was introduced by Soinov et al. [36]. $\uparrow g$ ($\downarrow g$) is used to describe positive (resp. negative) change of the expression level of g. $+g$ ($-g$) means that gene g is 'upregulated resp. downregulated; '$\Leftrightarrow$' is used for simultaneous events, and '$\Rightarrow$' is used to distinguish between events that are divided in time. For instance, $+g_1+g_2 \Leftrightarrow -g_3$ means that g_3 is 'downregulated' while g_1 and g_2 are 'upregulated'; $+g_2 \Rightarrow +g_1$ means that g_1 is 'upregulated' if g_2 was 'upregulated' (for example, in the previous time point of the time series); $\uparrow g_1 \downarrow g_2 \Leftrightarrow \downarrow g_3$ means that positive change in the expression level of g_1 along with simultaneous negative change in expression of g_2 coincides with simultaneous negative change of g_3 expression; $\uparrow g_2 \Rightarrow \downarrow g_1$ means that positive change in expression level of g_2 precedes negative change of g_1 expression.

BioTapestry is a graph tool introduced by Longabaugh et al. [37] for representing regulatory networks (see figure 3). In this tool, a horizontal bar stands for the regulatory region of a gene. The incoming edges (TFs) point to the corresponding binding places. An activating TF ends in an arrow tip, an inhibitory TF in a short bar. The outgoing edge stands for the gene's transcription. A Boolean function on the incoming edges defines the regulating modules.

Boolean networks are known to be successful in modeling real world problems in general, and specifically for genetic regulatory networks. In the simplest case, the expression of gene g_i, as obtained from time series array data, is described as a Boolean value: TRUE representing active (expressed gene), FALSE inactive (not expressed gene) [18, 35, 36]. State of a regulatory network with n entities g_1, …,

g_n is then modeled by a Boolean vector of dimension n. The behavior of each g_i is described by a Boolean function f_i defining the next state from the current states of $g_1, \ldots, g_n$. The Boolean functions are written in disjunctive normal form as an ORed set of AND-clauses of negated or non-negated variables. The binding sites of an AND clause define a *cis*-regulatory module that separately regulates the expression of the gene. The ORed AND-clauses hence define alternative ways of gene regulation. Such model is supported by the observation that a gene can be regulated by different sets of TFs (see, e.g., [3]).

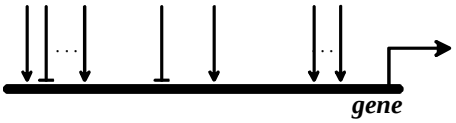


Figure 3: Representation of a gene function in BioTapestry

Multivalued networks. Microarray data often allow finer specification of a gene's state. Accordingly, Banks et al. [20] proposed a generalization of Boolean networks to multi-valued networks: Suppose gene g_i has possible states s_j , $j = 1, \ldots, k_i$. For each state s_j a Boolean variable $b_i[s_j]$ is introduced and is set TRUE if g_i is in state s_j , FALSE otherwise. The next state function is given as a set of Boolean expressions over the variables $b_i[s_j]$. The Boolean expression for $b_i[s_j]$ defines the next state of g_i from the variables $b_{i'}[s_{j'}]$ ($i' = 1, ..., n$; $s_{j'} = 1, ..., k_i$). This means that in different states, the activity of gene g_i is regulated in different ways.

Unfortunately, the microarray data are not precise. Consequently the Boolean functions describing the network structure are burdened with some degree of uncertainty. If the state of a gene g_i is uncertain or unknown, the Boolean network representation has problems. Steggles et al. [18] proposed the introduction of *don't cares*. This uncertainty motivated Zhao et al. [19] to propose the use of random Boolean networks, where the expression values of $f_i(g_1, \ldots, g_n)$ are interpreted as probabilities.

Petri nets have two types of nodes, a set P of places and a set T of transitions, both finite and non-empty. Edges (set E) either lead from places to transitions, or from transitions to places: $E \subseteq (P \times T) \cup (T \times P)$. Places can be marked by tokens. If all places leading to a transition are marked, the transition "fires" by removing the tokens from all incoming places and placing tokens on all outgoing places. There are many variations of the Petri net concept: various token types, more than one token allowed on a place, and timing conditions for transition delays for firing.

When applying Petri nets for GRN modeling, genes are represented as places, and conditions are modeled in the transitions. Gene expression is modeled by a firing condition based on a Boolean function of the incoming values. Steggles et al. [18] describe a transformation from Boolean networks to Petri Nets. There is only one transition, and each place g_i communicates by two arcs: $c(g_i)$ goes from g_i to the transition, and $n(g_i)$ goes from the transition to g_i . The firing condition is defined as a *guard* which is a Boolean expression on the values of incoming places that defines the next state function of the multi-valued network.

Multivalued networks discussed above can also be modeled as high-level Petri nets (HLPN, [20]) which extends the original notion of Petri nets. Again, each entity g_i is modeled as a place in the HLPN, but now tokens of different type reflect the gene states.

Hybrid Petri nets (HPNs, see [21, 22]) are a further extension of Petri nets that allow quantitatively modeling discrete and continuous processes. A HPN [68] has discrete and continuous places and transitions. Each discrete place is associated with a non-negative integer representing the number of tokens. Each continuous place is associated with a non-negative real number called mark. A continuous transition fires continuously, the firing speed is a function of the values of the ingoing places. For each pair (P_i , T_j) representing an arch from a place to a transition, a function defines the weight of the arc, which is a non-negative integer if P_i is discrete and non-negative real otherwise. Likewise, for each pair (T_j, P_i) from a transition to a place, a function defines the weight of the arc, which is a non-

negative integer if P_i is discrete and a non-negative real number otherwise. Each transition T_j is assigned a delay time (if T_j is discrete), and a speed (if T_j is continuous). Initial marking assigns each place an initial non-negative integer. HPNs can be used to model protein and mRNA concentration dynamics which is coupled with discrete switches [23].

Continuous Petri nets can be converted to differential equation systems: Let m_i be the initial marking of the continuous place P_i representing the expression intensity of the corresponding gene g_i . The change rate of m_i , dm_i/dt, is assumed to linearly depend on the concentrations of other proteins,

$$\frac{dm_i}{dt} = \sum_{j=1}^{n} \alpha_j r_j ,$$

where $\alpha_i \in \{-1, 0, 1\}$, and the concentration parameters r_i are in direct relationship with other expression intensities, such as $r_i = k_i * m_1 * m_2$. Here, k_i is the rate of reaction, and the m_j's are the tokens resp. experiment intensities of parent places, see Gilbert et al. [24].

4.4 Hypergraphs

Hypergraphs as generalization of classical undirected graphs are defined as pairs (G, E), where G is the set of vertices and E, the hyperedges, is a set of non-empty subsets of G, $E \subseteq 2^G - \varnothing$. For GRN modeling, G represents the set of genes. Hyperedges can be used to directly model modules of regulatory genes [69, 70, 71]. To capture orientations from cause-effect relationships, Klamt et al. [70] define a directed hypergraph as a pair (G, D) with the set of hyperedges (or hyperarcs) $D \subseteq E \times E$. Each hyperedge is thus a pair of subsets of nodes representing an n:m relationship between the vertices. Such structure nicely models the observed situation that a module of transcription factor genes regulates a set of target genes.

Another concept with apparently interesting modeling features defines a hypergraph as a pair (G, E) where each element of E is a set of nodes with a relation within the set. In the following we have a closer look at this concept and demonstrate its suitability for modeling GNRs. Let $E = \{H_1, H_2, \ldots\}$ be the set of hyperedges. A hyperedge H_k is a pair (S_k, r_k) with a non-empty subset of nodes (genes) $S_k \subseteq G$ and a relation $r_k \subseteq S_k \times S_k$. An oriented correlation edge $(g_i , g_j) \in r_k$ is introduced if g_i directly influences the expression of g_j , without any intermediate expression step. A positive or negative weight $\in \{-1, 0, 1\}$ associated with (g_i , g_j) is deduced, for example, from the correlation matrix W.

To begin with we understand the notion of a hyperedge in a very basic sense and use it for modeling m:n conditions between genes. Accordingly we first constrain hyperedge H_k to a two-level graph $H_k \subseteq I_k \times O_k$ with the interpretation that the genes of the input set I_k commonly regulate the genes of the output set O_k. The elements of $I_k \cap O_k$ are considered as loops of self-regulation. Under this assumption the model is essentially equivalent to that of Klamt et al. [70] and to Boolean networks. To see the latter, let $f_i(g_1, \ldots, g_n)$ be a Boolean function determining the next state of g_i from the current states of $g_1 , \ldots, g_n$. Let f_i be given in disjunctive normal form, i.e., an ORed set of AND-clauses where each clause is a conjunction of literals. Transforming $g_i = f_i(g_1 , \ldots, g_n)$ into the hypergraph model, for each clause $g_{\alpha_1} \wedge \ldots \wedge g_{\alpha_r}$ a hyperedge is introduced, where each g_{α_j} has activating or blocking influence. The complete function $g_i = f_i(g_1 , \ldots, g_n)$ is hence modeled by a set of directed hyperedges, one for each clause. Comparing with the Boolean model, the hypergraph model in fact puts more emphasis on the modularity of regulation conditions. This offers the option of adding expression conditions from cell type specific expression conditions or experimental settings separately for each module resp. clause of a Boolean function.

Without going into detail here, we mention that two-level hyperedges can be combined into increasingly complex structures by linking output nodes from one hyperedge to the input nodes of another. Figure 4 shows the Boolean definition of three functions and corresponding hyperedge representation.

Hypergraphs are therefore expected to be useful for modeling sub-circuits, in particular groups of genes that cooperatively realize special kernel or key functions [3]. It also shows the flow of regulatory information in the network. Hence the idea is to ultimately model complete pathways by higher order hypergraphs.

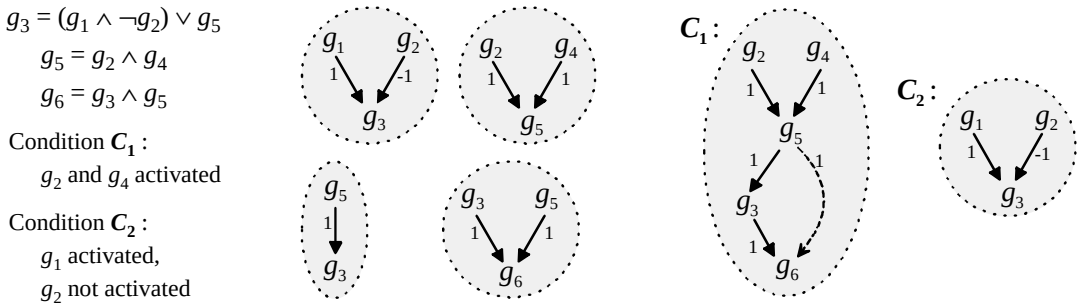


Figure 4: Left: Boolean functions.
Center: Hyperedges representing the Boolean graph.
Right: Hyperedges modeling the pathways in conditions C1 and C2.

5 Conclusion: Summary and Outlook

This paper deals with gene regulatory networks and methods to reveal its coarse structure. An important point concerns network representations because the interpretation of experimental data puts model-dependent emphasis on special structural aspects. The research aims to receive a picture of the gene regulatory network as complete as possible. The GRN may be considered as the skeletal frame along which, depending on the current cell state, i.e., the availability of transcription factors and other signals, gene expression pathways are realized.

It is important to distinguish between GRNs and pathways. Pathways can be considered as avalanche-like gene activities flowing through the network. They can be measured through, for example, microarray experiments, showing snapshots of the pathways at particular points of time. But one has to accept the fact that these experiments only present discrete pictures with many intermediate stages possibly missing. As a consequence, the inferred results have some degree of uncertainty. However, with the help of additional information from other sources, both experimental and computational, the creditability can be raised. These are determination of transcription factors, their binding sites and *cis* - regulatory modules, but also concentration, intensity and disintegration rate of transcription factor proteins, binding forces, expression rates, and other factors. From this data, various computational approaches infer the pathway of gene expressions which after all opens up, albeit small, a window to the GRN structure.

In a reverse view, and assuming that the GRN analysis has reached a sufficiently high degree of accurate completeness, one should be able to use the accumulated knowledge for simulating pathways. Starting from special cell condition with a mixture of TFs, the GRN structure should propose pathways of expressed genes. The hyperedge data structure introduced in this paper should allow simulations for verifying experimentally revealed pathways. In case of a disagreement, hints for correcting and improving the GRN structure could be given. Ultimately such simulations can be expected to predict the effect of drugs and medical applications.

References

1 K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, A. Califano (2005), Reverse engineering of regulatory networks in human B cells. Nature Genetics 37, 382-390.

2 A. Stark, M. F. Lin, P. Kheradpour, J. S. Pedersen, L. Parts, J. W. Carlson, M. A. Crosby, M. D. Rasmussen, S. Roy, A. N. Deoras, J. G. Ruby, J. Brennecke, Harvard FlyBase curators, Berkeley Drosophila Genome Project, E. Hodges, A. S. Hinrichs, A. Caspi, B. Paten, S.-W. Park, M. V. Han, M. L. Maeder, B. J. Polansky, B. E. Robson, St. Aerts, J. van Helden, B. Hassan, D. G. Gilbert, D. A. Eastman, M. Rice, M. Weir, M. W. Hahn, Y. Park, C. N. Dewey, L. Pachter, W. J. Kent, D. Haussler, E. C. Lai, D. P. Bartel, G. J. Hannon, Th. C. Kaufman, M. B. Eisen, A. G. Clark, D. Smith, S. E.

Celniker, W. M. Gelbart, M. Kellis (2007), Discovery of functional elements in 12 Drosophila genomes using evolutionary signatures. Nature 450, 219-232.

3 E. H. Davidson (2006), The Regulatory Genome. Elsevier.

4 M. Elati, P. Neuvial, M. Bolotin-Fukuhar, E. Barillot, F. Radvanyi, C. Rouveirol (2007), LICORN: learning cooperative regulation networks from gene expression data. Biouinformatics 23, 2407-2414.

5 D. di Bernardo, M. Thompson, T. Gardner, S. Chobot, E. Eastwood, A. Wojtovich, S. Elliott, S. Schaus, j. Collins (2005), Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. Nat Biotechnol 23, 377-383.

6 M. Bansal, V. Belcastro, A. Ambesi-Impiombato, D. di Bernardo (2007), How to infer gene networks from expression profiles. Molecular System Biology 3, 78.

7 St. E. Halford, J. F. Marko (2004), How do site-specific DNA-binding proteins find their targets? Nucleic Acids Research, 32, 3040-3052.

8 D. Gilbert, H. Fuß, X. Gu, R. Orton, St. Robinson, V. Vyshemirsky, M. J. Kurth, C. St. Downes, and W. Dubitzky (2006), Computational methodologies for modelling, analysis and simulation of signalling networks. Briefings in Bioinformatics 7, 339-353.

9 L. Diambra (2011) Coarse-grain reconstruction of genetic networks from expression levels. Physica A: Statistical Mechanics and its Applications 390, 2198-2207.

10 M. Hecker, S. Lambeck, S. Toepfer, E. van Someren, R. Guthke (2009), Gene regulatory network inference: Data integration in dynamic models: A review. Biosystems 96, 86-103.

11 A. J. Walhout (2011), What does biologically meaningful mean? A perspective on gene regulatory network validation. Genome Biology 12(4), 109.

12 T. Chen, V. Filkov, St. S. Skiena (1999), Identifying Gene Regulatory Networks from Experimental Data, RECOMB 1999, Lyon, France.

13 J. M. Stuart, E. Segal, D. Koller, S. K. Kim (2003), A gene-coexpression network for global discovery of conserved genetic modules. Science 302, 249–255.

14 J. J. Rice, Y. Tu, G. Stolovitzky (2005), Reconstructing biological networks using conditional correlation analysis. Bioinformatics 21, 765-773.

15 P. D'haeseleer, S. Liang, R. Somogyi (2000), Genetic network inference: from co-expression clustering to reverse engineering. Bioinformatics, 16, 707–726.

16 H. Bolouri, E. H. Davidson (2002), Modeling transcriptional regulatory networks. BioEssays 24, 1118–1129.

17 X. J. Zhou, M.-Ch. J. Kao, H. Huang, A. Wong, J. Nunez-Iglesias, M. Primig, O. M. Aparicio, C. E Finch, T. E. Morgan, W. H. Wong (2005), Functional annotation and network reconstruction through cross-platform integration of microarray data. Nat. Biotechnol., 23, 238–243.

18 L. J. Steggles, R. Banks, O. Shaw, A. Wipat (2007), Qualitatively modelling and analysing genetic regulatory networks: a Petri net approach. Bioinformatics 23, 336–343.

19 W. Zhao, E. Serpedin, E. R. Dougherty (2006), Inferring gene regulatory networks from time series data using the minimum description length principle. Bioinformatics 22, 2129–2135.

20 R. Banks, L. J. Steggles (2007), A High-Level Petri Net Framework for Genetic Regulatory Networks. Journal of Integrative Bioinformatics, 4(3):60, 2007.

21 H. Matsuno, A. Doi, M. Nagasaki, S. Miyano (2000), Hybrid Petri net representation of gene regulatory network. Pacific Symposium on Biocomputing 5, 338-349.

22 R. Vasireddy, S. Biswas (2004), Modeling Gene Regulatory Network in Fission Yeast Cell Cycle Using Hybrid Petri Nets. N.R. Pal et al. (Eds.): ICONIP 2004, LNCS 3316, pp. 1310–1315, 2004. © Springer-Verlag Berlin Heidelberg.

23 C. Chaouiya (2007), Petri net modelling of biological networks. Briefings in Bioinformatics 8, 210-219.

24 D. Gilbert, M. Heiner (2006), An integrative approach for biochemical network analysis. Computer Science Reports 04/05, Brandenburg University of Technology at Cottbus, Dec. 2005.

25 A. V. Werhli, D. Husmeier (2007), Reconstructing gene regulatory networks with Bayesian networks by combining expression data with multiple sources of prior knowledge. Stat. Appl. Genet. Mol. Biol., 6, Article 15.

26 C. J. Needham, J. R. Bradford, A. J. Bulpitt, D. R. Westhead (2007). A primer on learning in Bayesian networks for computational biology. PLoS Comput. Biol. 3 (8), e129.

27 P. J. Park (2009), ChIP-seq: advantages and challenges of a maturing technology. Nature Reviews Genetics 10, 669-680.

28 P. Collas (2010), The current state of chromatin immunoprecipitation. Mol Biotechnol 45, 87-100.

29 H. E. Arda and A. J. M. Walhout (2007), Gene-centered regulatory networks. Briefings in functional genomics 9, 4 -12.

30 A. J. M. Walhout (2006), Unraveling transcription regulatory networks by protein-DNA and protein -protein interaction mapping. Genome Res 16, 1445-1454.

31 F. Geier, J. Timmer, Ch. Fleck (2007), Reconstructing gene-regulatory networks from time series, knock-out data, and prior knowledge. BMC Systems Biology 2007, 1:11.

32 K. Ecker, L. Welch (2009), A concept for ab initio prediction of cis-regulatory modules. In Silico Biology 9, 285-306.

33 Z. Bar-Joseph (2004), Analyzing time series gene expression data. Bioinformatics Review 20, 2493-2503.

34 N. A. Barker, C. J. Myers, H. Kuwahara (2011), Learning genetic regulatory network connectivity from time series data. IEEE/ACM Transactions on Computational Biology and Bioinformatics / IEEE, ACM 8(1):152-65.

35 T. G. Dewey and D. J. Galas (2001), Dynamic Models of Gene Expression and Classification. Functional and Integrative Genomics 1, 269-271.

36 L. A. Soinov, M. A. Krestyaninova, A. Brazma (2003), Towards reconstruction of gene networks from expression data by supervised learning. Genome Biol 4(1), R6.

37 W. J. R. Longabaugh, E. H. Davidson, H. Bolouri (2005), Computational representation of developmental genetic regulatory networks. Developmental Biology 283, 1 – 16.

38 R. Guthke, U. Möller, M. Hoffmann, F. Thies, S. Töpfer (2005), Dynamic network reconstruction from gene expression data applied to immune response during bacterial infection. Bioinformatics 21, 1626-1634.

39 P. D'haeseleer, X. Wen, S. Fuhrman, R. Somogyi (1999), Linear modeling of mRNA expression levels during CNS development and injury. Pac. Symp. Biocomput., 4, 41–52.

40 N. S. Holter, A. Maritan, M. Cieplak, N. V. Fedoroff, J. R. Banavar (2001) Dynamic modeling of gene expression data. Proc. Natl Acad. Sci. USA, 98, 1693–1698.

41 M. K. Yeung, J. Tegner, J. J. Collins (2002), Reverse engineering gene networks using singular value decomposition and robust regression. Proc. Natl Acad. Sci., USA, 99, 6163–6168.

42 E. P. van Someren, L. F. A. Wessels, M. J. T Reinders, E. Backer (2001), Searching for limited connectivity in genetic network models. Proceedings of the International Conference on Systems Biology, Pasadena, CA, pp. 222–230.

43 C. Granger. Testing for causlity: A personal viewpoint. Journal of Economic Dynamics and Control, 2:329–352, 1980.

44 Y. Liu, A. Niculescu-Mizil, A. Lozano, Y Lu (2011), Temporal graphical models for cross-species gene regulatory network discovery. Journal of Bioinformatics and Computational Biology 9(2), 231-250.

45 A. Lozano, N. Abe, Y. Liu, and S. Rosset (2009), Grouped graphical granger modeling for gene expression regulatory networks discovery. In Proceedings of International Conference on Intelligent Systems for Molecular Biology (ISMB-09).

46 C. Rangel, J, Angus, Z. Ghahramani, M. Lioumi, E. Sotheran, A. Gaiba, D. L. Wild, F. Falciani (2004), Modeling T-cell activation using gene expression profiling and state-space models. Bioinformatics 20, 1361-72.

47 O. Hirose, R. Yoshida, S. Imoto, R. Yamaguchi, T. Higuchi, D. S. Charnock-Jones, C. Print, S. Miyano (2008), Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models. Bioinformatics 24, 932-942.

48 S. More, V. Accaamma (2011), Computational methods for the inference of gene regulatory networks. IJCSNS International Journal of Computer Science and Network Security 11, 170-176.

49 de Jong, H. (2002) Modeling and simulation of genetic regulatory systems: a literature review. J. Comp. Biol., 9, 67–103.

50 R. Gupta, A. Stincone, P. Antczak, S. Durant, R. Bicknell, A. Bikfalvi, F. Falciani (2011), A computational framework for gene regulatory network inference that combines multiple methods and datasets. BMC Systems Biology 5, 52.

51 R. Tibshirani (1996), Regression Shrinkage and selection via the Lasso. J Royal Statistical Society, Series B, 58, 267-288.

52 M. Gustafsson, M. Hornquist, A. Lombardi (2005), Constructing and analyzing a large-scale gene-to-gene regulatory network – lasso constrained inference and biological validation. IEEE/ACM Trans. Comput. Biol. Bioinform. 2, 254-261.

53 M. Gustafsson, M. Hörnquist, J. Björkegren, J. Tegnér (2009), Genome-Wide System Analysis Reveals Stable yet Flexible Network Dynamics in Yeast. IET Syst. Biol. 3, 219-228.

54 M. Bansal, G. Della Gatta, D. di Bernardo (2006), Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics 22, 815-822.

55 H.-C. Chen, H.-C. Lee, T.-Y. Lin, W.-H. Li, B.-S. Chen (2004), Quantitative characterization of the transcriptional regulatory network in the yeast cell cycle. Bioinformatics 20, 1914-1927.

56 B. E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, F. d'Alché-Buc (2003), Gene networks inference using dynamic Bayesian networks. Bioinformatics 19 (Suppl. 2), ii138–ii148.

57 F. d'Alche-Buc, P.-J. Lahaye, B.E. Perrin, L. Ralaivola, T. Vujasinovic, A. Mazurie, S. Bottani (2005), A dynamic model of gene regulatory networks based on inertia principle. In: Bioinformatics Using Computational Intelligence Paradigms, Studies in Fuzziness and Soft Computing, Volume 176, 93-117.

58 T. T. Vu, J, Vohradsky (2007), Nonlinear differential equation model for quantification of transcriptional regulation applied to microarray data of Saccharomyces cerevisiae. Nucleic Acids Research 35, 279–287.

59 S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi, M. Tomita (2003), Dynamic modeling of genetic networks using genetic algorithm and S-system. Bioinformatics 19(5), 643–650

60 C. Spieth, N. Hassis, F. Streichert (2006), Comparing mathematical models on the problem of network inference. In: Proceeding of the 8th Annual Conference on Genetic and evolutionary computation (GECCO 2006),Washington, USA, pp. 279–285.

61 M. Kaern, T. C. Elston, W. J. Blake, J. J. Collins (2005), Stochasticity in gene expression: from theories to phenotypes. Nat. Rev. Genet. 6 (6), 451–464.

62 A. Climescu-Haulica, M. D. Quirk (20079, A stochastic differential equation model for transcriptional regulatory networks. BMC Bioinform. 8 (Suppl. 5), S4.

63 N. Friedman, M. Linial, I. Nachman, D. Pe'er (2000), Using bayesian networks to analyze expression data. Journal of Computational Biology 7, 601-620.

64 J. Yu, V. A. Smith, P. P. Wang, A. J. Hartemink, E. D. Jarvis (2004), Advances to Bayesian network inference for generating causal networks from observational biological data. Bioinformatics 20, 3594-3603.

65 K. Sachs, O. Perez, D. Pe'er, D. A. Lauenburger, G. P. Nolan (2005), Causal protein signaling networks derived from multiparameter single-cell data. Science 22, 523-529.

66 S. Bergmann, J. Ihmels, N. Barkai (2003) Similarities and differences in genome-wide expression data of six organisms. PLoS Biol, 2(1), e9.

67 Y. Lu, P. Huggins, Z. Bar-Joseph (2009), Cross species analysis of microarray expression data. Bioinformatics 25, 1476–1483.

68 H. Alla, R. Dravid (1998), Continuous and hybrid Petri nets. Journal of Circuits. Systems and Computers, 8:159-188.

69 S. Klamt, J. Saez-Rodriguez, J. A. Lindquist, L. Simeoni, E. D. Gilles (2006), A methodology for the structural and functional analysis of signaling and regulatory networks. BMC Bioinformatics 7, 56.

70 S. Klamt, U.-U. Haus, F. Theis (2009), Hypergraphs and Cellular Networks. PLoS Comput Biol 5, 5.

71 E. Ramadan, S. Perincheri, D. Tuck (2010), A hyper-graph approach for analyzing transcriptional networks in breast cancer. BCB '10 Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology.

72 X. Yan, M. R. Mehan, Y. Huang, M. S. Waterman, P. S. Yu, X. J. Zhou (2007), A graph-based approach to systematically reconstruct human transcriptional regulatory modules. Bioinformatics 23, ISMB/ECCB 2007, i577–i586.

Protein structure quality assessment and modeling support techniques

M. Antczak [1]*, P. Lukasiak [1,2], K. Fidelis [3],
A. Kryshtafovych[3], J. Blazewicz [1,2]

[1]Institute of Computing Science, Poznan University of Technology,
ul. Piotrowo 2, 60-965 Poznan, Poland
[2]Institute of Bioorganic Chemistry, Polish Academy of Sciences,
ul. Z. Noskowskiego 12/14, 61-704 Poznan, Poland
[3]Protein Structure Prediction Center, Genome and Biomedical Sciences
Facility, University of California, Davis, USA

1 Introduction

Understanding details of machinery of human organism has been a great challenge for humanity. Proteins are the machinery of life as they are involved in all important processes which occur in an organism. In the last decade the number of identified protein sequences gathered in databases increased tremendously but only for the fraction of them the three dimensional structure is known. Determination of a native folded structure of a particular protein is a key to understand its function. Such a determination is difficult and requires time and money consuming experiments such as crystallography or NMR techniques. Hence, prediction of the secrets of protein structure nature using efficient computer aided modeling techniques is of great interest because progress in that area can generate profits in medicine, chemistry.
Moreover, reliable computational methods designed to evaluate protein models quality without knowing native structure is relevant in the context of protein tertiary structure refinement as currently available computational models outnumber experimentally derived.

1.1 Proteins local descriptors library

A protein descriptor is a set of several short, non-overlapping fragments of the polypeptide chain. Each substructure describes local environment of a particular residue and includes only those segments of the main chain that are located in the proximity of that residue (Kryshtafovych et al. 2003). The detailed descriptor construction process description is described in (Hvidsten et al. 2003). Next, descriptors clustering should be conducted relies on grouping many descriptors from different proteins which represent the same structural shape. Finally, we have created a library of protein structures building blocks that are common to

*Corresponding author: maciej.antczak@cs.put.poznan.pl

proteins independent of their global fold. We use the library as the structural context in quality assessment approach.

1.2 Side-chain conformation prediction problem

In order to assess the quality of structural templates in the homology modeling there is need to learn how to thread unknown protein sequence on the known homology protein backbone. To solve this problem one use *SCWRL3*(Canutescu et al. 2003) program which is most popular tool due to its speed, accuracy and ease-of-use for the purpose of homology modeling.

1.3 Molecular potential energy functions

During research one need to select most appropriate potential scoring functions which can be useful for reliable quality assessment with high structural accuracy. Potential energy functions for evaluating protein conformations range from quantum mechanics, which is accurate but very slow, to more heuristic energy functions that include statistical terms. In molecular mechanics potential functions one can distinguished two types of potentials: *"bonded"* and *"non-bonded"*. The bonded energy potentials apply to sets of 2 to 4 atoms that are covalently linked, and they serve to constrain bond length and angles near their equilibrium values and also include a torsional potential that models the periodic energy barriers encountered during bond rotation. The non-bonded energy potentials consist of the *Lennard-Jones function* (LJ) (Brooks et al. 1983) (which includes van der Waals attraction, and repulsion due to orbital overlap), and *Coulomb's law*(Cornell et al. 1995, Duan et al. 2003). Modern molecular mechanics model hydrogen bonds as a combination of an electrostatic interaction and an LJ interaction.
An alternative type of potential energy function is the knowledge-based, or statistical, energy function (e.g. *DFIRE*(Hongyi & Yaoqi 2002), *self-rotamer population energy*(Canutescu et al. 2003)). This type of energy function derives from the database of known protein structures. The advantage of a knowledge-based energy function is that it can model any behavior seen in known protein crystal structures, even if no good physical understanding of the behavior exists. The disadvantage is that these energy functions are phenomenological and can't predict new behaviors absent from the training set.

2 Problem Formulation

Nowadays, general protein structure quality assessment is one of the most important problem in the field of protein analysis which remains unsolved. There is no good protein structure quality assessment methods which work with high accuracy without knowing the corresponding, native protein structure. The most important step in comparative modeling relies on structural template choosing with high structural identity according to unknown protein identified only by a sequence. Main aim of our research is to design and implement a novel protein structures quality assessment method which provides high accuracy estimation.

3 Method

In this paper, we present a protein structure quality assessment and modeling support web framework (PSQAMSF), which allows to identify and visualize possible chemical/physical, folding, packing inconsistencies in order to refine analyzed protein model. It is also particularly suited to assess applicability of the target sequence to structural template alignments, a major source of comparative modeling errors. In presented approach a library of structurally similar local fragments of proteins have been used. It combines several types of potential energies such as statisticala and physics-based potentials. In generally protein structure quality assessment method consists of following phases:

- the descriptor is built for every particular amino-acid from input model and the total of non-bondig atom-atom interaction energy between central residue and other in contact residues is computed for every integrated potential scoring function,
- for every model descriptor the most suitable descriptors group is assigned with using structural similarity function,
- the quality of the model descriptor is measured in the context of assigned descriptor group taking into consideration potentials normal distributions and descriptors comparison features.

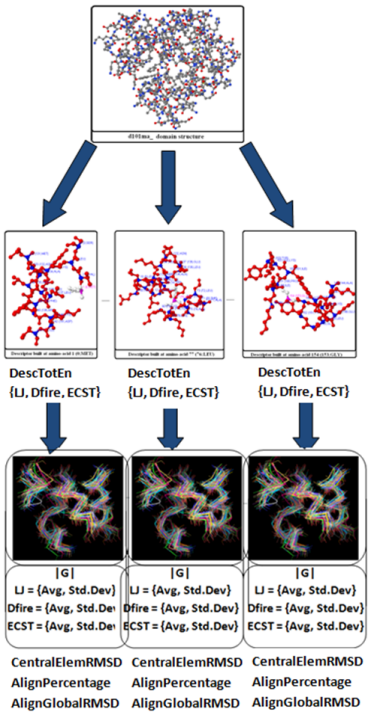


Figure 1: Quality assessment scheme

The measureable results are represented in the table and charts view. Our software allows also to visualize structural interactions which were identified in current model with graph and 2D profile. Additionally our framework has integrated Jmol tool which can be used for visual inspection of possible structural problems of whole molecule, chose descriptor or particular residue-residue interaction.

The PSQAMSF framework provides amino-acids based measurements which generally includes density, descriptors comparison and quality measurements and structurally similar descriptors groups characteristics. The most important measures from above described table are presented with using 2D charts with colored scale. Red color represents possible structural error. In other case the green color represents that local environment around chose amino-acid which is possibly correct.

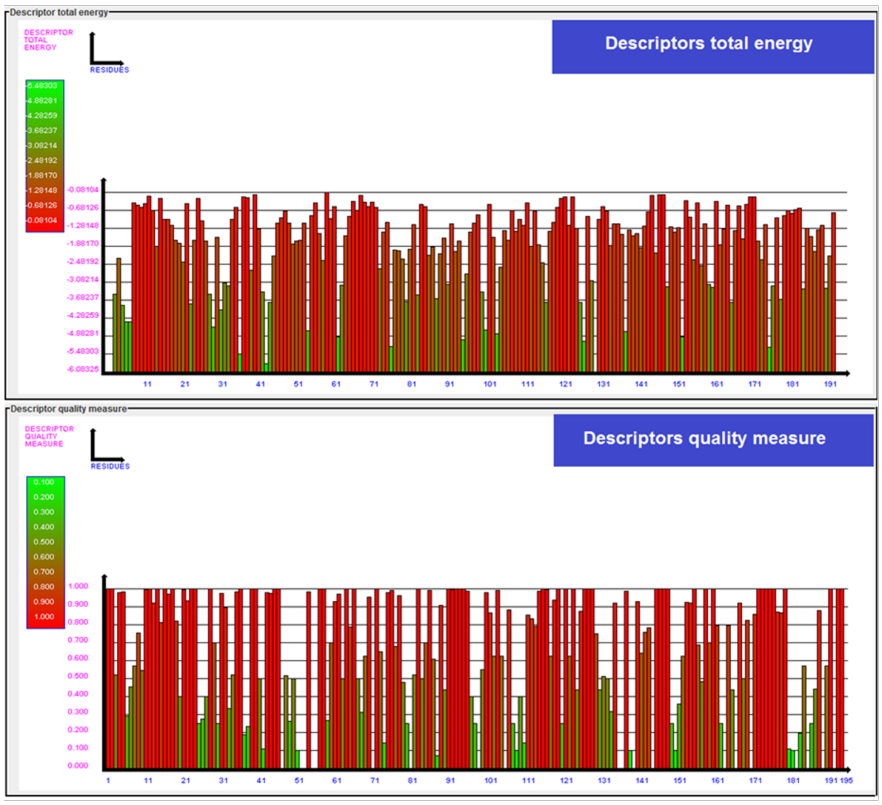


Figure 2: 2D visualization chart examples

The software provides also interactions based measurements table and its visualization called as interactions profile. In the profile every pixel is represented in one of two colors: *orange* where represents identification of non-bonding structural interaction between two residues and *white* in other case.

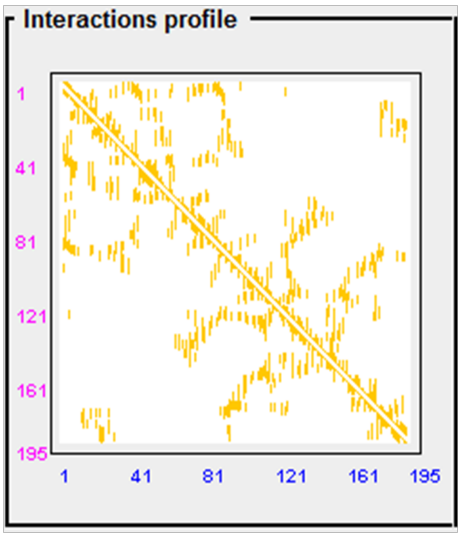


Figure 3: Interactions profile example

Finally the non-bonding structural interactions graph is provided. The user can analyse non-bonding structural interactions graph for three levels of details: whole model, particular amino-acid descriptor and even particular structural interaction. This framework integrates *Jmol*(Herraez 2006) viewer and allows to look at corresponding tertiary structure of protein fragment described in the interactions graph.

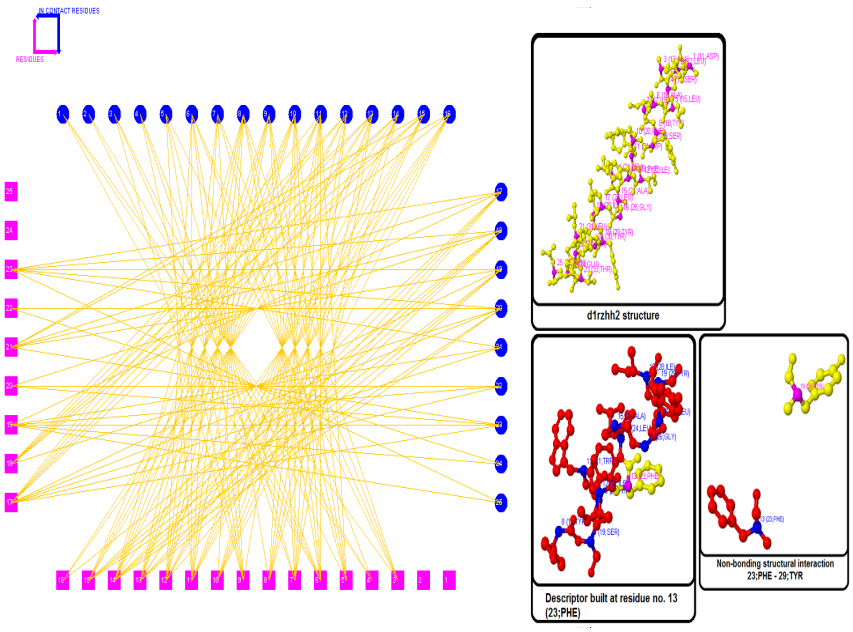


Figure 4: Interactions graph example

4 Conclusion

In this paper a novel general model/structural template quality assessment method has been proposed. Next, the Protein Structure Quality Assessment and Modeling Support Framework (PSQAMSF) has been designed in order to analyze and recognize structural errors in the protein structures. The system provides 2-dimensional plots, where structural neighborhood quality for all model amino-acids is described. The framework allows to visualize amino-acid structural environment which is classified as possibly irregular. This can highly improve an evaluation and refinement of predicted protein models.

References

Brooks, B., Bruccoleri, R., Olafson, B., States, D., Swaminathan, S. & Karplus, M. (1983), 'Charmm: a program for macromolecular energy, minimization, and dynamics calculations', *J. Comput. Chem.* **4**, 187–217.

Canutescu, A. A., Shelenkov, A. A. & Dunbrack, R. L. (2003), 'A graph-theory algorithm for rapid protein side-chain prediction', *Protein Science* pp. 2001–2014.

Cornell, W. D., Cieplak, P., Bayly, C. I., Gould, I. R., Merz, K. M., Ferguson, D. M., Spellmeyer, D. C., Fox, T., Caldwell, J. W. & Kollman, P. A. (1995), 'A second generation force field for the simulation of proteins, nucleic acids, and organic molecules', *J. Am. Chem. Soc.* pp. 5179–5197.

Duan, Y., Wu, C., Chowdhury, S., Lee, M., Xiong, G., Zhang, W., Yang, R., Cieplak, P., Luo, R., Lee, T., Caldwell, J., Wang, J. & Kollman, P. (2003), 'A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations', *J. Comput. Chem.* pp. 1999–2012.

Herraez, A. (2006), 'Biomolecules in the computer: Jmol to the rescue.', *Biochemistry and Molecular Biology Education* **34**, 255–261.

Hongyi, Z. & Yaoqi, Z. (2002), 'Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction', *Protein Science* pp. 2714–2726.

Hvidsten, T. R., Kryshtafovych, A., Komorowski, J. & Fidelis, K. (2003), 'Anovel approach to fold recognition using sequence-derived properties from sets of structurally similar local fragments of proteins', *Bioinformatics* **19**, ii81–ii91.

Kryshtafovych, A., Hvidsten, T. R., Komorowski, J. & Fidelis, K. (2003), 'Fold recognition using sequence fingerprints of protein local substructures', *IEEE Computer Society Bioinformatics Conference* pp. 517–519.

Mind the Gap: Problem Definition and Properties

Maciej Drozdowski*, Dawid Kowalski†,
Jan Mizgajski‡, Dariusz Mokwa§, Grzegorz Pawlak¶

1 Introduction

Mind the gap is a combinatorial optimization problem, which generalizes classic problems like TSP or set cover. It's a graph problem and has some problem specific graph properties. Graphs involved in MTG are a model for metro, train, tram or any other communication network. This type of graph will be called a subway graph.

Each subway graph has nodes corresponding to stations and arcs corresponding to connections between stations. In addition to that, subway graphs have lines. A line is a set of connected nodes. This is the only requirement for lines, but most of them are routes in graph from node A to B and from node B to A. Every node in subway graph must belong to at least one line. Lines can overlap, just like in real life networks, which means that one arc can belong to more than one line. Since lines on communication networks maps are very often denoted by different colors, we will also refer to line as to color.

The question behind the problem is: what is the shortest cycle, starting from a given node, which visits all lines in the graph? Visiting a line is traveling between two stations (nodes) that belong to that line. This means that we can visit multiple lines at once (because lines can overlap).

2 Problem variants

Basic problem consists of finding a cycle that contains a given node. Other version of MTG consists of finding shortest cycle in a graph without determining any nodes that must be visited. This problem can be solved by running MTG optimal algorithm from all nodes in the graph and choosing the solution with shortest cycle. That variant is called universal MTG (uMTG for short).

These two versions can also be distinguished by the type of the graph, which can be either directed (directed or asymmetric MTG) or undirected (undirected

*Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Maciej.Drozdowski@cs.put.poznan.pl
†Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Dawid.Kowalski@skno.cs.put.poznan.pl
‡Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Jan.Mizgajski@skno.cs.put.poznan.pl
§Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Dariusz.Mokwa@skno.cs.put.poznan.pl
¶Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Grzegorz.Pawlak@skno.cs.put.poznan.pl

or symmetric MTG). In most of our work we focus on directed MTG with starting node.

All of above variants can be either a weighted problem, where each edge or arc has its own weight or unweighted where all weights are unitary.

3 Similar problems

During research we came across many problems of nature similar, in some aspect, to the nature of MTG. Among others, these were:

- Eulerian cycle problem
- Chinese postman problem
- Traveling salesman problem
- Rural postman problem
- Covering salesman problem

In Eulerian cycle and Chinese postman problems [2, 4] one needs to visit all edges exactly once and at least once respectively. In MTG there is no requirement about visiting edges. We just need to visit at least one edge from all lines.

Traveling salesman problem involves visiting all nodes in the graph. In MTG we don't need to do that. Some similarity can also be found in rural postman problem [4], where we need to go through all of edges from a given set. But in MTG we have a set of sets of edges and we need to visit at least one edge from each set.

Last one is covering salesman problem [1, 3], where you need to cover all nodes, where cover means that the node must be at most d distance from route. All these problems have simple transformations to MTG problem.

4 Problem properties

4.1 Complexity

We will prove that the decision version of MTG problem is NP-complete and that the original optimization version of MTG is NP-hard. MTG in decision version is in NP because a solution can be encoded as a string of arcs (or edges). The string has length $O(m)$ and the fact of visiting all lines can be verified in time $O(mL)$, where L is the number of lines. We will show a polynomial transformation from undirected Hamiltonian Circiut to MTG.

Undirected Hamiltonian Circuit can be defined as follows [4, 5]:

Given graph $G'(V', E')$ is there a circiut in G' which includes each node exactly once?

For every node $i \in V'$ we create an additional node (twin node i') in transformed instance. Then we connect each pair of original node and it's twin with an edge i, i'. Final step is to put lines into graph. We create one global line, to which every edge belongs. Then we create one line for every pair of original node and it's twin. In such a graph there will be $|V'| + 1$ lines, $2|V'|$ nodes and $|E'| + |V'|$ edges.

We ask if there is a circiut of length at most $3|V'|$ visiting all lines.

If we suppose that the answer to MTG is positive then the distance we need to travel between each original node and it's twin is equal to $2|V'|$. This leaves us with $|V'|$ distance for travels between the orignal nodes. If we managed to visit all lines we've also visited all nodes and if we've done it by traveling $|V'|$ distance it means that the anwser to Hamiltonian Circuit is positive.

If we suppose that the anwser to Hamiltonian Circuit is positive we add a loop (i, i'), (i', i) to visit a line between node and it's twin. Then all lines are visited and the length of the circuit is $3|V'|$.

4.2 Dominance Properties

We can show that in symmetric (undirected) MTG there are optimum solutions in which each edge is traversed at most once in each direction [6].

Below we denote traversing from node u to node v with (u, v) and (v, u) for traversing in the opposite direction.

Suppose that arc (u, v) must be visited many times in the optimum solution. If in between of traverses from u to v and from v to u one would go through a subpath, it would be possible to visit all subpaths first and then go through (u, v). Other traverses are redundant.

If in optimal solution one would go through (u, v) an odd number of times and from node v to node u through different subpaths, it would be possible to go through even subpaths (b, c) in the opposite direction. That is possible, because it's the symmetric MTG. This means that we need to traverse (u, v) just once.

If in optimal solution one would go through (u, v) an even number of times and from node v to node u through different subpaths, it would be possible to go through odd subpaths (a, c) in the opposite direction. This also means that we need to traverse (u, v) just once.

The above covers all possibilities and is applicable to all edges in the subway graph. Such reasoning will reduce number of (u, v) traversals down to one. In asymmetric case an arc (v, u) could be the only one leading from node v to node u. This means that for every arc or subpath going from u to v we need to go through arc (v, u) to get to node v.

4.3 Network compression algorithm

One of the properties of subway graph is that we are able to preprocess instances. We called this preprocessing a network compression. It removes some nodes and edges and since all algorithms complexity bases on one or both of these, instances can be solved quicker. Solution found for compressed instance is the same as solution found for instance not compressed.

The algorithm is fairly simple:

- For every node in graph check if it's:
  - connected to exactly two other stations and
  - all 3 stations belong to the same lines and
  - connections to other stations are in both directions.
  - If all conditions are met then qualify node for removal.

- For every node in graph not qualified for removal:
  - For every node that one is connected to:
    - if it's qualified for removal then disqualify it.
- Remove nodes that are qualified for removal by connecting nodes that the node being removed is connected to. New edge weight will be a sum of weights of removed edges.

The whole idea comes from observation, that if there is a very long route between two intersections we only need to decide if we want to go through that route all the way to the next intersection or step one edge in and get back. This is because whole route contains the same colors and we can visit them all just by traveling between intersection and the first station on the route. Going further in that route makes sense only if we want to get to the next intersection, otherwise it's redundant. This allows us to remove all nodes between the first and the last one. These two must be left in case of situation when we would like to step in, visit all colors and get back.

5 Conclusion

In this paper we introduce properties of Mind the Gap problem. We describe and show multiple variants of the problem. We also present similar problems and describe their similarities and differences. We prove computational hardness and dominance properties of MTG. Finally, we propose an algorithm shortening (compressing) instances by making use of MTG problem properties.

References

[1] Arkin E, Hassin R. Approximation algorithms for the geometric covering salesman problem. Discrete Applied Mathematics 1994; 55(3):197-218.

[2] Black PE. Chinese postman problem. In: Dictionary of Algorithms and Data Structures, U.S. National Institute of Standards and Technology. [online, 23 February 2010], `http://www.nist.gov/dads/HTML/chinesePostman.html`

[3] Current J, Schilling D, The covering salesman problem. Transportation Science 1989; 23(3): 208-213.

[4] Garey MR, Johnson DS, Computers and Intractability: A Guide to the Theory of NP-Completeness, San Francisco: W.H.Freeman and Co.; 1979.

[5] Karp R. Reducibility Among Combinatorial Problems. In: Miller RE, Thatcher JW, editors. Complexity of Computer Computations, New York: Plenum Press, 1972, p. 85103

[6] Drozdowski M, Kowalski D, Mizgajski J, Mokwa D, Pawlak G, Mind the Gap: A Study of Tube Tour, submitted.

Mind the Gap: Solutions and Algorithms Analysis

Maciej Drozdowski*, Dawid Kowalski†,
Jan Mizgajski‡, Dariusz Mokwa§, Grzegorz Pawlak¶

1 Introduction

Mind The Gap (MTG) was first introduced in [1] and is the problem of finding a minimum cycle in a subway graph G visiting all lines (connected sub-graphs of G) embedded in this graph. By visiting a line we understand traveling between two nodes connected by an edge belonging to this line. Since lines can overlap it is possible to visit many lines when traversing a single edge.

MTG comes in many variations including:

- Sourced MTG - in which a proposed solution must include (start at) a certain node labeled as the **source**.
- Universal MTG - in which we relax the above-mentioned constraint and look for a globally shortest cycle
- Directed (asymmetric) MTG - with a directed subway graph.
- Undirected (symmetric) MTG - with an undirected subway graph.

In this paper we will discuss both Sourced (referred to as MTG for simplicity) and Universal MTG (uMTG) but only for undirected subway graphs. It is noteworthy that uMTG can be solved with an algorithm designed for MTG by taking each node of the subway graph as source.

*Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Maciej.Drozdowski@cs.put.poznan.pl
†Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Dawid.Kowalski@skno.cs.put.poznan.pl
‡Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Jan.Mizgajski@skno.cs.put.poznan.pl
§Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Dariusz.Mokwa@skno.cs.put.poznan.pl
¶Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland - E-Mail: Grzegorz.Pawlak@skno.cs.put.poznan.pl

2 Sample solution

We present a sample MTG solution for London Tube [5] having Oval station for the source node and assuming unit edge lengths. This cycle also happens to be a feasible solution for the uMTG version of the problem. In Fig. 1 we present a section of London Tube map with graphical depiction of the MTG solution drawn with dark-red markers and arrows. In Fig. 2 we show in a step-by-step manner how does the MTG solution visit lines of the subway graph.

Figure 1: MTG solution for London subway network starting from Oval

Edge	New lines visited on this edge
Oval → Kennington	★
Kennington → Waterloo	
Waterloo → Bank	★
Bank → Shadwell	☆
Shadwell → Bank	
Bank → Liverpool Street	★
Liverpool Street → Moorgate	★★★
Moorgate → Barbican	
Barbican → Farringdon	
Farringdon → St. Pancras	

St. Pancras → Euston	
Euston → South Hampstead	☆
South Hampstead → Euston	
Euston → Warren Street	
Warren Street → Oxford Circus	★
Oxford Circus → Piccadilly Circus	★
Piccadilly Circus → Green Park	★
Green Park → Westminster	★
Westminster → Embankment	
Embankment → Waterloo	
Waterloo → Kennington	
Kennington → Oval	

Figure 2: Step by step explanation of the MTG solution for London network

The second and fourth column in Fig. 2 depicts what lines are visited by each edge *for the first time* in the solution. Lines that were visited earlier in the cycle are not taken into consideration - for example, the edge Kennington → Waterloo belongs to the black line, but this line was visited earlier at edge Oval → Kennington, so it does not appear in second column. Other important moments in the cycle are:

- The Liverpool Street → Moorgate edge in which many lines are visited while traversing a single edge
- Reaching the Westminster station - a this point the solution has visited all lines and has to return to the source. For all algorithms based on the principle of incremental path-building, visiting all lines means that the next step is to return to the source via the shortest route.

3 Optimization Algorithms

Double Recursion: Double recursion algorithm is a branch and bound (B&B) algorithm which recursively enumerates permutations of colors, and routes between the pairs of colors.

The first recursion is an enumeration of possible color permutations. Suppose X, Y are two consecutive colors in the permutation. The second recursion is enumerating all shortest routes between any station in line X, and any edge in line Y. Furthermore if on a path induced by the constructed permutation line Z is visited "by the way" this path is eliminated in order to disallow isomorphic optimal solutions. On each stage of path construction, the sum of the length of the current path and shortest route to the source is treated as Lower Bound.

DR gives optimum solutions at the expense of exponential running time dependent on both the number of lines and the number of nodes and edges in the graph.

All Cycles BB: All Cycles (AC) is based on the dominance property of the subway graph for symmetric MTG. It states: *For a symmetric subway graph there exist an optimal solutions for which each edge can be traversed at most once in each direction*[1].

For AC each edge in the subway graph is represented as a pair of arcs and, in accordance with the Dominance Property, each arc can be traversed at most once. Cycles are constructed by a depth-first-search method. Similarly to Double Recursion on each stage of path construction, the sum of the length of the current path and shortest route to the source is treated as Lower Bound for current solution.

The main advantage of AC over Double Recursion is that it's computational complexity is independent of the number of lines.

4 Heuristic algorithms

Single Recursion: Single Recursion can be viewed as a simplified Double Recursion. Simplification is achieved by taking the shortest path between two lines of a permutation instead of enumerating all possible paths. Single Recursion has no optimality guarantee.

Closest Unvisited Line: Closest Unvisited Line (CUL) is a greedy algorithm iteratively choosing a closest edge containing a yet unvisited line. When all lines are visited, the algorithm returns to source via the shortest path.

Closest Most Efficient Edge: Closest Most Efficient Edge (CMME) can be viewed as an upgrade of CUL. Instead of traveling to a closest edge containing an unvisited line, it chooses an edge with the best ratio of unvisited lines to the cost of traversing it. It follows that CMEE prefers edges with overlapping lines.

5 Experiments and algorithms analysis

All aforementioned algorithms were studied both theoretically and experimentally. The theoretical study of algorithms included defining their computational complexities and introducing lower bounds of worst case behavior for the heuristics.

The experimental study included solving instances based on 12 real-life communication networks with random source nodes (120 instances in total). We achieved encouraging results with most instances solved to optimality within a 4 hour time limit.

In general Double Recursion outperformed AC both in execution time and quality of solutions. AC often exceeded time limit and was terminated prematurely. Single recursion delivered good results although it's typical execution time was 3 orders of magnitude bigger than those of CUL and CMEE. In most cases CMEE dominated CUL in terms of quality of results[1].

Additional experiments included finding uMTG solutions for all studied networks.

6 Conclusions

In conclusion, MTG is a new and challenging problem of combinatorial optimization. Despite apparent similarity to other well known problems in the field [2, 3, 4, 6], it is essentially different by the fact that its objectives can only be achieved by traversing an edge in the graph. This results in a dynamic view of the subway graph and demands algorithms assuming a different perspective than traditional approaches used in many other transportation problems. Although satisfactory results were achieved for real-life instances, further study is needed to asses the performance of proposed algorithms on different classes of subway graphs.

References

[1] Drozdowski M, Kowalski D, Mizgajski J, Mokwa D, Pawlak G, Mind the Gap: A Study of Tube Tour, submitted for publication.

[2] Arkin E, Hassin R. Approximation algorithms for the geometric covering salesman problem. Discrete Applied Mathematics 1994; 55(3):197-218.

[3] Current J, Schilling D, The covering salesman problem. Transportation Science 1989; 23(3): 208-213.

[4] Eulero L, Solutio problematis ad geometriam situs. Commentarii Academiae Scientiarum Petropolitanae 1741; 8: 128-140. [online, 23 February 2011], http://www.math.dartmouth.edu/~euler/docs/originals/E053.pdf

[5] Transport for London. [online, 23 February 2011]. http://www.tfl.gov.uk/

[6] Black PE. Chinese postman problem. In: Dictionary of Algorithms and Data Structures, U.S. National Institute of Standards and Technology. [online, 23 February 2010], http://www.nist.gov/dads/HTML/chinesePostman.html

The software development for optimization of production technology for selected energy crops

Anna Gotfryd [*], Grzegorz Pawlak [†], Wojciech Wojciechowicz [‡]

Acknowledgments: The work has been partially supported by the European Fund for Regional Development within the framework of the Operation Programme "Innovative Economy", 2007-2013, project no. POIG.01.03.01-00-132/08-00.

1 Introduction

With the present situation of increasing energy demand, rising energy prices, and reinforcement of countermeasures for global warming, the interest in renewable energy sources is continuously growing. In the context of new European Union directives requiring Poland to increase the share of energy from renewable sources in overall energy balance of the country, there is a need for significant amounts of biofuels [1, 3]. As a result of this bio-economic situation, the research project on the "Development of a species index and optimization of production technology for selected energy crops" is being conducted. The entire project is realized within a consortium representing four scientific institutions, i.e. the Poznan University of Life Sciences, the Institute of Soil Science and Plant Cultivation in Pulawy, the Lodz University of Technology and the Poznan University of Technology. A comprehensive approach to production technology of crops to be used for energy purposes is provided by the realization of three basic modules:

- Module I: Optimization of quality parameters of cultivars/strains of cereals, sugar beets, maize and sorghum from the point of view of production of biofuel and biogas.
- Module II: Optimization of cultivation methods for cereals, sugar beets, maize, sorghum, Virginia fanpetals and reed Canary-grass to be used for biofuel and biogas in terms of variation in soil and climatic conditions.
- Module III : Determination of prospects for cultivation of energy crops in Poland and determination of profitability of production and energy balance for investigated crops and proposed technologies [4].

[*]Institute of Computing Science, Poznań University of Technology, ul. Piotrowo 2 60-965 Poznań, Poland E-Mail: Anna.Gotfryd@cs.put.poznan.pl

[†]Institute of Computing Science, Poznań University of Technology, ul. Piotrowo 2 60-965 Poznań, Poland E-Mail: Grzegorz.Pawlak@cs.put.poznan.pl

[‡]Institute of Computing Science, Poznań University of Technology, ul. Piotrowo 2 60-965 Poznań, Poland E-Mail: Wojciech Wojciechowicz@cs.put.poznan.pl

The determination of the prospects such as the relation of plant productivity to the production costs may be a difficult task for an ordinary farmer. In a consequence, some of them may abandon the idea of founding a plantation of energy plants for the wrong reasons. On the other hand the others may plant the energy plants without any analysis of the impact for the natural environment. That is why an innovative computer program to determine optimal production technology of tested crops, taking into consideration climatic and soil conditions, ecological and economic aspects is being developed. The software will help the farmer with choosing the most profitable technology for energy crops production thanks to reliable estimation of profitability for initially specified conditions and thanks to suggestions optimizing its cultivation - both in terms of cost effectiveness and environmental friendliness, which as a result will reduce the risk of the investment. The general specifications of functionalities regard three levels:

- Level 1 - the software will use and process data obtained from the Institute of Soil Science and Plant Cultivation within:
  - the level and timing of fertilization
  - the type and quality of soil,
  - the nutrient content in soil,
  - the amount and distribution of precipitations and temperature
  - the level of organic fertilization (dose and timing of applications)

After gathering this data, a database is being created for storing and creative use of the results. A module that optimizes and supports decision is being designed and manufactured. The external databases can be used as well.

- Level 2 - the module evaluating the environmental friendliness is being designed and manufactured; in the absence of positive evaluation of the farming unit, the technology will not be accepted and appropriate changes will be suggested.
- Level 3 - the module will also evaluate the economic aspect of the farm unit - it will calculate the production profits on the basis of information about the hardware base of farm unit.

In other words the application shall provide three main functionalities - gathering and processing the input data, assessment of the environmental friendliness of technology, evaluation of the economical factor [2].

2 The prototype of BIOPOWER application

During the pre-project analysis, it was decided that the BIOPOWER tool should be implemented as a web application. It facilitates the application management, in particular the introduction of changes and updates, as well as the process of acquiring data from external sources. This data is stored on the server administrated by Service Provider which facilitates the aggregation of data and statistics' generation, etc. The web application is of a great convenience for end-users. It gives them the ability to access information from any computer, requires no

software installation and the user's data is independent of the machine on which the person logs in . The user registers on the site. After creating an account the user must be logged in to access the more advanced functionalities. Registered users can proceed to fill in the data regarding the farm characteristics. Then, on this basis, the user creates a list of farming activities. The next step is to create a technology chart - defined by linking the list of activities with a list of fields on which those activities will be performed. The technology cards are essentials for cost calculations.

3 Architecture of the application

Due to the research project characterization, thus relatively long time frames and unstable user needs the authors decided to use the spiral software development model [5]. The Figure 1 presents the main spiral model approach:

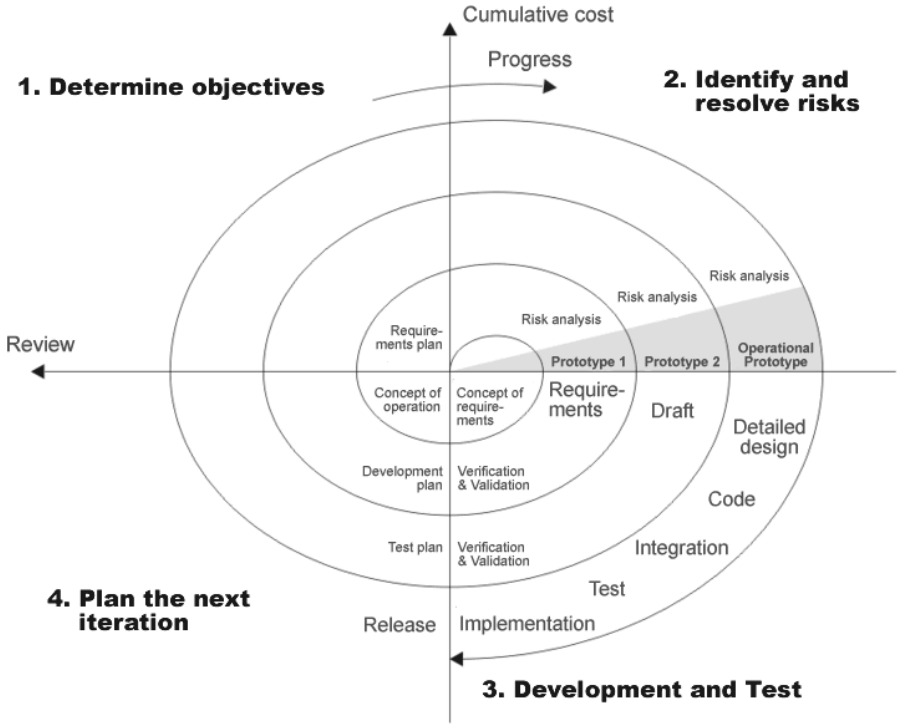


Figure 1: Spiral model
source: http://en.wikipedia.org/wiki/File:Spiral_model_(Boehm,_1988).png

Thus, in the first step the initial prototype has been build. Current prototype is focused on the first functionality, thus on gathering and processing the input data. The range of functionalities include user authentication and authorization mechanisms based on the RBAC principle, gathering and storing information from the user (incl. data regarding homestead, machines and plants), development and fulfillment of process sheet and estimation of unit costs. This version has been released and a feedback from potential end-users

has been collected. Also white-box and black-box tests have been executed; these activities were performed by independent entities, thus not engaged in the application development process. This ensured objectiveness of the developers' point of view, thus different comments and remarks were documented. After merging with developer's remarks a comprehensive feedback (to be used when designing and developing the next prototype version) was gathered. The version BIOPOWER prototype v.2.0 under development at present is based on the ASP.NET MVC 3 technology. The MVC DESIGN PATTERN ensure that the model (data management), view (user interface) and controller (intermediate in communication between model and view; provides a set of function for using models) are separated. In the BIOPOWER prototype, the models are representing main user assets (i.e. homestead, plants, machines and animals), views are web pages enabling user to execute operations on models. The logic behind managing models is implemented in controllers. The MVC have been used as application framework for the BIOPOWER prototype. The project has been organized using aspect approach; thus each component (i.e. managing user data, creating process sheet) is separated from the rest of application, so it has an own set of models, views and controllers. The ORM (object - relational mapping) tool "Entity Framework" provided by Microsoft have been used to enable the access to data. The object-relational mapping ensures the data compliance between the application and database; the tables in database can be used as classes with relative variables. In the BIOPOWER prototype, the user data is stored in Oracle Database. The cooperation with Entity Framework is provided by Oracle connector.

4 Conclusions

While elaborating application prototypes (currently 2.0 version), it was essential to harmonize the approaches to a single and coherent concept. Development of the project specification requirements included such items as: the use-cases diagram, and nonfunctional requirements and database schema. The choice of three-layer architecture allows easy modification of the application at any level of its development by exchanging its individual components. The conception of the system will contribute to a reliable estimation of profits from energy plants cultivation.

References

[1] Roszkowski A. *Efektywnosc energetyczna roznych sposobow produkcji i wykorzystania biomasy [Energy efficiency of different production methods and utilization of biomass]. In: Uprawa roslin energetycznych a wykorzystanie rolniczej przestrzeni produkcyjnej w Polsce [Growing of energy crops and use of agricultural production area in Poland].* IUNiG PIB Pulawy, pp.30-41, 2008.

[2] Szczutowska A. *Analiza przedprojektowa w celu okreslenia zalozen do utworzenia pakietu oprogramowania (opis procesu produkcji). [Pre-analysis to determine the assumptions to create a software package (description of the production process)]* Technical notes, 2010.

[3] Wajszczuk K., Baum R., Wielicki W. *A proposal of a logistics model for the use of biomass for energy for local communities within the concept of sustainable rural development.* 107-th Seminar of European Association of Agricultural Economists: "Modeling Agricultural and rural Development Policies" Sevilla, Spain: 29-th January - 1-st February 2008. Paper published on a CD ISBN 978-92-79-08068-5 and available on the Internet: www.eaaeseminar.es, 2008.

[4] Wajszczuk K. *Development of a species index and optimization of production technology for selected energy crops for the Polish conditions.* Technical notes, 2011

[5] Boehm B. *Spiral Development: Experience, Principles, and Refinements* Spiral Development Workshop February 9, 2000,

Buffer Management in Car Sequencing Problem

Grzegorz Pawlak, *Wojciech Wojciechowicz †

1 Introduction

In this paper we evaluate the sequence quality issue in the car sequencing problem [1]. This work was inspired by the encountered in the real production lines where the process stability (thus sequence quality) is a key issue for applying the JIS (Just In Sequence) policy. Thanks to using JIS, it is possible to reduce the inventories level(thus amount of resources frozen in factory) and still to satisfy customer demand [2]. In the article, the model of the production area (based on the actual car factory layout) was proposed. The sequence quality coefficient (PKG) was applied as a goal function to estimate the lines and buffers behavior in the production process. As a result, the production process was described, the above problem was formalized and the solutions improving the process stability were proposed.

2 Formal definition

In the considered system, there are two assembly lines ($Line1$ and $Line2$) in parallel. On the first one only car sides are produced, while floors are on the second one. Both lines are independent; i.e. the cycle times can varies, each line can be stopped in any point of time. The $Buffer1$ and $Buffer2$ have sizes $b1$ and $b2$ respectively. Both buffers are FIFO type, thus no resequencing is possible there.
The quality control is deterministic, thus at the beginning of planning horizon the cars to be controlled are specified. For selected vehicle the quality control is conducted on both lines; thus both sides and floor are controlled. Two types of quality control can be distinguished - a long quality control (during which a part is taken from a line for exactly q_{cl-l} time units) and a short one (when a part is taken from a line for exactly q_{cl-s} time units).
The car bodies from Line1 and Line2 are being stored in $Buffer3$. After coupling (since only coupled cars are allowed to) a car can leave the $Buffer3$. This process requires presence of both sides and floor for particular vehicle in $Buffer3$ at once. Since there are freely access to each part in $Buffer3$, there are no further restrictions (except on stated above) on leaving parts from the $Buffer3$. The figure 1 presents the considered system.
Let's denote:

*Institute of Computing Science, Poznan University of Technology, ul. Piotrowo 2 60-965 Poznan, Poland E-Mail: Grzegorz.Pawlak@cs.put.poznan.pl

†Institute of Computing Science, Poznan University of Technology, ul. Piotrowo 2 60-965 Poznan, Poland E-Mail: Wojciech.Wojciechowicz@cs.put.poznan.pl

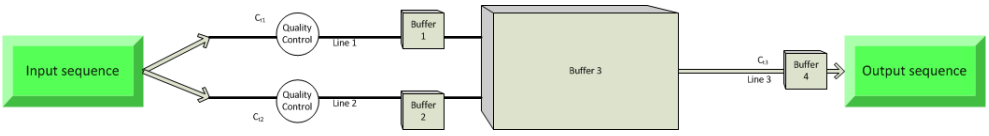


Figure 1: System model

- X - decision space (including all decision in the considered system),
- Y - evaluation space,
- $f : X \rightarrow Y$ - objective function,
- Ω - set of feasible solutions,
- K - set of feasible solutions grades,

where:

- $K = \ y \in Y : y = f(x), x \in \Omega$,
- D - dominance relation ($D \subset Y \times Y$),
- K^* - set of the best (according to the dominance relation) grades ($D(K^* \in K)$),
- Ω^* set of optimal solutions, ($\Omega^* = x \in \Omega : f(x) \in K^*$),

Let's introduce an ordered sixth describing a car $s_i = (h_i, j_i, k_i, q_{cli}, q_{csi}, m_i)$, where:

- h_i - part $i-th$ position in a input sequence, thus at I,
- j_i - part $i-th$ position at a sequence in $Buffer1$ and $Buffer2$, thus after the quality control,
- k_i - part $i-th$ position at a output sequence, thus at O,
- q_{cli} - indication, if the part i is selected for the long quality control, where:

$$q_{cli} = \begin{cases} 1 & - \ if \ s_i \in Q_{cl}, \\ 0 & - \ otherwise. \end{cases} \tag{1}$$

- q_{csi} - indication, if the part i is selected for the short quality control, where:

$$q_{csi} = \begin{cases} 1 & - \ if \ s_i \in Q_{cs}, \\ 0 & - \ otherwise. \end{cases} \tag{2}$$

- m - the type of car (in case the mixed model constraints are present).

Let's focus on the $Buffer3$, since all crucial decisions in the system (including re-sequencing, $Line1$ stoppage, $Line2$ stoppage) are to be made on a basis of information from $Buffer3$.

Let's consider a matrix $C = (c_{ji})m \times n$, where:

- m - a space at $Buffer3$,
- n - a part to be assembly,
- x_{ji} - possibility to use the space j at the Buffer3 for the car i,
- c_{ji} - the cost of using the space j at the Buffer3 for the car i.

According to our optimization criteria, let's define the cost as:

$$c_{ji} = \begin{cases} 1 \ - \ if \ the \ part \ i \ is \ late, \\ 0 \ - \ if \ the \ part \ i \ is \ not \ late, \end{cases} \tag{3}$$

Let's assume, that $n <= m$ and determine a matrix:

$$x^* = (x^*_{ji}) \ m \times n, x^*_{ji} \in \Omega, \tag{4}$$

where:

$$f(x^*) = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ji} x^*_{ji} = \min_{x \in \Omega} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ji} x_{ji} \tag{5}$$

thus

$$\Omega = \begin{cases} \sum\limits_{i=1}^{m} x_{ji} = 1, j = 1, ..., n \\ \sum\limits_{j=1}^{n} x_{ji} <= 1, i = 1, ..., m \end{cases} \tag{6}$$

where:

$$x_{ji} = \begin{cases} 1 \ - \ if \ the \ car \ body \ s_i \ is \ assigned \ to \ the \ b3_j \ place \ at \ buffer3, \\ 0 \ - \ if \ the \ car \ body \ s_i \ is \ not \ assigned \ to \ the \ b3_j \ place \ at \ buffer3, \end{cases} \tag{7}$$

Due to constraint 7 each part is assigned to unique space at $Buffer3$, and each space at $Buffer3$ is used at most once.

The constraint 8 ensures that parts are not moved further than it is possible with given buffer size.

$$\forall_i \ k_i - j_i <= b_3 - 1 \tag{8}$$

The system can be described using following formulas. The i-th car acceleration (namely a_{i-ql}) caused by long quality control performed on other cars is determined by the equation:

$$a_{i-ql} = \sum_{x=j_i}^{q_{cl}-l+j_i} s_i \forall s_i \in Q_{cl} \tag{9}$$

similar rule can be formulated for the short quality control; let's denote the offset caused by short quality control by a_{i-qs}

$$a_{i-qs} = \sum_{x=j_i}^{q_{cl}-l+j_i} s_i \forall s_i \in Q_{cs} \tag{10}$$

A car, by being controlled, can also be delayed; thus we determine the delay d_{i-ql} caused by the long quality control for the i-th car:

$$d_{i-ql} = q_{cl} \forall i \in Q_{cl} \tag{11}$$

similar rule can be formulated for the short quality control; the delay d_{i-qs} caused by the short quality control for the $i-th$ car:

$$d_{i-qs} = q_{cs} \forall i \in Q_{cs} \tag{12}$$

thus the car s_i position after the quality control is:

$$h_i = j_i - a_{i-qs} - a_{i-ql} + d_{i-qs} - d_{i-ql} \tag{13}$$

as a result, the sequence after quality control displacement can be determined using an equation:

$$j_i = h_i - q_{cl} * q_{cli} - q_{cs} * q_{csi} + \sum_{z=i+1}^{i+q_{cl}} q_{clz} + \sum_{u=i+1}^{i+q_{cs}} q_{csu} \tag{14}$$

3 Sequence quality measurement

In hereby article, the sequence quality will be measured using the PKG (ger. Perlenkettegte) rate. This factor is defined as:

$$PKG = \frac{card(P) - card(P_l) - card(P_o)}{card(P)} \cdot 100\% \tag{15}$$

where

- P is a set of car bodies in the considered window,
- P_l is a set of late car bodies in the considered window (inc. car bodies, which originally were outside the window but due to the delay are inside the consider window),
- P_o is a set of car bodies, which steps over the considered window,

The PKG rate can have values from -100 (when each body car steps out of the window, and their places have been taken by late car bodies from the previous window) up to 100 when no car body is late.
The reason for choosing such measure is the fact that PKG is currently used by the car manufacturers. That is why we decided to take it into consideration in this paper.

In order to compute the PKG factor, we need to determine two following sets:

- P_l,
- P_o,

The set P is given on input.
Thus

$$P_l = \{S_i | k_i < h_i \; and \; h_i \in P\}, \tag{16}$$

and

$$P_o = \{S_i | k_i < h_i \; and \; h_i > \max P\}, \tag{17}$$

Since the re-sequencing can be performed at $Buffer3$, this activity shall be taken into account to compute the PKG between the planned and realised sequence. For that reason, let's denote as P_r the set of cars, which position can be restored in the $Buffer3$.
Thus

$$P_r = \{S_i | S_i \in P_l \; \cup S_i \in P_o \; and \; b_3 > j_i - h_i\} \tag{18}$$

As the result, the PKG (incl. re-sequencing at $Buffer3$) can be computed using the equation:

$$PKG = \frac{card(P) - card(P_l) - card(P_o) - card(P_r)}{card(P)} \cdot 100\% \tag{19}$$

4 Quality control

In the considered system, there are two quality control stations, namely q_{c1} and q_{c2}. On each station, two types of quality controls can be performed - long quality control, during which a part is comprehensively examined; and short quality control when only selected tests are performed. The quality control is deterministic, thus at the beginning of planning horizon, the sets of cars for testing are determined. The time needed for performing quality controls is also given in advance.

The interferences caused by quality control, resulting in sequence quality degradation can be repaired in the $Buffer3$, since any body/floor pair can be released on the $Line3$ from the $Buffer3$.

Let's assume that the full quality control, needs more than TU in comparison with the short quality control time. Then for the reconstruction of the schedule, it is sufficient that

$$b_3 > q_{cl-l}. \tag{20}$$

Let's consider an example. The initial sequence is

1	2	3	**4**	**5**

the item **4** is to be shortly controlled, and **5** is to be fully controlled. The
$q_{cl-s} = 2$
and
$q_{cl-l} = 3$

Thus, the long quality control will firstly take place, and will delay the part 5 by a 3 time units; the sequence will be changed to:

1	**5**	2	3	4

as a result, the PKG factor will be diminished to the value of 80.

Afterwards the short quality control will occur. Thus, by delaying the part 4 by a 2 time units the sequence will be changed to:

1	5	**4**	2	3

and the PKG factor will be further degradated to the value of 60. Let's focus on particular parts:

- part #1 - neither delayed nor accelerated,
- part #5 - delayed by 3 time slots,
- part #4 - delayed by 1 time slot,
- part #2 - accelerated by 2 time slots,
- part #3 - accelerated by 2 time slots.

Since a re-sequencing can be performed only in the $Buffer3$, we shall focus on this module. As there is freely access to any part, any car for which both upper body and chassis that exist in the $Buffer3$ can be assembled, thus directed to the output. So, for the example considered above, let's use the $b3 = 4$. Thus, we are able to fit at once each item with changed position by the quality control, as presented in the figure below.

5	4	2	3

The part #1 is not needed on the Buffer3 at this point, since it's position is not changed. As a result, the part #5 can be directed to the output; thus the Buffer3 will contain

1	4	2	3

Afterward, the parts #4, #3, #2 and #1 can be directed to the output; as a result the sequence

1	2	3	4	5

is reconstructed. Since no part is delayed, the $PKG = 100$. It is worth to note, that if size of $Buffer3$ is not larger than the smallest delay caused by the quality control, the PKG ratio can not be increased. In that case, we can not move the part to its initial position; we can decrease the delay, but not eliminate it completely. Since in the PKG factor we consider the delay as a binary value (part can be delayed or not), it does not matter how big the delay is.

5 Conclusion

In this work the sequence quality problem in the CSP was evaluated. Based on the industrial factory layout, the simplified model was proposed and the problem formalized. As a result, solutions to maximize the sequence quality coefficient factor were proposed. This work pointed out also several directions for further research, including developing more complex system (including several production lines) and proposing new sequence quality coefficient.

References

[1] A. Scholl. *Balancing and Sequencing of Assembly Lines.* Physica-Verlag, 1995.

[2] N. Boysen, M. Fliedner, A. Scholl. *Sequencing mixed - model assembly lines: Survey, classification and model critique.* European Journal of Operational Research vol. 192, 2009.

Advanced multi-item Internet shopping

Jacek Blazewicz, Jedrzej Musial

Institute of Computing Science, Poznan University of Technology, ul. Piotrowo 2, 60-965 Poznan, Poland

Corresponding author - Jedrzej.Musial@cs.put.poznan.pl

Key Words: Internet shopping, computational complexity, optimization, algorithms.

1 Problem definition

We study an optimization aspect of Internet shopping with price sensitive discounts from customer perspective (which is a specific case of the Internet Shopping Optimization Problem [1, 5]). Specifically, we consider a problem in which a customer would like to buy products of a given set $N = \{1, \ldots, n\}$ in a given set of Internet shops $M = \{1, \ldots, m\}$ at the minimum total final price. There are the following given parameters and decision variables:

d_i - delivery price of all products from shop i to the customer;

p_{ij} - standard price of product j in shop i, $p_{ij} = p_j$ if standard prices of product j are the same in all shops;

N_i - subset of products of the set N in shop i (eligible products for shop i), $N_i \subseteq N$;

M_j - subset of shops in which product j can be bought (eligible shops for product j), $M_j \subseteq M$;

S_i - subset of products selected by the customer in shop i (basket of shop i, decision variable), $N = \cup_{i=1}^{m} S_i$ and $S_i \cap S_j = \phi$, $i \neq j$, for a feasible solution;

$T_i(S_i) = d_i + \sum_{j \in S_i} p_{ij}$ - total delivery and standard price in shop i for a given set of products $S_i \subseteq N_i$; if there is no ambiguity, notation S_i in $T_i(S_i)$ can be omitted;

$f_i(T)$ - discounting function for final price, a concave increasing differentiable or concave piecewise linear function of total delivery and standard price T in shop i at all points $T > 0$, $f_i(0) = 0$.

We denote the above problem as ISD, where the abbreviation stands for Internet Shopping with Discounts. Its mathematical program can be written as follows:

$$\min \quad \sum_{i=1}^{m} f_i(d_i y_i + \sum_{j \in N_i} p_{ij} x_{ij}), \tag{1}$$

$$\text{s.t.} \sum_{i \in M_j} x_{ij} = 1, \ j = 1, \ldots, n, \tag{2}$$

$$0 \leq x_{ij} \leq y_i, \ i = 1, \ldots, m, \ j = 1, \ldots, n, \tag{3}$$

$$x_{ij} \in \{0,1\},\ y_i \in \{0,1\},\ i = 1,\ldots,m,\ j = 1,\ldots,n. \tag{4}$$

It is worth notice that no discounts version of the ISOP problem with flat shipping rates can be reduced to the well known Facility Location Problem (FLP). Discussions of FLPs can be found in [3].

2 Greedy algorithm

We developed and experimentally tested a simple greedy algorithm for problem ISD. Algorithm was based on the one proposed in [5]. In this algorithm, denoted as G, products are considered in a certain order. The algorithm is run for various product orders and the best solution found is presented to the customer. Let the products be ordered $1,\ldots,n$. Values of total delivery and standard price for all shops are initially set as $T_i = d_i$, $i = 1,\ldots,m$. In iteration j of algorithm G, product j is selected in its eligible shop $i \in M_j$ with minimum value $f_i(T_i + p_{ij})$, and the corresponding T_i-value is re-set: $T_i := T_i + p_{ij}$.

We performed computer experiments, in which solutions obtained by algorithm G were compared against optimal solutions and those provided by algorithm of *Price Comparison Sites* for the examples of problem ISD, which are prepared on the basis of data from the online book industry reported in Clay et al. [2]. In these examples, $m \in \{10, 15, 20, 25, 30\}$, $n \in \{2, 3, 4, 5\}$. For our research we assume that we prepare simple discounting function with two thresholds,

$$f_i(d_i + P) = \begin{cases} d_i + P, & \text{if } 0 < P \leq 50, \\ d_i + 50 + 0.95(P - 50), & \text{if } P > 50, \end{cases} \tag{5}$$

where P is the total standard price of books selected in bookstore i. It is assumed that each bookstore has all the required books. For each pair (n, m), 10 instances were generated. In each instance, the following values were randomly generated for all i and j in the corresponding ranges. Delivery price: $d_i \in \{5, 10, 15, 20, 25, 30\}$, *publisher's recommended price* of book j: $r_j \in \{5, 10, 15, 20, 25\}$, and price of book j in bookstore i: $p_{ij} \in [a_{ij}, b_{ij}]$, where $a_{ij} \geq 0.69r_j$, $b_{ij} \leq 1.47r_j$, and the structure of intervals $[a_{ij}, b_{ij}]$ follow information in Table V in Clay et al. [2]. For each instance, algorithm G was run two times - for a sequence of books in the non-decreasing order of the recommended price and for the reverse sequence. In the worst case, solution found by algorithm G was 4.1% more expensive than the optimal solution, it was 36.1% cheaper than solutions provided by Price Comparison Sites without taking delivery prices into account. The average values of the above mentioned deviations are 2.3%, 45.9% respectively.

New algorithm with forecasting G2 is under development. Changes we made will prevent to provide bad solutions for specific data (even if it is unrealistic) - it also should improve overall performance and provide even better solutions. One of the key aspects is to run a test considering real word data. The efficiency of the new algorithm will be compared with other approaches. Problem size (number of stores, number of shops) will be greatly increased. The performance of the greedy algorithms with respect to optimal and the other heuristic solutions (like Price Comparison Sites) will be re-analyzed and re-computed for the new version (much more complicated) of the problem ISOP.

References

[1] J. Błażewicz, M.Y. Kovalyov, J. Musiał, A.P. Urbański and A. Wojciechowski. Internet Shopping Optimization Problem. *International Journal of Applied Mathematics and Computer Science*, 20(2):385-390, 2010.

[2] K. Clay, R. Krishnan and E. Wolff. Prices and price dispersion on the Web: Evidence from the online book industry. *National Bureau of Economic Research, Inc.*, NBER Working Papers number 8271, 2001.

[3] C. ReVelle, H. Eiselt and M. Daskin. A bibliography for some fundamental problem categories in discrete location science. *European Journal of Operational Research*, 184:817-848, 2008.

[4] The Future Foundation, 2008. E-commerce across Europe - Progress and prospects.

[5] A. Wojciechowski and J. Musial. Towards Optimal Multi-item Shopping Basket Management: Heuristic Approach, in: R. Meersman et al. (eds.). *OTM 2010 Workshops, LNCS 6428*:349-357, Springer, 2010.

Genetic Algorithm for Order Completion and Delivery Problem

Mateusz Cicheński, Mateusz Jarus, Michał Miszkiewicz, Małgorzata Sterna and Jarosław Szymczak

Abstract—**The order completion and delivery problem is a common problem e.g. for small companies which resells products bought from producers. Such companies try to minimize both the cost of purchasing products and the cost of transportation. We present the formal mathematical model of the problem and the lower bound for the criterion value. We propose specialized list heuristic methods and the genetic algorithm solving the case, which is NP-hard. The efficiency of implemented methods was checked in extensive computational experiments. The proposed algorithms have been integrated with the software system designed with a view of supporting charity organizations.**

***Index Terms*—Computer applications, Genetic algorithms, Scheduling**

INTRODUCTION

Scheduling theory [1]–[3] is strictly associated with practice. Real world problems are good inspiration for research which brings new scheduling models. We investigate the problem of satisfying the demand for a set of products. Products needed by a customer are offered in different prices by a few shops or wholesalers located in different distances from the customer. Small companies tend to pick up ordered products using their own means of transportation. To supply the company, one has to select depots at which products should be bought (order completion) and then to construct the route for a vehicle collecting ordered goods (order delivery).

The problem under consideration apparently consists of two optimization subproblems: selecting depots providing required products at the lowest prices and determining the shortest route (namely Hamiltonian cycle) for the selected depots. The first subproblem is computationally easy, while the latter one is intractable, but obviously they cannot be solved separately. Since we can easily obtain population of solutions, evolutionary algorithms [4] are a natural choice for further optimization of initial solutions. In the paper, we propose genetic algorithm [5], [6] based on the list heuristic for selecting depots and the minimum spanning tree heuristic for constructing a route for a vehicle. The metaheuristic approach was tested in the extensive computational experiments, performed for instances reflecting the real world conditions. The efficiency of the proposed genetic algorithm was validated in terms of the improvement of the criterion value in comparison to initial solutions, as well as in terms of the distance to the lower bound proposed within the paper.

PROBLEM FORMULATION

To define the problem under consideration in the more formal way, we use the following parameters:

- m - the number of required products (types of products),
- d_j - the demand for product j, i.e. the number of units of product j required by the customer ($j = 1..m$),
- n - the number of depots offering products,
- c_{ij} - the cost of one unit of product j offered by depot i ($i = 1..n, j = 1..m$),
- a_{ij} - the availability of product j at depot i ($i = 1..n, j = 1..m$),
- t_{ir} - the distance between depot i and r ($i = 1..n$, $r = 1..n$); (t_{0i} and t_{i0} denote the distance from the customer to depot i and from depot i to the customer respectively),

Manuscript received June 15, 2011. This work was supported in part by the grant of National Science Center (N519 643340).

M. Cicheński is with the Institute of Computing Science, Poznań University of Technology, Piotrowo 2, 60-965 Poznań, Poland (e-mail: hrkm@poczta.fm).

M. Jarus is with the Institute of Computing Science, Poznań University of Technology, Piotrowo 2, 60-965 Poznań, Poland (e-mail: mateusz.jarus@gmail.com).

M. Miszkiewicz is with the Institute of Computing Science, Poznań University of Technology, Piotrowo 2, 60-965 Poznań, Poland (e-mail: brennt@interia.pl).

M. Sterrna is with the Institute of Computing Science, Poznań University of Technology, Piotrowo 2, 60-965 Poznań, Poland (corresponding author to provide phone: +48 61 6652982; fax: +48 61 8771525; e-mail: malgorzata.sterna@cs.put.poznan.pl).

J. Szymczak is with the Institute of Computing Science, Poznań University of Technology, Piotrowo 2, 60-965 Poznań, Poland (e-mail: jarek.szymczak@gmail.com).

T - the unit transportation cost.

To find a solution of the problem, we use two types of decision variables. They correspond to the subproblems of: completing order from products offered by depots and constructing a route between depots:

x_{ij} - the non-negative integer variable representing the number of units of product j delivered to the customer from depot i ($i = 1..n, j = 1..m$),

y_{ik} - the binary variable, which takes value 1 if depot i is at position k in the route ($i = 1..n$, $k = 1..n+1$) and 0 otherwise ($y_{i,n+1} = 0$, for $i = 1..n$).

Based on the provided notation, the case under consideration can be formulated as the following integer linear programming problem:

Minimize

$$\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}c_{ij} + \tag{1a}$$

$$T\left(\sum_{i=1}^{n} y_{i1}t_{0i} + \right. \tag{1b}$$

$$\sum_{i=1}^{n}\sum_{r=1}^{n}\sum_{k=1}^{n-1} \max\{0, y_{ik} + y_{r,k+1} - 1\}t_{ir} + \tag{1c}$$

$$\left.\sum_{i=1}^{n}\sum_{k=1}^{n}\left(y_{ik}\left(1 - \sum_{r=1}^{n} y_{r,k+1}\right)\right)t_{i0}\right) \tag{1d}$$

under constraints:

$$\sum_{i=1}^{n} x_{ij} = d_j, \qquad j = 1..m, \tag{2}$$

$$x_{ij} \le a_{ij}, \qquad i = 1..n,\ j = 1..m, \tag{3}$$

$$x_{ij} \ge 0 \text{ and integer}, \qquad i = 1..n,\ j = 1..m, \tag{4}$$

$$\sum_{k=1}^{n} y_{ik} \ge \min\{1, \sum_{j=1}^{m} x_{ij}\}, \qquad i = 1..n, \tag{5}$$

$$\sum_{k=1}^{n} y_{ik} \le 1, \qquad i = 1..n, \tag{6}$$

$$\sum_{i=1}^{n} y_{ik} \le 1, \qquad k = 1..n, \tag{7}$$

$$\sum_{i=1}^{n} y_{i,k+1} \le \sum_{i=1}^{n} y_{ik}, \qquad k = 1..n\text{-}1, \tag{8}$$

$$y_{ik} \in \{0,1\}, \qquad i = 1..n,\ k = 1..n+1. \tag{9}$$

Constraints (2) ensure that the customer's demand is satisfied. Formulas (3) guarantee that the number of product units taken from the depot does not exceed the availability of this product at the location, while constraints (4) ensure that this number is a non-negative integer. According to formulas (5), the position in the route is assigned only to those depots which deliver any product to the customer. Constraints (6) and (7) ensure that each depot can be assigned to at most one position in the route and each position is occupied by at most one depot. Thanks to formulas (8) the numbers of positions given to depots form continuous sequence. Constraints (9) ensures that the decision variables are binary ones.

Constraints (2)-(4) model the problem of selecting depots delivering products to the customer at the minimal cost, while constraints (5)-(9) model the problem of constructing the shortest tour containing all selected depots exactly once, i.e. the subproblem equivalent to the travelling salesman problem.

The criterion function (1) describing the quality of a solution consists of two components. The first one (1a) corresponds to the total cost of products delivered from particular depots. The latter one (1b-d) shows the transportation cost expressed as the total distance multiplied by the unit transportation cost. The total distance is determined as the sum

of the distance from the customer to the first depot in a sequence (1b), the length of the sequence of depots (1c) and the distance from the last depot in the route to the customer (1d).

As we have mentioned the investigated problem of buying and delivering products at the lowest cost consists of two subproblems. The subproblem concerning selecting depots which satisfies the customer's demand at the lowest cost, concerned separately, is computationally easy. It can be solved by the greedy algorithm in $O(mn\log n)$ time. On the contrary, the subproblem of determining the route for a vehicle, which collects ordered products, is obviously computationally hard. It is equivalent to Travelling Salesman Problem (TSP) [7], which is strongly NP-hard. Consequently, the problem under consideration is also strongly NP-hard, since if particular products demanded by the customer are available at only one depot (there is no choice for selecting depots), the problem reduces to TSP (constructing the shortest cycle of depots). It is worth to be mentioned, that the set of depots optimal from the point of view of the total products' cost, may form a very long route causing high transportation costs. On the other hand, closely located depots may offer products at very high prices.

HEURISTIC ALGORITHMS

To provide not only random solutions to initiate a genetic algorithm two heuristics were developed. Both of them consist of two phases: selecting depots and constructing a route. The second stage is the same in both approaches.

Selecting Depots

The first heuristic (greedy heuristic) developed for depots selection is an optimal algorithm for determining a set of depots satisfying the customer's demand at the lowest products cost (taking into account no transportation cost). As an input, the algorithm uses the list of depots associated with each product. Depots are ordered ascending by prices of this product. A final solution is created by taking product units from depots one by one, until the total demand is satisfied.

The second heuristic (priority heuristic) is based on depot priorities. Contrary to the first approach, depots are ordered by the weighted sum of priorities. These priorities are determined with regard to a few factors, not only with regard to the product price. Each priority is calculated in following way: depots are ordered from the worst to the best according to a certain priority. The priority value is calculated as a position in this order divided by the number of depots. The most desired depot has the priority value equal to 1, the worst one – 0. The final evaluation of the depot is based on the weighted sum of particular priorities.

The proposed algorithm takes into consideration the following three priorities:

- the distance from the customer,
- the prices of demanded products present in a certain depot,
- the prices of all demanded products assuming that products not present in a depot are treated as they were the most expensive ones.

It is worth to mention that both algorithms allow to construct more than one solution of the problem. Once solution is found, algorithms proceed to create another solution built from depots that were not used in previous solutions.

Constructing Tour for Selected Depots

Greedy and priority heuristics calculate only solutions of the subproblem of selecting depots which will provide products to the customer. When depots are selected, the shortest path necessary to visit all of them and come back to start point should be found. This path corresponds to the shortest Hamiltonian cycle that has to be constructed between all the selected depots and the customer location. The second subproblem is therefore TSP (Travelling Salesman Problem). TSP is obviously a strongly NP-hard problem. However in case of depots, which are connected with paths that satisfy the triangle inequality we deal with metric TSP (also known as Δ-TSP). For such a special type of TSP a classical Minimum Spanning Tree Heuristic (*MSTH*) [7], [8] can be applied to construct a tour, i.e. to determine the order of the depots in the cycle. *MSTH* constructs the minimum spanning tree for the graph with usage of Kruskal Algorithm in $O(\tilde{n}^2)$ time [9] (where $\tilde{n}$ is the number of the depots selected in the first phase). Then *DFS* (depth first search) on this tree is applied in $O(\tilde{n})$ time. The sequence obtained with *DFS* is transformed to Hamiltonian cycle also in $O(\tilde{n})$ time. *MSTH* is efficient – the length of the resulting tour is at most twice as long as the optimal tour [8].

The criterion value for the solution constructed by the two-phase method (greedy heuristic or priority heuristic with different weights settings for depots selection and with *MSTH* for the route calculation) provides an upper bound of the optimal criterion value (*UB*). The upper bound together with the lower bound, presented in the next section, can be used to estimate the efficiency of the genetic algorithm.

LOWER BOUND

An optimal solution for the order completion and delivery problem cannot be constructed in polynomial time. To estimate the quality of heuristic solutions for large instances we propose the lower bound of the optimal criterion value (*LB*).

As mentioned before, the optimal solution of the selecting depots subproblem may not be optimal for the complete problem due to long distances between selected depots. The greedy heuristic algorithm constructs a solution optimal only from the point of view of product's costs. The quality of a complete solution obviously depends also on distances between selected depots. Hence, the lower bound is equal to the sum of the optimal cost of products and the minimal possible distances between the customer location and one of selected depots (products have to be taken from at least one depot). The lower bound is defined by the following formula:

$$LB = Solution_{GreedyHeuristic} + \left(\min_{i=1..n}\{t_{0i}\} + \min_{i=1..n}\{t_{i0}\}\right) \cdot T$$

GENETIC ALGORITHM

In the presented research a classical framework of the genetic algorithm [4]-[6], adjusted to the specificity of the problem under consideration, was used.

Genetic Algorithm (*GA*)
generate initial population *P*;
evaluate population *P*;
while (termination conditions not met) do
  select mating population *M* from P
  recombine *M* obtaining *P*';
  mutate *P*' obtaining new population *P*;
  evaluate population *P*;

Usually an initial population is created from random solutions. Each of them is evaluated (according to the fitness function). Then a mating population is selected. A given number of solutions are picked to this population (the better solution the bigger probability that it will be chosen). The next step is recombination phase, in which two parent solutions are chosen from the mating population randomly and their offspring is added to a new population. Such an action, called crossover, is performed with a certain probability, which is a control parameter of a genetic algorithm. The recombination phase is over when a new population reaches its desired size. Then, the mutation operator is applied with a certain probability (another control parameter of GA) on each solution from a new population. After all, a new generation of solutions is created. The algorithm stops when it meets one of the predefined termination conditions.

Solution representation

A solution is an ordered sequence of assignments of all units of demanded products to depots that will deliver them. The representation is optimized in such way, that it groups units of a certain product delivered by the same depot.

Population

An initial population is created from random solutions and solutions created by greedy and priority heuristic methods (with different settings of weights).

Selection

In the genetic algorithm three selection methods are used:

- *roulette selection* – in which probability of choosing each solution is inversely proportional to its quality,
- *ranking selection* – in which probability of choosing each solution is proportional to its position in quality descending order (it's similar to roulette selection),
- *tournament selection* – in which a number of solutions are randomly picked to form a group, then the best solution from the group is chosen to mating population.

Recombination

The following genetic operators are used in the recombination phase:

- *one-point crossover* – a crossover point in a solution is chosen randomly (and hierarchically, i.e. firstly the product is chosen, secondly its certain unit among all demanded units is selected). A chosen crossover point splits each parental solution in two pieces. These cross-connected pieces form the offspring.

- *two-point crossover* – it's very similar operator to the one described above. It uses two crossover points and divides each parental solution into three pieces. Cross-connection of these pieces creates the offspring.

Usage of any of presented crossover operators may lead to infeasible solutions. Such a situation occurs, when the demand for a product taken from a certain depot exceeds the number of product units that this depot can deliver. Such a solution is repaired by decreasing the amount of this product taken from the critical depot and increasing, if it's possible, the amount of the considered product ordered from another depot included in the analyzed solution. If the demand is not satisfied, the missing part of the order is taken from one of the parental solutions.

Mutation

Mutation is implemented in a very simple way: k positions (where k is a control parameter) are randomly removed from a solution and replaced with random data, ensuring that a mutated solution is feasible.

Evaluation

Fitness function evaluating solutions is defined as the cost of all the products and the transportation cost, which is determined by *MSTH*.

Termination Conditions

The algorithm is stopped, if one of the following termination conditions is met:

- the maximum number of generations occurs,
- the maximum number of generations without improvement occurs,
- the satisfying initial solution to actual solution quality ratio is reached.

COMPUTATIONAL EXPERIMENTS

In purpose of validating the efficiency of the genetic algorithm and heuristic methods proposed for the order completion and delivery problem a number of computational experiments were performed.

Data Set

Input data for experiments was generated randomly, however, in such a manner that it corresponds to reality. Instances used for tests had the following features:

- 48 Polish major cities were used as depots locations,
- distances between cities were equal to road distances taken from Bing Maps [10],
- the customer location was placed in Poznań,
- the number of different product types in orders was a parameter ranging from 5 to 200,
- the transport cost was equal to 1 PLN (to the rounded official price for one kilometer, set by the Polish Government in 2007 to 0,8358 PLN),
- each product type had a base price taken with uniform distribution from 10 to 100 PLN,
- in each depot, the cost of a product was determined as the modified base price, changed by factor taken with normal distribution from -50% to +50%,
- the number of units of each product type in orders was picked with uniform distribution from 1 to 10.

To determine contents of depots, the number of demanded product units were distributed among depots with following three strategies:

- round robin – products from orders were placed in depots unit by unit,
- clone – each depot had all the products from orders in its offer,
- even – one unit of a certain product type was added to each depot at once (i.e. 48 units in 48 depots at once) until demand for this product was satisfied.

For even and round robin strategies reference values were defined to evaluate the quality of heuristic solutions. A reference value is similar to the lower bound, however, contrary to the lower bound, it is valid only for a certain type of instances; it is not an instance independent lower bound. Reference values are presented below, where k is the maximum number of needed units of any product type, $\tilde{n}$ is the number of depots in a solution, $t_{[x]}$ is the x-th distance in the sequence of distances ordered in ascending manner, l is the sum of numbers of units of all products.

$$RV_E = Solution_{GreedyHeuristic} + \left(\min_{i=1..n}\{t_{0i}\} + \sum_{x=1}^{k-1} t_{[x]} + \min_{i=1..n}\{t_{i0}\} \right) \cdot T \cdot$$

The reference value for even strategy of generating instances is calculated for k such that $k \leq \tilde{n}$. It's based on the fact that to collect all needed units of products the customer has to visit k depots offering the maximum number of needed items.

$$RV_{RR} = Solution_{GreedyHeuristic} + \left(\min_{i=1..n}\{t_{0i}\} + \sum_{x=1}^{\min(l-1,\bar{n}-1)} t_{[x]} + \min_{i=1..n}\{t_{i0}\} \right) \cdot T$$

The reference value for round robin strategy of generating instances is based on the fact that the customer has to visit all the depots which offer at least one unit of all demanded products.
For clone strategy the reference value is defined as the lower bound: $RV_C = LB$.

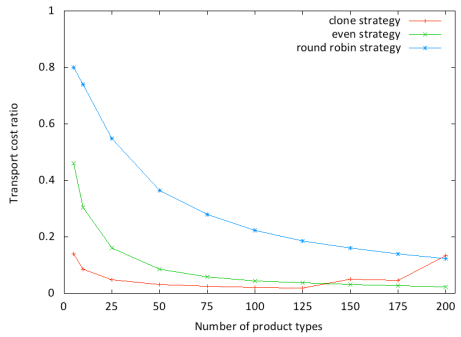
Genetic Algorithm Control Parameters

Before the main phase of computational experiments, the genetic algorithm was tuned: its efficiency was tested for a number of parameters settings. The highest efficiency was achieved for the following configuration of control parameters:

- population size: 100,
- mutation probability: 0.15,
- crossover probability: 0.30,
- maximum number of iterations: 500,
- maximum number of iterations with no improvement: 100,
- selection method: tournament with group size ranging from 10 to 30,
- number of genes to replace in mutation: 1.

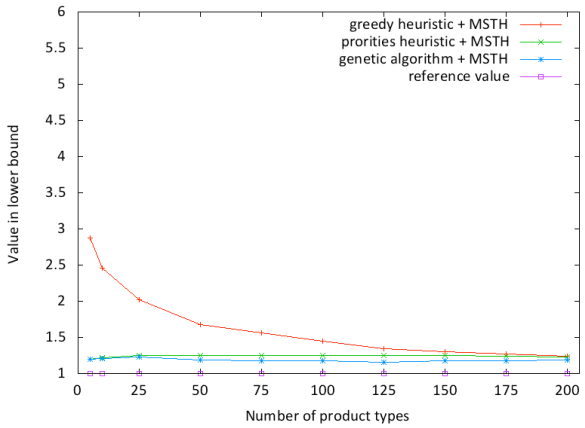
Results of Computational Experiments

Before presenting more detailed results for particular types of instances (corresponding to particular strategies of distributing products among depots), we show the ratio between the transport cost and the overall solution cost for these three types of input instances. The performance of algorithms solving the problem is obviously related to this ratio.

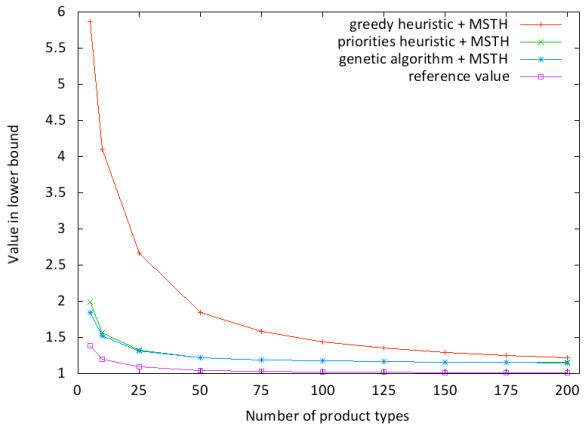


Transport cost to overall cost ratio

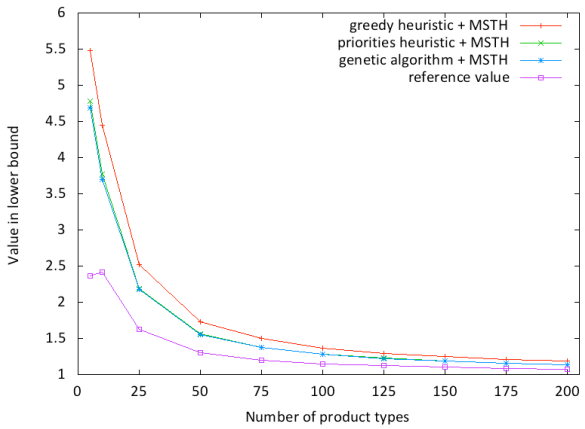
The performance of all implemented algorithms are analyzed separately for particular strategies of distributing products among depots in input instances. The efficiency of algorithms is expressed with regard to the reference values (it shows how many times the criterion value for heuristic solution is worse than the reference value).



Results for clone strategy

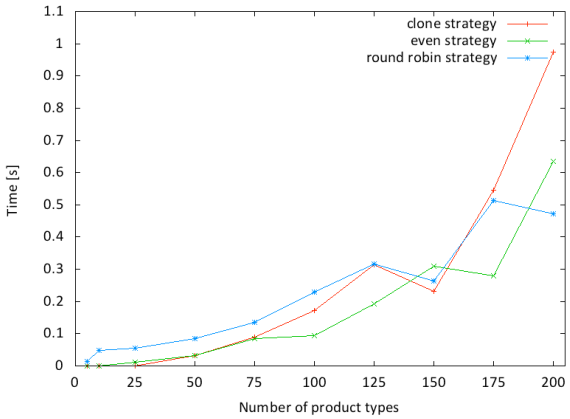


Results for even strategy



Results for round robin strategy

To evaluate time efficiency of the genetic algorithm, for each type of input instances an average CPU time per GA generation was measured.



Time per generation in genetic algorithm

CONCLUSIONS

The problem of order completion and delivery can be met in many small companies. It also appears to be an interesting combinatorial case.

The results of experiments proved the efficiency or heuristic and metaheuristic algorithms proposed in this paper. The genetic algorithm generated solutions of the highest quality for all types of input instances, however in some cases the difference between GA and priority heuristic was marginal. Worth to mention is fact, that in such cases the criterion value for GA solution was at most few dozen percent worse than the reference value (which is probably smaller and certainly not worse than the optimal criterion value). For all types of instances the efficiency of the algorithms is connected with the number of ordered products (the bigger number of products the better ratio to the lower bound). It is caused by the fact, that the transport cost is only a small fraction of the overall cost in such solutions, so the computationally harder part of the problem (connected with calculation of TSP route) has less influence on the overall cost.

Computational results disclosed two facts especially worth explanation. First one is the growth of the transport cost to the overall solution cost ratio for clone strategy for input instances with the number of products bigger than 125. This phenomena is probably connected with the higher probability of existence of the depot that is located far from the customer, but offering products at very small prices (for this strategy, all the depots offer all the products, but in different prices). In such case it may be profitable to take all the products from one depot, even when it is located far from the customer. Second fact is the difference among algorithms efficiencies for round robin strategy. It is caused by the influence of the depots order given as input for *MSTH*. For each algorithm depots are ordered in different manner and therefore also the order of visiting the depots, (which is a result of the depth first search in *MSTH*) may vary, causing differences in the solutions quality.

The genetic algorithm presented in this paper is a part of software system, which may serve charitable organizations via Internet. The system allows registered users to exchange information on demanded products and offered goods. It is able to suggest efficient solutions to carry out a set of orders, based on solutions provided by the algorithms described within this work.

REFERENCES

[1] J. Błażewicz, K. H. Ecker, E. Pesch, G. Schmidt and J. Węglarz, *Handbook on Scheduling: From Theory to Applications*. Berlin-Heidelberg, New York: Springer, 2007.
[2] J. Y-T. Leung, Ed, *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. Boca Raton: CRC Press, 2004.
[3] M. Pinedo, *Scheduling: Theory, Algorithms and Systems*. New York: Springer, 2008.
[4] T. Bäck, D. B. Fogel and Z. Michalewicz, Eds, *Handbook of Evolutionary Computation.* Bristol: OP Publishing Ltd., 1997.
[5] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975.
[6] K. Sastry, D. Goldberg and G. Kendall, "Genetic algorithms", in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Methodologies*, E. K. Burke and G. Kendall, Eds., New York: Springer, 2005, pp. 97–126.
[7] D. L. Applegate, R. E. Bixby, V. Chvátal and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*. Princeton: Princeton University Press, 2006.
[8] M. Held and R. M. Karp, "The traveling salesman problem and minimum spanning trees", *Operations Research*, vol. 18, 1970, pp. 1138–1162.
[9] J. B. Kruskal Jr., "On the shortest spanning subtree of a graph and the travelling salesman problem", *Proceedings of the American Mathematical Society*, vol. 7, No. 1, 1956, pp. 48-50.
[10] *www.bing.com/maps/*

RFID – Possible Applications and Challenging Problems

Grzegorz Fenrich (grzegorz.fenrich@cs.put.poznan.pl),
Małgorzata Sterna (malgorzata.sterna@cs.put.poznan.pl)

1. Introduction

In the contemporary world every container, box, every product should be identified. For many years the bar codes have been used to label different kinds of items. They can be seen in shops on food or toys, in shipyards on containers, nearly on every product which appears in human hands. Bar codes are commonly used, but they don't give to many pieces of information, they are only numbers which identify products. In XXI century that is not enough. People want to know from which country package is from, which company sent it and what was the path the package traveled. Customers choosing food want to know, for example, on which field e.g. carrots were grown, which fertilizers were used for them, etc. These pieces of information are only tiny parts of knowledge people want to know about products they are buying or processing. Bar code is not enough to provide it. The contemporary world needs a new method of items identification - it needs RFID.

2. RFID - What is it?

RFID is a radio frequency identification technology [D03] consisting of a reader (also known as an interrogator) and a tag (or a transponder), which is a chip connected to an antenna. When a tag passes through a field covered by a reader, it transmits information stored in it.

Tags, which may be attached to different kinds of objects, can be either passive or active. Passive tag cannot send any information by itself. It only sends information when, for example, it reaches the gate with the reader. Therefore passive tags are cheaper. In 2011 the average price of a passive tag does not exceed $0.05. They have simple construction and, particularly, they do not contain any battery. But they need certain (usually expensive) infrastructure consisting of readers - gates. Active tags have more advanced structure (therefore they are larger). They can be programmed when to send information and what type of information it should be, e.g. about a country from which the package with a tag came, about a serial number of the container on which a tag is on, about the whole road a tag traveled. The range of an active tag is larger than the passive one, so the infrastructure may be less dense. Active tag contains a battery thanks to which the information from it can be sent any time. For example, a user sitting with a computer can check where some containers with active RFID tags are located. Active tags are more expensive but they can be read without static gates (readers). For this kind of RFID, commonly used Wi-Fi networks are fully sufficient. The rough idea of passive and active RFID system is showed in Figure 1 and in Figure 2 respectively. [DP10]

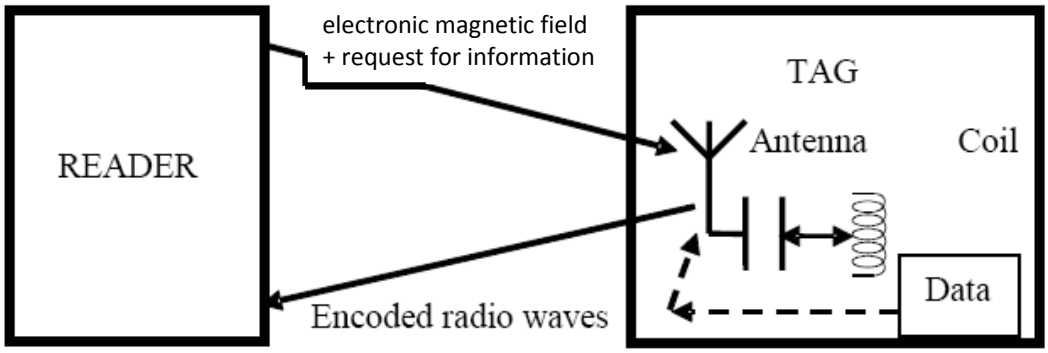


Figure 1 Passive RFID

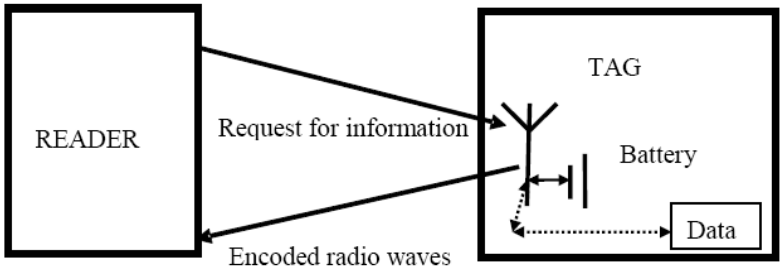


Figure 2 Active RFID

Besides passive and active RFID, a third type of tags, called semi-passive tags, are also available. From one hand they contain a battery that provides energy to the chip (as in active RFID), but on the other hand the reader field is still necessary for the transmission from a tag to a reader (as in passive RFID). Active and semi-passive tags are used to track high-value items which have to be scanned over longer distances. [D08]

As it has been mentioned, RFID is a successor of a bar code. Bar codes work well, but RFID has two main advantages:

a) RFID emits a unique identifier for each item, distinguishing it from other identical items,
b) RFID is readable without precise positioning and line-of-sight contact.

3. Possible Applications of RFID

Thinking about RFID, about the technology and its usage, most probably one brings in her/his mind: big shipyards coping with containers equipped with tags, corporations labeling their products with RFID or rent-a-car firms tracking their cars. That's all true, but RFID is used by many more business branches of different kinds than mentioned ones.

Military applications

Like almost every technology, also RFID owes much to army and military. In USA, the military services and the Defense Logistics Agency (DLA) have invested millions in radio-frequency identification tags and their infrastructure [KR11]. The Department of Defense of U.S. Army

mainly invested in in-transit visibility information system. The knowledge gathered by RFID system is used to make decisions whether to order more goods and where these goods should be delivered. Actually, RFID is used to monitor whole supply chain: to optimize supply process, inventory management and transportation in every piece of the world where the U.S. Army is present. More than 3 million active RFID tags are in circulation today in this military distribution system, with approximately 3100 tag-reader devices distributed in Kuwait, Iraq, Pakistan and Afghanistan [KR11].

As it was said before, the cost of a basic passive tag is about $0.05, but army uses more complicated tags. First of all they have to be more durable, for example more weather resistant: to rain at the raining season or sand in the deserts. Cost of these types of tags is about $60 per piece, so they are 1200% more expensive then the basic model. Additionally, the price of RFID infrastructure is also higher - interrogator equipment can cost between $2000 and $20000 per site.

RFID technology is used by different forces of U.S. Army. For example, the U.S. Navy, in partnership with DLA, is taking advantage of its RFID investments to improve business processes in Hawaii [KR11]. Navy organizations placed RFID readers at receiving points and warehouse doors throughout their supply chain in Hawaii, and they established interfaces with distribution systems. In addition to using the RFID-to-Automatic Information System (AIS) interface to automate their business processes, the Navy made each tag-read transaction visible to its collaborators. Because they are satisfied with the progress of these improvements to shore-based operations, Navy ordnance and supply experts are seeking approval to establish RFID capability on vessels to support their onboard supply processes.

The Air Force is using RFID technology not only to improve its business processes, but also to control the inventory management of sensitive items such as nuclear weapons-related material [KR11]. Air Force inventory experts are using the passive RFID technology to optimize the time and effort invested in an individual item management process. In addition to two-person identification and documentation on each item, the Air Force uses passive RFID to control nuclear weapons-related items. Tags are also used to identify when these items are moved from one area to another inside a facility, between separate facilities, or between installations. The system provides an alert when items are not detected by a receiving installation before the expected delivery date.

The Marine Corps has equipped each of its main operating bases with the capability to read passive RFID located on items shipped from DLA. The information is used to document delivery process. Once distribution managers complete the integration with their information systems, the Marine Corps expects to reduce errors in supply system and increase its efficiency by automating processes that are currently being completed by hand [KR11].

Pay Pass. RFID = end of paper money?

According to Jeremy N. Smith, in 2005, for the first time, purchases done by credit cards exceeded those made by cash [Ch10]. Cards with RFID chips are more and more popular today. Almost every bank has them in its offer. Couples of banks give even no choice to their customers – the only choice is a pay-pass card. But this is only the beginning, in the near future, all credit cards will be gone and all the payment will be executed with mobile phones equipped with a special chip - RFID of course. This chip will be a pay-pass card. To pay a bill with a pay-pass

card, for example, for gasoline, a customer has only to pass near the reader with her/his card (phone) in a pocket. In the future, perhaps one will only fuel up a car and just drive away. The reader will be installed near a car and will send the information about a customer and her/his bill to the bank when she/he finishes refueling, the bank will do the rest. This situation is not so far away from the present day, this is possible even today. Current RFID technology can be embedded in any device larger than a pea. Modern cards or phones can send user information to properly equipped kiosks, such as vending machines, subway turnstiles, or store checkout lines. They complete transactions with no contact necessary and with higher security than more easily replicated magnetic strip credit cards. Last year, over 25 million consumers used chip-embedded credit cards to make contactless payments. By 2013, according to Javelin Strategy & Research [Ch10], the number of users of such cards will likely exceed 57 million; they will pay their bills in fast-food restaurants, sports and concert stadium concessionaires, convenience stores, and gas stations. [J10]

Nokia, as the first company, has placed contactless payment technology into its C7 handset at October 2010. For some time company has not informed about RFID chip in this phone. The existence of the chip in the phone was not mentioned anywhere: neither on the package nor in advertisements. The 'near field communication' (NFC) chip had not been activated till 2011. Now it works and customers can pay with their modern Nokia phones. Future has arrived! Analytics suspect that at year 2014, 13% of cell phones will be equipped with NFC. [J10]

Another advantage by having RFID inside a mobile phone is possibility of using vouchers. Vouchers can be sent directly to a phone within seconds. Thanks RFID loyalty accounts and receipts for purchases will be more personalized. Buying a ticket will be unnoticed by customers: people will just pass the gate, and the system will take money directly from their accounts. Hotels will be able to send 'keys' to guests' handsets to open their doors with. People will be able to open their own front doors at home with handsets. In some far-flung future one will have no use for a wallet, for keys or even for cash. NFC may be the beginning of the end of cash. [OSK05]

Medicine

The unexpectedly vivid application field for RFID is medicine. Actually, surprisingly many branches of medicine use RFID technology. For example, RFID tags are applied in monitoring the healing process of bones fractures instead of X-Rays and CT scans [GMR09]. As it is known, booth X-Rays and CT scans emit radiation which is not good for human health. In contrary, according to researchers, the RFID system is highly sensitive and radiation free. It is based on implants with RFID. This system consists of a transponder module on the implant and an external wireless reader. The passive transponder, which is used, contains no power source, so it can withstand high temperatures such as those used in sterilization [ChOCh10].

RFID appears also in operating theaters, making surgeries safer. Tags are located on surgical instruments and other items in order to control their number. RFID detection devices allowed decreasing the probability of leaving sponges or other object, such as surgical instruments, in the patient body from one in 1000 operations to one in 18000. [D08]

Libraries

Libraries are another large application field for RFID technology. Every book can be tagged by RFID, thanks to which each book can be easily found in the library. Moreover, tags can store all necessary information about books. In the near future, in the library there will be no need to fill any document by users or passing every book by the barcode reader. Someone who wants to borrow a book will just take it and live the library without any formalities.
In United Kingdom there already exists the ISO data tag standard, which gives RFID an opportunity to be present in all British libraries. It will enable borrowing a book in one place and return it in another, because the ISO standard governs how RFID tags store information. The ability for all compliant tags to be read in the same way will support the interoperability of disparate self-service solutions. [B06]

Other application fields

The examples presented before do not cover all application fields for RFID. For example Hannes Harms, a design engineering student at Royal College of Art in London, has developed NutriSmart [C09] – a food tracking system which uses edible RFID tags. The markers would let consumers trace the entire supply chain, hidden behind every item in their cupboard. They will alert dieters or people with serious food allergies about dangerous ingredients. Moreover, NutriSmart refrigerators could support food management, because tags could warn people when some products are about to pass its expiration date. The system also can hook up with a smart plate. Once one put her/his food on the plate, an embedded reader can analyze the feast and transmit to a mobile phone the information: where food came from, which is its history, which are its nutritional and caloric data. Of course the questions arise what happens to the tags after digestion or how difficult it would be to convince people to start ingesting such technology. [C09]

4. Challenging problems

RFID technology can greatly benefit companies when implemented correctly, but it can also create numerous unique security risks that expose vulnerabilities and shortcomings of RFID devices. Before deciding to implement RFID solutions, firms should examine the threats to determine the amount of vulnerability and risk they are willing to take on. [D03]

One of major problems appearing in RFID systems are unauthorized reads. This simplest RFID threat comes from unauthorized users gaining access to the information stored on tags. Basic RFID tags are not able to validate users. They were developed to respond to a read request generated by any device – authorized or not authorized to do this. When sensitive information is stored on tags, the risk of stealing the information by an unauthorized user appears. A hacker using, for example, the open source program called RFDump can read virtually any RFID tag. The software was developed to work on almost all RFID protocols. RFDump highlights the vulnerability of RFID tags to hackers. Because of this threat, firms should not place sensitive information on RFID devices. [LG10]

The problem of cloning is the opposite of the unauthorized read threat. Cloning is when an attacker mimic authentic RFID tags by writing appropriately formatted data on blank RFID tags.

By cloning the tag, a hacker is able to trick the RFID reader into believing it is an authentic tag, thereby granting the hacker unauthorized access. Researchers at John's Hopkins University demonstrated that by cloning an authentic RFID credit card, a hacker can purchase items without the original tag [R03]. Unlike with traditional credit cards, it would be impossible to notice that a hacker has stolen an RFID credit card, because the theft can be done wirelessly, in the way which the card owner is not aware of. [Ch10]

Furthermore, RFID devices could be used to transmit malicious viruses and malware. Through SQL injections on RFID tags, they are able to exploit vulnerabilities in the backend software that supports RFID devices [R03]. This vulnerability is not a threat to RFID tags themselves, but to the system they are used in. Hopefully, now researchers are aware of the possibility of introducing viruses into RFID devices and they work on proper defense solutions. Another threat to RFID technology is the potential buffer overflow attack. In a buffer overflow attack, a hacker sends more data than it is expected by the software. In such a situation the software cannot handle the excess of data and crashes [KR11].

Other problems with RFID result from the technology itself, not from malicious human actions. In active tags, a battery causes serious problems. When a tag runs out of power, it might be useless, since a battery should be replaced or recharged. Another problem is the range and localization of a reader/tag. RFID system works well in open spaces without any interrupting signals. But most of factories, warehouses have walls, ceilings, working machines and technical infrastructure, which influence the precision of RFID system. Designers of the RFID infrastructure have to take into consideration all these issues. [R03]

5. Future research

Object localization systems based on radio frequency identification technology give many promising opportunities. By combining localization and identification capability, existing applications can be enhanced and new ones can be developed for RFID technology.

Within the future research, we would like to design and implement RFID system simulator. It will model a modern factory or warehouse equipped with RFID readers and tags. The simulator will allow checking how many readers are necessary in the building, where these readers should be located or how often tags have to be localized to fully know their positions [SFJMPRSS05]. Such a simulator, in which a designer will be able to reflect the structure of a whole warehouse with all its levels, walls and details of the technical infrastructure will provide useful information how RFID system works. Based on these data, we would like to propose algorithms optimizing the usage of RFID tags to improve e.g. storing area usage, tools or vehicle usage etc.

6. Conclusions

RFID technology is for sure the technology of the future. It is a successor of bar codes. But has the future already come? Are we ready to fully use the benefits of it? It seems that the answer is "no". The security protocols are still not good enough. Banks and shops are not ready to fully use the pay-pass payment. Also people are not ready to give up money and use only electronic ways of payment. Furthermore, RFID may store information, about people, about their health and other important issues. Until this information will not be 100% safe, people will not be willing to use RFID.

For the present day RFID is perfect solution to support supply chain management, for shipyards, for big companies like car rental business or libraries. It seems that industry and business is ready for RFID, it could and should use RFID with all its benefits - but what about ordinary people? Future seems to be bright: no wallets, no passports, no identification cards; all information closed in one tiny RFID chip under human skin. Such a solution is already possible, because RFID tags can be powered by human warmth. The optimal places for locating them in a human body are – hand and forehead. But do people want to be labeled? Such idea brings into mind the mark of the beast mentioned by Saint John Book of Revelation. Is this the right way?

Acknowledgement: This work was partially supported by the grant of National Science Center (N519 643340).

References

[B06] Bean L. (2006): RFID: Why the Worry?, *The Journal of Corporate Accounting & Finance* No. 17, pp. 3–13

[CL10] Caldwell-Stone D. (2010): RFID in Libraries*, Library Technology Reports 46*, No. 8, pp. 38-44

[ChOCh10] Choi Y. B., Oh T. H., Chouta R. (2011): RFID Implementation and Security Issues*, Information Security and Assurance: Communications in Computer and Information Science*, Vol. 200, pp. 236-249

[Ch10] Choueke M. (2010): This is the beginning of the end for cash, *Marketing Week* http://www.marketingweek.co.uk/disciplines/digital/this-is-the-beginning-of-the-end-for-cash/3020625.article

[C09] Coomes S. (2009): RFID chips turn cell phones into coupons, *Nation's restaurant news* http://www.nrn.com/article/rfid-chips-turn-cell-phones-coupons

[D08] Dirjish M. (2008): RFID Technology Monitors Bone Fractures As They Heal, *ElectronicDesignProducts* *http://electronicdesign.com/article/commentary/rfid_technology_monitors_bone_fractures_as_they_heal.aspx*

[DP10] Dolgui A., Proth J. (2010): *Supply Chain Engineering: Useful Methods and Techniques,* Springer, London

[D03] Dubendorf V. A. (2003): *Wireless Data Technologies,* John Wiley & Sons Ltd, West Sussex

[GMR09] Gandino F., Montrucchio B., Rebaudengo M. (2010): Tampering in RFID: A Survey on Risks and Defenses, *Mobile Network Applications*, Vol. 15, pp. 502–516

[J10] Johnson T. M. (2010): Radio Frequency Identification The Future is Now!, *Defense AT&L* http://findarticles.com/p/articles/mi_m0QMG/is_5_39/ai_n56141898/

[KR11] Kelly P., Robertello C. (2011): Radio Frequency Identification Tags in Modern Distribution Processes*, Army sustainment*

http://www.almc.army.mil/alog/issues/May-Jun11/rfid_moddistrib.html

[LG10] Lekkas D., Gritzalis D. (2010): e-Passports as a means towards a Globally Interoperable Public Key Infrastructure, *Journal of Computer Security*, Vol. 18, No. 3, pp. 379-396

[OSK05] Ohkubo M., Suzuki K., Kinoshita S. (2005): RFID privacy issues and technical challenges, *Communications of the ACM,* Vol. 48, pp. 66 - 71

[R03] Rappold J. (2003): The risks of RFID, *Industrial Engineer*
http://www.allbusiness.com/specialty-businesses/742505-1.html

[SFJMPRSS05] Smith J. R., Fishkin K. P., Jiang B., Mamishev A., Philipose M., Rea A. D., Sumit R., Sundara-Rajan K. (2005): RFID-based techniques for human-activity detection, *Communications of the ACM*, Vol. 48, pp. 39 - 44

Extensions of Learning Vector Quantization for Relational Data

Xibin Zhu, Frank-Michael Schleif, Barbara Hammer
{xzhu,fschleif,bhammer}@techfak.uni-bielefeld.de
Bielefeld University, CITEC-*Centre of Excellence*,
33615 Bielefeld, Germany

February 17, 2012

Abstract In many application areas machine learning techniques play a very important role by helping people to inspect data and to simplify the job to deal with data. But the prominent methods such as Support Vector Machine (SVM) act as a black box, and the decision can not be easily inspected by human beings. In contrast, prototype-based learning methods offer a very intuitive way to inspect the data in input space: they represent their decisions in terms of typical representatives (prototypes) in the same space, which can be directly inspected.

Classical unsupervised prototype-based methods such as k-Means, Topographic Mapping, Neural Gas (NG), or the Self-Organizing Maps (SOM) and statistical counterparts such as Generative Topographic Mapping (GTM) infer prototypes based on input data only [9, 8, 2]. In the supervised domain Learning vector quantization (LVQ) is one of the most popular prototype-based methods, which take class labeling into account and find decision boundaries as accurately as possible corresponding known class labels.

In modern application areas, not only the size of modern data sets, but also its complexity increases rapidly. Improved sensor technology, for example, leads to very high dimensional measurements corresponding to a very detailed resolution of the available information. At the same time, dedicated data formats such as XML files, network data, graph structures and the like become more and more common. Classical prototype-based methods as mentioned above usually deal with Euclidean vectors only. Hence these algorithms are no longer suitable in these settings. While the Euclidean distance yields to almost meaningless values for high dimensionality, a lossless vectorial representation is not even possible for data structures such as sequences, trees, or graph structures.

This fact has led to a variety of extensions of prototype-based techniques to deal with more complex data formats, see e.g. [1]. One prominent interface is offered by a general similarity or dissimilarity matrix: only pairwise similarities for dissimilarities of data have to be defined based on which learning takes place. Various dissimilarity measure are available for dedicated data formats: for

example, alignment for sequences [4], functional norms for functional data [11], divergences for probability distributions [12], graph and tree kernels [6], or the compression distance for general symbolic sequences [3]. Hence a formulation in terms of dissimilarities extends the applicability of prototype-based techniques to a large variety of modern application areas. Since data are characterized by pairwise relations rather than Euclidean vectors, we refer to these data as 'relational data'. Although this problem can be partially avoided by appropriate metric learning or by kernel variants, if data are inherently non-Euclidean, these techniques are not applicable.

In this contribution we concentrate on two variants of LVQ, namely Generalized LVQ (GLVQ) and Robust Soft LVQ (RSLVQ), and introduce extensions of them with techniques used in unsupervised domain [5, 10], so that they can directly deal with relational data sets which are characterized in terms of a symmetric dissimilarity matrix only. The key ingredient of the technique is: if prototypes are represented implicitly as linear combinations of data in the so-called pseudo-Euclidean embedding, the relevant distances of data and prototypes can be computed without an explicit reference to a vectorial data representation. This principle holds for every symmetric dissimilarity matrix and thus, allows us to formalize a valid objective of RSLVQ and GLVQ for relational data (for more technical details, see [7]). We evaluated the techniques on several benchmarks, and the results are comparable to SVM [7].

References

[1] M. Biehl, B. Hammer, S. Hochreiter, S. Kremer, and T. Villmann, editors. *Similarity-based learning on structures, 15.02.09 - 20.02.09*, volume 09081 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, 2009.

[2] C. M. Bishop, M. Svensén, and C. K. I. Williams. Gtm: The generative topographic mapping. *Neural Computation*, 10:215–234, 1998.

[3] R. Cilibrasi and P. M. B. Vitányi. Clustering by compression. *IEEE Transactions on Information Theory*, 51(4):1523–1545, 2005.

[4] D. Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

[5] B. Hammer and A. Hasenfuss. Topographic mapping of large dissimilarity datasets. *Neural Computation*, 22(9):2229–2284, 2010.

[6] B. Hammer and B. Jain. Neural methods for non-standard data. In M. Verleysen, editor, *European Symposium on Artificial Neural Networks'2004*, pages 281–292. D-side publications, 2004.

[7] B. Hammer, F.-M. Schleif, and X. Zhu. Relational extensions of learning vector quantization. In B.-L. Lu, L. Zhang, and J. Kwok, editors, *Neural Information Processing*, volume 7063 of *Lecture Notes in Computer Science*, pages 481–489. Springer, 2011.

[8] T. Kohonen. *Self-organizing Maps*. Springer, 1995.

[9] T. Martinetz, S. Berkovich, and K. Schulten. ¨Neural-gas¨ Network for Vector Quantization and its Application to Time-Series Prediction. *IEEE-Transactions on Neural Networks*, 4(4):558–569, 1993.

[10] E. Pekalska and R. P. Duin. *The Dissimilarity Representation for Pattern Recognition. Foundations and Applications*. World Scientific, 2005.

[11] J. Ramsay and B. Silverman. *Functional data analysis*. Springer, 2005.

[12] T. Villmann and S. Haase. Divergence-based vector quantization. *Neural Computation*, 23(5):1343–1392, 2011.

Analyzing Motion Data by Clustering with Metric Adaptation

Bassam Mokbel, Mario Heinz, Georg Zentgraf

CITEC – Cognitive Interaction Technology Center of Excellence
Bielefeld University, P.O. Box 100131, D-33501 Bielefeld, Germany
{bmokbel|mheinz|gzentgra}@techfak.uni-bielefeld.de

1 Introduction

The 3-dimensional tracking of human and animal body movement is important in various areas of science. Researchers, for example in the fields of biology, medicine, robotics, or sports, investigate such data to reveal patterns and complex interaction rules in natural motion. Since the precision and the availability of motion tracking technology is increasing, intelligent analysis methods become necessary to assist researchers in identifying relevant information in large amounts of data. Although the raw data usually consists of 3-dimensional vectors, the data precision and characteristics vary depending on the kind of tracking system. Today, many kinds of systems are available, ranging from large expensive motion capturing setups involving several distributed cameras and delivering very robust data at a high spatial resolution, to less sophisticated, small, cheap, and mobile solutions using only a single camera. Hence, there is a variety of options available for researchers to gather motion data. However, regarding the automatic analysis of this complex data, there is no general recipe in order to extract high-level information. Tools for clustering and visualization (see overviews in e.g. [1], and [2, chap. 10]) are widely applicable and can make the data accessible for experts in order to gain motor-functional insights from complex motion scenarios. In this context, metric learning algorithms, as presented in [3, 4, 5], offer useful features. On the one hand, the prototype-based clustering technique can be used to categorize motion patterns, yielding a classifier for later recorded data, while the resulting prototypes may reveal typical poses or patterns, since they can be interpreted directly. On the other hand, with the addition of metric learning, the most relevant joint angles or spatial correlations can be identified automatically.

In this report, we briefly present first experiments in which we applied the metric learning extensions of learning vector quantization, see [3, 4], on a small data set of human poses, and a motion sequence, recorded with the single-camera tracking system *Kinect*[1] from Microsoft.

[1] http://research.microsoft.com/en-us/um/redmond/projects/kinectsdk/

2 Motion Data Representation

Considering a general setting, the tracking data is usually given as a sequence of 3-dimensional vectors over a certain number of time steps, in the following referred to as *frames*. Each vector represents the positions of certain points on the target body in a steady coordinate system defined by the tracking device, in the following referred to as the *world coordinate system*. In partially rigid bodies of animals or humans, the movement is constrained by the underlying skeleton and the capabilities of the joints. Rigid parts, called *segments* or *bones* are connected by flexible joints characterized by their *degrees of freedom* (DoF) and their *motion range*. Therefore, it is sufficient to track only a few points (*markers*) on the body, and model its skeletal properties based on prior knowledge about the tracking target, instead of tracking a high-resolution point cloud, for example. Usually, for every frame, the locations of the joints are calculated from the marker positions, but some markerless tracking devices yield joint positions directly, like in our technical setup.

Kinect, the single-camera system which we use, provides an RGB image and a depth view of the scene. To access the information, we used the software OpenNI[2], and the middleware NiTE[3] which infers a human skeleton structure by depth and texture cues only, without the need for special physical tracking markers, like reflective dots on the target body. NiTE & OpenNI provide 3D coordinates for every joint of this simplified human skeleton, see Fig. 1, and 2. In the following, we will refer to vectors of joint coordinates by the name of the joint as indicated in Fig. 2 with a top arrow ("Left" and "Right" abbreviated as L and R), e.g., $\overrightarrow{LShoulder}$. Bone vectors will be referred to with underlined names, e.g., $\underline{\texttt{LForearmBone}} = \overrightarrow{LElbow} - \overrightarrow{LHand}$, where their direction is always pointing away from the $\overrightarrow{Neck}$ in the skeletal structure.

Because of the system's technical limitations, simplifications are significant as compared to a natural skeleton: only the most important joints are considered, and some bones remain in a fixed orientation relative to each other. We utilized some restrictions for our data representations, as described below.

From the given joint positions expressed in world coordinates, we derived a

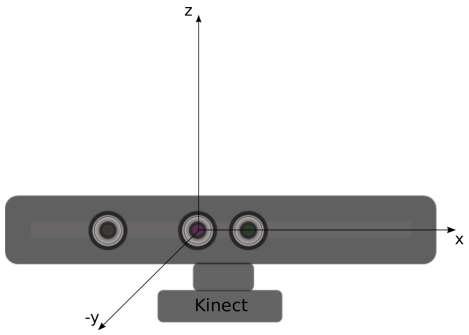


Figure 1: The world coordinate system as defined by the Kinect camera and the tracking software OpenNI & NiTE.

[2] http://openni.org
[3] http://www.primesense.com/Nite/

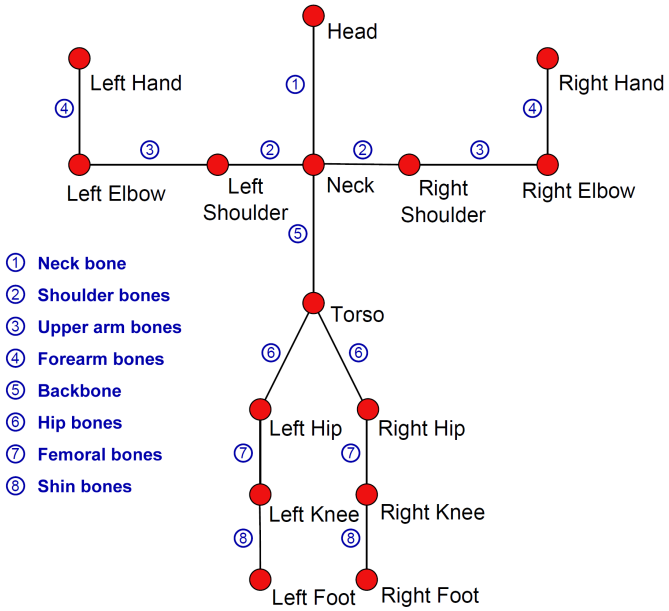


Figure 2: The joints and bones of the skeleton provided by OpenNI & NiTE (Version 1.3.0) when using a Kinect camera (view from the back).

more abstract representation, based on joint angles. This is important, since the data representation should reflect the skeleton's ability to move joints (mostly) independently from the other joints.

General representation of motion data To describe our calculation of joint angles, we first introduce the concept of a *kinematic chain*. A skeleton can be interpreted as a graph structure, commonly referred to as the kinematic chain of the body. Usually, this is an acyclic directed graph, i.e. a tree, with one joint serving as the root node. This (strictly hierarchical) structure is called an *open* kinematic chain and yields the basis for representing motion data in many technical domains, see e.g. [6] for a thorough description. Taking advantage of the kinematic chain, a corresponding hierarchy of local coordinate systems can be created, each centered at the respective node (joint). Every coordinate system is then defined by a translation and rotation of the parent node's system to the current node (the parent of the root being the world coordinate system). The translation shifts the coordinate system's origin from the parent joint to the considered joint, while the rotation of the system is predefined by the user, usually following a general scheme for all joints. For example, such a scheme might use the following steps to construct a 3-dimensional orthonormal system:

(I) align one of the axis with the bone that connects to the parent joint,

(II) use the cross product with another adjacent bone to get the second axis,

(III) again, take the cross product of these two axes to define the third axis.

Altogether, these axes define a rotation w.r.t. the previous coordinate system. Therefore, for every local coordinate system, a direct transformation based on the world coordinate system is defined as the sequential execution of all the step-wise transformations along the kinematic chain from the root to the considered joint, a central concept in *forward kinematics*, see [6] for details.

Data representation regarding the Kinect camera In the following, we concentrate on the special case of the Kinect skeleton. As mentioned, the skeleton that OpenNI & NiTE (version 1.3.0) provide based on the camera input, has some inherent limitations. Due to the lack of available documentation at the time, we recognized the following restrictive rules based on empirical evidence only[4]:

1. `NeckBone` and `Backbone` always remain in line.
2. `LShoulderBone` and `RShoulderBone` always remain in line.
3. The joints $\overrightarrow{Neck}$, $\overrightarrow{Head}$, $\overrightarrow{LShoulder}$, $\overrightarrow{RShoulder}$, $\overrightarrow{Torso}$, $\overrightarrow{LHip}$, and $\overrightarrow{RHip}$ remain in one plane. We will refer to this group of joints as the *torso group*.
4. On the plane described in rule 3, `LShoulderBone` always remains in a 90 degree angle w.r.t. `NeckBone`. (Because of the restrictions 1 and 2, this extends also to `RShoulderBone` and `Backbone`.)
5. On the plane described in rule 3, the angle between `LHipBone` and `RHipBone` in the $\overrightarrow{Torso}$ joint is about 50 degrees.

Due to this rigidity of joints in the torso group, NiTE only roughly approximates the true joint positions when tracking a pose which violates these restrictions. For every pose with joint angles that cannot be represented in the rigid torso group, NiTE will compensate by adjusting the skeleton accordingly, specifically in the angles of the neighboring joints, which sometimes leads to physically implausible angles. The bone lengths are not constant, but dynamically adapted to fit the tracked body's size and to conform to the mentioned restrictions. We know however, that the relative orientations of neck, backbone, and shoulders form a locally stable system, which we will utilize in the following to define a local coordinate system centered at the $\overrightarrow{Neck}$.

Joint angle representation In the next steps, we will derive joint angles from the given joint positions, in order to achieve a certain robustness in the data representation. The first goal is to achieve translational and rotational invariance of the entire body w.r.t. the world coordinate system: identical body poses should be represented equally, even if the tracked person is standing in a different location or orientation within the camera's field of vision. Therefore, we define a local coordinate system at the root joint in which we can represent all other joint vectors. We choose $\overrightarrow{Neck}$ as the root of our kinematic chain, and perform a translation of all other joints w.r.t. the root's position by simply subtracting the vector $\overrightarrow{Neck}$. Assuming that all bones and vectors now refer

[4] These restrictions hold up to a small numerical fluctuation of about 6° which we measured.

to the translated ones, the orientation of the neck's coordinate system is then determined by the following vectors, which form the axes of a right-handed coordinate system:

- X-axis: $\overrightarrow{\mathbf{x}_{\texttt{Neck}}} = \mathsf{norm}\,(\underline{\texttt{LShoulderBone}})$
- Y-axis: $\overrightarrow{\mathbf{y}_{\texttt{Neck}}} = \mathsf{norm}\,(\underline{\texttt{LShoulderBone}} \times \underline{\texttt{Backbone}})$
- Z-axis: $\overrightarrow{\mathbf{z}_{\texttt{Neck}}} = \overrightarrow{\mathbf{x}_{\texttt{Neck}}} \times \overrightarrow{\mathbf{y}_{\texttt{Neck}}}$

where the operation $\mathsf{norm}\,(\cdot)$ normalizes the vectors to unit length and $\times$ refers to the cross product. From the previously mentioned restriction rules 1 and 2, we know that this yields a consistent orthonormal system for every time step. Assembling the axes vectors as columns in a rotation matrix M, every joint vector can now be rotated by multiplying it with M. Thus, they are represented w.r.t. the new local coordinate system, with the origin at the $\overrightarrow{Neck}$. This provides the desired translational and rotational invariance w.r.t. the world coordinate system.

The next goal is to represent the joints (mostly) independently from each other, which is not given when using the world coordinates or the newly defined local coordinates. For example, a movement of the shoulder while the elbow joint of the person remains rigid, would result in a change of the coordinates of both, the hand and the elbow. However, from a data analysis perspective, the 'cause' of the movement should be represented independently, without adding it's 'effects' to other data dimensions, as long as no information is lost. Considering a joint angle representation for this example, the shoulder movement would alter only the shoulder angles, without involving the elbow angle (that represents the hand's relative position for the eventual data analysis). In general, we would like to represent joint vectors independently from their respective parent joint. To achieve this, we might want to generalize the above procedure of constructing local coordinate systems, and continue this scheme along the paths in the kinematic chain. However, our approach was based on two adjacent joint vectors (LShoulderBone and Backbone), serving as linearly independent basis vectors to define a plane, whereupon the cross product is orthonormal. Their linear independence was guaranteed by the named restriction rules. Now, we cannot depend on two adjacent bones being independent. As a compromise, we therefore refer to one of the parent node's coordinate axes ($\overrightarrow{\mathbf{z}_{[\texttt{parent}]}}$) to create the new coordinate system. The axes are then defined as:

- X-axis: $\overrightarrow{\mathbf{x}_{[\texttt{joint}]}} = \mathsf{norm}\left(\overrightarrow{[\texttt{joint}]} - \overrightarrow{[\texttt{parent}]}\right)$
- Y-axis: $\overrightarrow{\mathbf{y}_{[\texttt{joint}]}} = \mathsf{norm}\left(\overrightarrow{\mathbf{z}_{[\texttt{parent}]}} \times \overrightarrow{\mathbf{x}_{[\texttt{joint}]}}\right)$
- Z-axis: $\overrightarrow{\mathbf{z}_{[\texttt{joint}]}} = \overrightarrow{\mathbf{x}_{[\texttt{joint}]}} \times \overrightarrow{\mathbf{y}_{[\texttt{joint}]}}$

where [joint] stands for the currently considered joint vector, and [parent] refers to its parent (both after a translation w.r.t. $\overrightarrow{[joint]}$). There is still a problem for the special case that $\overrightarrow{\mathbf{x}_{[\texttt{joint}]}}$ equals $\overrightarrow{\mathbf{z}_{[\texttt{parent}]}}$ or $\overrightarrow{\mathbf{x}_{[\texttt{joint}]}}$ equals $-\overrightarrow{\mathbf{z}_{[\texttt{parent}]}}$, and thus the cross product yields the zero vector. Then, we instead define the Y-axis as: $\overrightarrow{\mathbf{y}_{[\texttt{joint}]}} = \overrightarrow{\mathbf{y}_{[\texttt{parent}]}}$. The angles of the joint are therefore not entirely independent from the parent's angles, wherefore our goals for the data representation could be fulfilled only partially.

Given a joint's position in the local coordinate system of the respective parent, we can now represent the orientation of their connecting bone in terms of spherical coordinates, i.e. angles, where ϕ defines the rotation around the Z-axis and θ is the elevation from the X-Y-plane. Since the joints in the torso group are mostly rigid, we are only interested in representing the joints of the four limbs. We know, that the total degrees of freedom and motion range in the limb joints are so far limited, that the two angles at each joint are enough to fully represent their movement. In the experiments presented in section 4, we use this local angle representation as input to train the GRLVQ and GMLVQ models.

3 Metric Adaptation in Clustering

To analyze the data, we applied two supervised prototype-based clustering algorithms which feature an inherent metric adaptation:

(I) *Generalized Relevance Learning Vector Quantization* (GRLVQ), see [3],

(II) *Generalized Matrix Learning Vector Quantization* (GMLVQ), see [4].

Both are based on the Generalized LVQ (GLVQ) [7] algorithm, that minimizes the cost function E to find prototype positions $\mathbf{w} \in W$ which quantize the data vectors $\mathbf{v} \in V$ according to the data's given class labeling:

$$E(W) = \frac{1}{2} \sum_{\mathbf{v} \in V} f(\mu(\mathbf{v})) \text{ with } \mu(\mathbf{v}) = \frac{d^+(\mathbf{v}) - d^-(\mathbf{v})}{d^+(\mathbf{v}) + d^-(\mathbf{v})} \tag{1}$$

Here, $d^+(\mathbf{v})$ is the distance of a vector $\mathbf{v}$ to its nearest correct prototype, i.e., carrying the same label, and $d^-(\mathbf{v})$ is the distance to its nearest incorrect prototype, i.e., associated with a different label. f is some monotonic function, like the sigmoid for example. Typically, a stochastic gradient descent is used to iteratively minimize the costs by adapting the prototype positions $w \in W$. This results in intuitive update rules resembling Hebbian learning principles.

In the extensions GRLVQ and GMLVQ, which feature metric adaptation, the distances $d^+(\mathbf{v})$ and $d^-(\mathbf{v})$ in the cost function (1) are exchanged by $d^+_{\mathbf{\Lambda}}(\mathbf{v})$ and $d^-_{\mathbf{\Lambda}}(\mathbf{v})$, respectively, which use a Mahalanobis-like scaling of the pairwise distances by a positive semidefinite matrix $\mathbf{\Lambda} = \mathbf{\Omega}\mathbf{\Omega}^{\mathrm{T}}$. Assuming that $\mathbf{w}$ is the closest correct prototype to data $\mathbf{v}$, the distance $d^+_{\mathbf{\Lambda}}(\mathbf{v})$ is defined as

$$d^+_{\mathbf{\Lambda}}(\mathbf{v}) = (\mathbf{v} - \mathbf{w})^{\mathrm{T}} \mathbf{\Lambda} (\mathbf{v} - \mathbf{w}) \quad \text{with} \quad \mathbf{\Lambda} = \mathbf{\Omega}\mathbf{\Omega}^{\mathrm{T}} \ .$$

The same definition holds for $d^-_{\mathbf{\Lambda}}(\mathbf{v})$, if $\mathbf{w}$ is the closest incorrectly labeled prototype. For GRLVQ, $\mathbf{\Omega}$ is a diagonal matrix, thus enabling a scaling of each dimension's contribution to the distance, i.e., the *relevance* of every feature in the vectorial data representation. In case of GMLVQ, $\mathbf{\Omega}$ is a full matrix, scaling additionally any pairwise correlation between the data dimensions. Apart from the substituted distance, the prototype updates are in both cases identical to the GLVQ algorithm. However, regarding the learning of the metric, the gradient descent optimization scheme is extended to the matrix $\mathbf{\Omega}$. Therefore, adaptation rules for the metric are constructed by deriving the cost function w.r.t. $\mathbf{\Omega}$. Thus, the algorithm performs in every iteration two separate updates: (i) the

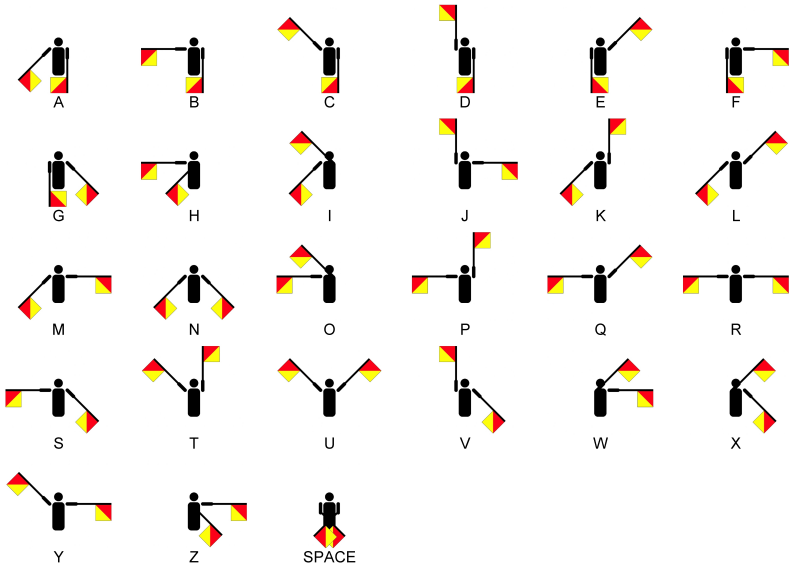


Figure 3: Flag semaphore signals. The 'Space' signal was not used in our recorded data set. (Picture taken from Wikimedia Commons [8].)

prototype update with a fixed metric, and (ii) a metric adaptation with fixed prototypes. While the resulting clustering and classification model is useful for data analysis, especially since the found prototypes can be interpreted directly, the adapted metric may also offers insight about the underlying processes behind the data.

4 Experiments

As a simple proof-of-concept study about the analysis of motion data with the GRLVQ and GMLVQ clustering methods, we used two small data sets where the expected outcome is clear.

Flag semaphore data The first dataset consists of 26 static poses, which can be distinguished by the shoulder angles only. The poses are taken from the *flag semaphore*, a code which can be used to communicate at a distance by means of visual signals, common in the maritime world prior to the Morse code. The signaling person would usually hold flags, rods, or paddles in their hands for better visibility, however, these would not encode any information. Instead, certain constellations of arm orientations represent the letters in the alphabet from A to Z, see Figure 3. The 26 poses (we omitted any special signals like the 'Space' signal) have been recorded from 4 different test subjects, resulting in 104 total data points with 4 samples per class. We trained the GRLVQ algorithm [3] on the local joint angle representation, achieving a classification accuracy of 100%. We used only the movable joints outside of the torso group. The relevance profile, shown in Figure 4, clearly singles out the angles of the left and right elbow elevation as the most important for the class separation, which

matches our expected response of the algorithm. Since the relevance learning scheme finds merely some possible configuration to separate the classes, the other angles are also contributing partially. It is fair to assume, that in a data set, where the underlying principle of the class separation was not priorly known, an expert could use this method to gain knowledge, especially when the number of joints is much larger, as it is common with high-tech motion capturing systems.

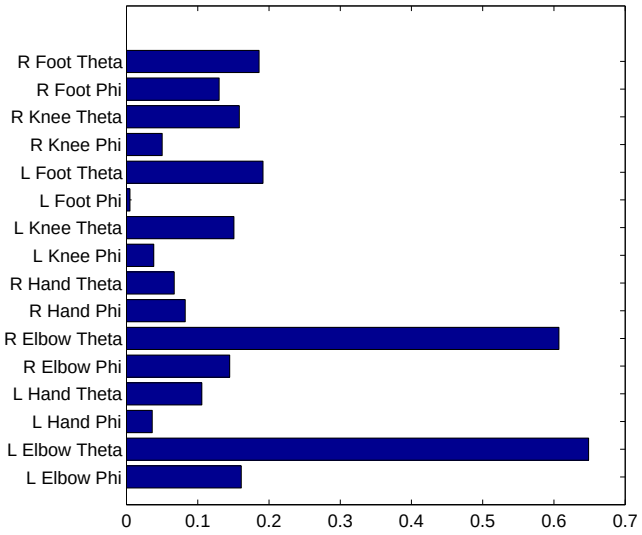


Figure 4: The bar plot shows the diagonal of the matrix Ω, i.e. the relevance profile of the GRLVQ classifier model, for the flag semaphore data set.

Walking-sequence data The second data set consists of four short sequences, showing two different walking styles, recorded from two different persons. The two walking behaviors represent our classes in the data, which are somewhat antagonistic w.r.t. joint angle progressions: The first is a normal straight human walk, where the left arm and right leg move in one direction at the same time (e.g. forward), while the right arm and left leg move in the opposite direction (e.g. backwards). The other walking style uses the opposite (unnatural) combination, where the left arm and left leg move in the same direction at a time, and the right limbs in the opposite direction at the same time. All sequences together consist of 265 frames, which were recorded at a rate of 30 Hertz, they show about 3 strides of each walking style per person. We used the joint angles of each frame as a separate data sample, without any handling or preprocessing of the time-series aspects in the data. The class labels per frame correspond to the walking style. The training with GMLVQ yields a classification accuracy of 82 percent, the resulting matrix Ω is shown in Figure 5. Note, that the sign of the values in the matrix are not really meaningful and interpretable as the respective positive or negative correlations of the joint angles in either one of the classes. Instead, only the absolute values in the matrix say, how much the pairwise correlation of these dimensions was utilized for the class separation found

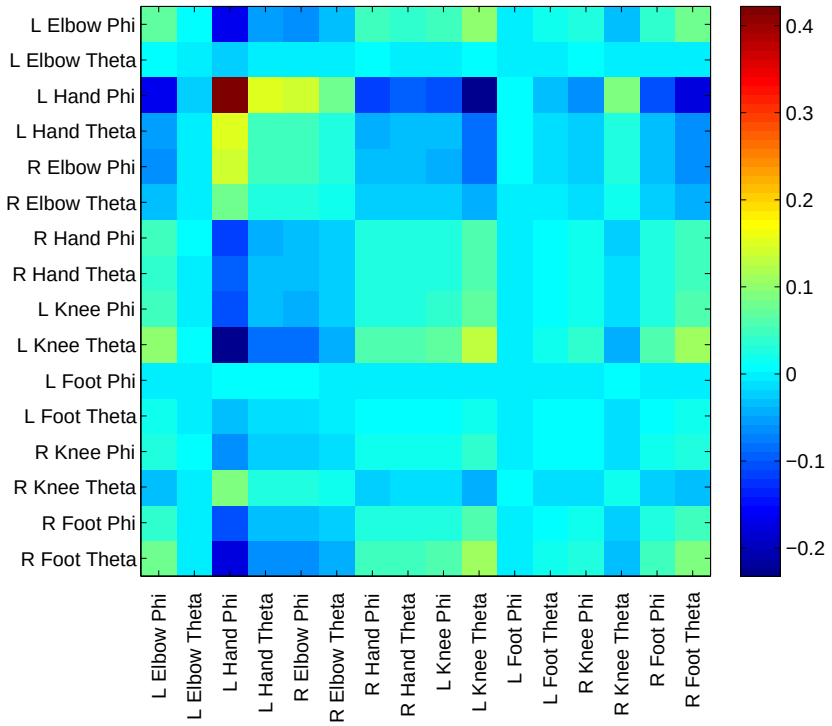


Figure 5: The matrix Ω of the GMLVQ classifier model trained on the walking-sequence data set. Correlations of the left and right limb angles have high absolute values, and are thus utilized more in the problem-adapted metric of the trained classifier.

by the classifier model, which might translate to semantic meaning regarding the classes. As expected in this case, a pattern of correlations of the left and right limb angles is clearly visible, with strongly expressed correlations between the left and right knee and elbow angles, for instance.

5 Conclusions

We showed how it is possible to derive (simplified) joint angles from a strongly constricted motion tracking skeleton provided by OpenNI & NiTE and a Kinect camera. Furthermore, we demonstrated in simple experiments, that the training of GRLVQ and GMLVQ on a joint angle representation of motion data yields plausible results with an adapted metric, providing an interpretable classification model. Metric learning in general clearly offers an interesting perspective for motion data analysis, since the semantic interpretation of the metric offers new ways for the analysis by experts.

References

[1] John A. Lee and Michel Verleysen. *Nonlinear dimensionality reduction*. Springer, 2007.

[2] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley, New York, 2. edition, 2001.

[3] Barbara Hammer and Thomas Villmann. Generalized relevance learning vector quantization. *Neural Netw.*, 15:1059–1068, October 2002.

[4] Petra Schneider, Michael Biehl, and Barbara Hammer. Distance learning in discriminative vector quantization. *Neural Computation*, 21:2942–2969, 2009.

[5] Andrej Gisbrecht and Barbara Hammer. Relevance learning in generative topographic mapping. *Neurocomput.*, 74:1351–1358, April 2011.

[6] Meinard Müller. *Information retrieval for music and motion*. Springer, 2007.

[7] Atsushi Sato and Keiji Yamada. Generalized learning vector quantization. In David S. Touretzky, Michael Mozer, and Michael E. Hasselmo, editors, *NIPS*, pages 423–429. MIT Press, 1995.

[8] `http://commons.wikimedia.org/wiki/File:Semaphore_Signals_A-Z.jpg`.

Theory of Patch Clustering for Variants of Fuzyy c-Means, Fuzzy Neural Gas, and Fuzzy Self-Organizing Map

T. Villmann*, M. Kästner, and M. Lange

Computational Intelligence Group,
University of Applied Sciences Mittweida,
Technikumplatz 17, 09648 Mittweida, Germany

Abstract

In this paper we consider the handling of the batch variant of the fuzzy-c-means algorithm as well as its extensions fuzzy neural gas and fuzzy self-organizing map in case of very large data sets. For those data sets the resulting dissimilarity matrix is huge and, therefore, cannot be handled due to memory restrictions. To solve the problem, we transfer the idea of patch learning introduced for batch neural gas to these algorithms. This leads also to a reduction of the computational complexity. We give in this article the theoretical justification of the new patch fuzzy neural gas algorithm as well as fuzzy self-organizing map.

1 Introduction

Clustering of very large data sets is still a crucial problem. Recently, new ideas for that problem were proposed for prototype based crisp vector quantizers: The so-called patch learning, introduced for the batch types of the

*corresponding author, *email: thomas.villmann@hs-mittweida.de*

c-means, the neural gas and the self-organizing map, offers a possibility to deal with such complex data. Batch variants of vector quantizers usually take all available data at once, which leads to serious problems in case of very large data sets. To deal with these problems, for patch clustering the data are divided into small patches, which can easily be handled subsequently. Thereby, the information already learned from earlier patches is fed into the learning of the current patch.

Related to these crisp vector quantizers are their fuzzy counterparts: fuzzy-c-means (FCM,[2, 7]), fuzzy neural gas (FNG) and fuzzy self-organizing maps (FSOM,[21, 20, 22]). In this paper we transfer the idea of patch learning proposed in [1] to these fuzzy vector quantization algorithms such that these approaches can be applied to very large data sets, too.

2 Fuzzy Vector Quantization

In this section we briefly provide the basic principles of FCM, FNG and FSOM to clarify notations. Thereby, we assume a data set $V = \{\mathbf{v}_i\}_{i=1}^{N} \subseteq \mathbb{R}^n$ and a set $W = \{\mathbf{w}_k\}_{k=1}^{C} \subset \mathbb{R}^n$ of prototypes. Further, we suppose an inner product norm $d_{i,k} = d(\mathbf{v}_i, \mathbf{w}_k)$ between data and prototypes, frequently chosen as the Euclidean distance.

2.1 The Basic Algorithms - batch modes

Following [23], a very general formulation in terms of a cost function to be minimized is given by

$$E(\mathbf{U}, \mathbf{T}, V, W, \delta, m, \eta, a, b) = \sum_{k=1}^{C} \sum_{i=1}^{N} \left(a \cdot u_{i,k}^{m} + b \cdot t_{i,k}^{\eta}\right) (d_{i,k})^2 + \sum_{k=1}^{C} \left(\delta_k \sum_{i=1}^{N} (t_{i,k} - 1)^{\eta}\right) \tag{1}$$

with $\mathbf{U} = \{u_{i,k}\}$ the matrix of the fuzzy assignments $u_{i,k} \in [0,1]$, $\mathbf{T} = \{t_{i,k}\}$ the matrix of the typicality assignments $t_{i,k} \in [0,1]$, m is the fuzziness parameter and η is the typicality parameter. The fuzzy assignments fulfill the constraints

$$\sum_{k} u_{i,k} = 1 \tag{2}$$

whereas for the typicality assignments

$$\sum_{i} t_{i,k} = 1 \tag{3}$$

is required. The parameters $\delta_k > 0$ are user defined and the values $a, b \geq 0$ are balancing parameter. The standard fuzzy-c-means model is obtained for $a = 1$ and $b = 0$ (FCM,[2, 7]). For the possibilistic c-means (PCM, [15]) $a = 0$ and $b = 1$ is valid, whereas fuzzy PCM is achieved as a merge of FCM and PCM for the case that both $a = 1$and $b = 1$ [19]. In the following we make use of the abbreviation $E(\mathbf{U}, \mathbf{T}, W) = E(\mathbf{U}, \mathbf{T}, V, W, \delta, m, \eta, a, b)$ to emphasize the parameters to be optimized.

The learning takes place as an alternating update scheme between prototypes and assignment variables. The prototype update is

$$\mathbf{w}_k = \frac{\sum_{i=1}^{N} \left(a \cdot u_{i,k}^m + b \cdot t_{i,k}^\eta\right) \mathbf{v}_i}{\sum_{i=1}^{N} \left(a \cdot u_{i,k}^m + b \cdot t_{i,k}^\eta\right)} \tag{4}$$

and the fuzzy assignments are updated according to

$$u_{i,k} = \frac{1}{\sum_{l=1}^{C} \left(\frac{d_{i,k}}{d_{i,l}}\right)^{\frac{2}{m-1}}} \tag{5}$$

whereas the typicality assignments are modified to

$$t_{i,k} = \frac{1}{1 + \left(\frac{\left(d_{i,k}\right)^2}{\delta_k}\right)^{\frac{1}{\eta-1}}} \tag{6}$$

taking the δ_i-values into account. As pointed out in [16], it is recommended to initialize the PCM by FCM and to determine the δ_i-values as

$$\delta_k = K \frac{\sum_{i=1}^{N} \left(u_{i,k}\right)^\eta \left(d_{i,k}\right)^2}{\sum_{i=1}^{N} \left(u_{i,k}\right)^\eta} \tag{7}$$

with the $u_{i,k}$ obtained from FCM and setting $\eta = m$.

Many other variants are proposed such as FCM for relational data [3] or median clustering [8] or using several kinds of dissimilarities like divergences [13, 24] or kernels [12]. A comprehensive overview can be found in [18].

2.2 Fuzzy Self-Organizing Map and Fuzzy Neural Gas

The fuzzy self-organizing map (FSOM, [5, 4, 6, 20, 21, 22]) and the fuzzy neural gas (FNG,[23]) combine the idea of fuzzy vector quantization with

neighborhood cooperativeness for learning improvement.. The FSOM supposes an external topological structure A between the prototypes, which defines a dissimilarity $d_A(k,l)$ between the prototypes $\mathbf{w}_k$ and $\mathbf{w}_l$ with respect to the structure A. Usually, the structure A is assumed to be a regular hypercubical lattice as known from SOMs, and d_A is taken as the Euclidean distance in A for that case. The neighborhood cooperativeness is realized by the neighborhood function

$$h_\sigma^{SOM}(k,l) = c_\sigma \cdot \exp\left(-\frac{(d_A(k,l))^2}{2\sigma^2}\right) \tag{8}$$

with neighborhood range σ and the constraint $\sum_l h_\sigma^{SOM}(k,l) = 1$ ensured by the constant c_σ. Using the concept of local costs

$$lc_\sigma^{SOM}(i,k) = \sum_{l=1}^{C} h_\sigma^{SOM}(k,l) \cdot \left(d_{i,k}^E\right)^2 \tag{9}$$

introduced by [11], an similar methodology can be applied, if the neighborhood between prototypes is determined on the basis of the ranked distances between them in the data space, as known from neural gas (NG):

$$h_\sigma^{NG}(k,l) = c_\sigma^{NG} \cdot \exp\left(-\frac{(rk_k(\mathbf{w}_l, W))^2}{2\sigma^2}\right) \tag{10}$$

for a given neighborhood range σ. The constraint $\sum_l h_\sigma^{NG}(k,l) = 1$ again is ensured by a constant c_σ^{NG} as before. This neighborhood function is based on the rank function

$$rk_k(\mathbf{w}_i, W) = \sum_{l=1}^{N} \Theta\left(d(\mathbf{w}_i, \mathbf{w}_k) - d(\mathbf{w}_i, \mathbf{w}_l)\right) \tag{11}$$

where

$$\Theta(x) = \begin{cases} 0 & if \quad x \leq 0 \\ 1 & else \end{cases} \tag{12}$$

is the Heaviside function [17]. The respective local costs are

$$lc_\sigma^{NG}(i,k) = \sum_{l=1}^{C} h_\sigma^{NG}(k,l) \cdot (d_{i,l})^2 . \tag{13}$$

A cost function $E_{FNG/FSOM}(\mathbf{U},\mathbf{T},W)$ for unsupervised fuzzy clustering with neighborhood cooperativeness according to SOM or NG is then defined as

$$E_{FNG/FSOM}(\mathbf{U},\mathbf{T},W)=\sum_{k=1}^{C}\sum_{i=1}^{N}\left(a\cdot u_{i,k}^{m}+b\cdot t_{i,k}^{\eta}\right)lc_{\sigma}^{SOM/NG}(i,k)+\sum_{k=1}^{C}\left(\delta_k\sum_{i=1}^{N}(t_{i,k}-1)^{\eta}\right) \tag{14}$$

with the critical points in the alternating optimization fulfilling the requirement that the derivative

$$\frac{\partial E_{FNG/FSOM}(\mathbf{U},\mathbf{T},W)}{\partial\mathbf{w}_k}=2\sum_{i=1}^{N}\sum_{l=1}^{C}\left(a\cdot u_{i,l}^{m}+b\cdot t_{i,l}^{\eta}\right)\cdot h_{\sigma}^{SOM/NG}(k,l)\cdot\frac{\partial d_{i,k}}{\partial\mathbf{w}_k} \tag{15}$$

has to be zero. The prototype adaptation reduces to

$$\mathbf{w}_k=\frac{\sum_{i=1}^{N}\sum_{l=1}^{C}\left(a\cdot u_{i,l}^{m}+b\cdot t_{i,l}^{\eta}\right)\cdot h_{\sigma}^{SOM/NG}(k,l)\cdot\mathbf{v}_i}{\sum_{i=1}^{N}\sum_{l=1}^{C}\left(a\cdot u_{i,l}^{m}+b\cdot t_{i,l}^{\eta}\right)\cdot h_{\sigma}^{NG}(k,l)} \tag{16}$$

if the Euclidean distance is used for $d_{i,k}$. The adaptation of the fuzzy assignments $u_{i,l}^{m}$ (5) and typicality assignments $t_{i,l}^{m}$ (6) remain structurally unchanged, but replacing there the dissimilarity measure $(d_{i,k})^2$ by the local costs $lc_{\sigma}^{SOM/NG}(i,k)$ such that

$$u_{i,k}=\frac{1}{\sum_{l=1}^{C}\left(\frac{lc_{\sigma}^{NG/SOM}(i,k)}{lc_{\sigma}^{NG/SOM}(i,l)}\right)^{\frac{1}{m-1}}} \tag{17}$$

and

$$t_{i,k}=\frac{1}{1+\left(\frac{lc_{\sigma}^{NG/SOM}(i,k)}{\delta_k}\right)^{\frac{1}{\eta-1}}} \tag{18}$$

are valid.

For vanishing neighborhood parameter $\sigma\searrow 0$ the original algorithms are obtained.

3 Median and Relational Variants

If the data $\mathbf{v}_l$ are not metric objects but the dissimilarities $D_{j,l}$ between them are known, so called *median variants* of (fuzzy) vector quantization algorithms can be applied. For these median variants the prototypes have

been restricted to be data objects itself. Hence, the dissimilarity $d_{l,k}$ between a prototype $\mathbf{w}_k$ and a data object $\mathbf{v}_l$ is a certain data dissimilarity $D_{l,j}$ if the prototype is identified with the data object $\mathbf{v}_j$. Updating prototypes then is realized setting the prototypes to the so-called generalized median [14]

$$\mathbf{w}_k = \mathbf{v}_l \text{ with } l = argmin_{l'} \left(\sum_j \left(a \cdot u_{j,l'}^m + b \cdot t_{j,l'}^\eta \right) lc_\sigma^{NG/SOM}(j, l') \right) \quad (19)$$

whereas the adjustment of the assignment variables $u_{i,l}$ and typicality assignments $t_{i,l}$ are kept according to (17) and (18). For $a = 1$ and $b = 0$ the standard median fuzzy-c-means is obtained [8].

The restriction of the prototypes to data objects in median variants may lead to serious additional distortions, in particular, if the potential data space is sparsely covered by the data. This drawback can be reduced, if the data are supposed to be embeddable into the Euclidean space but still only the dissimilarities $d_{i,k}$ between them are known, as before. In this case *relational variants* of vector quantizers can be used [9, 10], which assume that the prototypes can be written as a convex linear combination of the data objects $\mathbf{v}_j$

$$\mathbf{w}_i = \sum_j a_{i,j} \mathbf{v}_j \text{ with } \sum_j a_{ij} = 1 \quad (20)$$

with the matrix $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{C \times N}$. It turns out that dissimilarity between the ith data object and the prototype $\mathbf{w}_k$ can be expressed

$$d_{i,k} = \left(\mathbf{a}_i \cdot \mathbf{D}^{\star 2} \right)_k - \frac{1}{2} \mathbf{a}_i \cdot \mathbf{D}^{\star 2} \cdot \mathbf{a}_i^\top \quad (21)$$

with $\mathbf{D}^{\star 2} = (D_{j,l})^2$, i.e. the dissimilarities can be calculated without knowing the explicit form of the data objects. Similarly, one has

$$d_{k,l}^W = \mathbf{a}_k \cdot \mathbf{D}^{\star 2} \mathbf{a}_l^\top - \frac{1}{2} \mathbf{a}_k \cdot \mathbf{D}^{\star 2} \cdot \mathbf{a}_k^\top - \frac{1}{2} \mathbf{a}_l \cdot \mathbf{D}^{\star 2} \cdot \mathbf{a}_l^\top \quad (22)$$

for the dissimilarity between prototypes [9]. Prototype adaptation now is realized by the adaptation of the coefficients. Combining the relational fuzzy-c-means approach from [10] and the relational neural gas algorithm provided in [9] we obtain the update

$$a_{i,k} = \frac{\sum_{l=1}^C \left(a \cdot u_{i,l}^m + b \cdot t_{i,l}^\eta \right) \cdot h_\sigma^{SOM/NG}(k, l)}{\sum_{j=1}^n \sum_{l=1}^C \left(a \cdot u_{j,l}^m + b \cdot t_{j,l}^\eta \right) \cdot h_\sigma^{SOM/NG}(k, l)} \quad (23)$$

which then allows a recalculation of the dissimilarities $d_{k,l}$ and $d^W_{i,k}$. These are needed for the subsequent recalculation of the fuzzy assignments $u_{i,l}$ and typicality assignments $t_{i,l}$ via the formulae (17) and (18) using the local costs (9) and (13), respectively for FSOM and FNG.

4 Patch Clustering

We now are ready to turn to the idea of patch clustering [1]. We partition the data set into N_p patches P_i of size $p = \frac{N}{N_p}$. Let $\mathbf{D}$ be the matrix of data similarities $D_{i,j}$ and $\mathbf{D}_k$ the dissimilarity matrix for patch P_k. The idea is to learn prototypes subsequently from the patches. However, prototypes learned in a certain step k are fed into the next patch as additional data points such that an extended patch P^*_{k+1} is obtained with dissimilarities $\mathbf{D}^*_{k+1}$. However, these new data points (former prototypes) already collected implicit information such that their influence should be weighted. Following [1], this can be implemented assigning to each data object $\mathbf{v}_j$ a multiplicity μ_j, which is set $\mu_j = 1$ for original data objects $\mathbf{v}_j$ and $\mu_j = \omega_l\,(k)$ if the prototype $\mathbf{w}_l$ from the kth step is used as new data object. Here the prototype multiplicity ω_l is defined as

$$\omega_l = \frac{1}{a+b} \sum_{i=1}^{N} \left(a \cdot u_{i,l} + b \cdot t_{i,l}\right) \cdot \mu_i$$

which reduces to number of data points represented by the respective prototype in case of crisp c-means.

These multiplicities are used in prototype adaptation. In this way we get for the median variant (19)

$$\mathbf{w}_k = \mathbf{v}_l \text{ with } l = argmin_{l'} \left(\sum_j \mu_j \cdot \left(a \cdot u^m_{j,l'} + b \cdot t^\eta_{j,l'}\right) lc^{NG/SOM}_\sigma\,(j,l') \right) \tag{24}$$

and for the relational variant the prototype update is determined by the new rule

$$a_{i,k} = \frac{\mu_i \sum_{l=1}^{C} \left(a \cdot u^m_{i,l} + b \cdot t^\eta_{i,l}\right) \cdot h^{SOM/NG}_\sigma\,(k,l)}{\sum_{j=1}^{n} \mu_j \sum_{l=1}^{C} \left(a \cdot u^m_{j,l} + b \cdot t^\eta_{j,l}\right) \cdot h^{SOM/NG}_\sigma\,(k,l)} \tag{25}$$

incorporating the multiplicities instead of the former rule (23).

Yet, the updates of the assignments remain unchanged as proposed in [1].

5 Conclusion

In this article we provide the theoretical framework for the combination of the fuzzy-c-means models (FCM,PCM) and its variants like fuzzy neural gas and fuzzy self-organizing map with idea of patch clustering to deal with very large data sets in case of batch learning. For this purpose, we first justify the batch variants of the general fuzzy vector quantization models for median and relational learning. Thereafter we coopt the idea of patch clustering from the crisp variants and transfer them to the fuzzy models.

References

[1] N. Alex, A. Hasenfuss, and B. Hammer. Patch clustering for massive data sets. *Neurocomputing*, 72(7-9):1455–1469, 2009.

[2] J. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum, New York, 1981.

[3] J. Bezdek, R. Hathaway, and M. Windham. Numerical comparison of RFCM and AP algorithms for clustering relational data. *Pattern recognition*, 24:783–791, 1991.

[4] J. C. Bezdek and N. R. Pal. A note on self-organizing semantic maps. *IEEE Transactions on Neural Networks*, 6(5):1029–1036, 1995.

[5] J. C. Bezdek and N. R. Pal. Two soft relatives of learning vector quantization. *Neural Networks*, 8(5):729–743, 1995.

[6] J. C. Bezdek, E. C. K. Tsao, and N. R. Pal. Fuzzy Kohonen clustering networks. In *Proc. IEEE International Conference on Fuzzy Systems*, pages 1035–1043, Piscataway, NJ, 1992. IEEE Service Center.

[7] J. Dunn. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3:32–57, 1973.

[8] T. Geweniger, D. Zühlke, B. Hammer, and T. Villmann. Median fuzzy c-means for clustering dissimilarity data. *Neurocomputing*, 73(7–9):1109–1116, 2010.

[9] B. Hammer and A. Hasenfuss. Topographic mapping of large dissimilarity data sets. *Neural Computation*, 22(9):2229–2284, 2010.

[10] R. Hathaway and J. Bezdek. NERF c-means: Non-Euclidean relational fuzzy clustering. *Pattern recognition*, 27(3):429–437, 1994.

[11] T. Heskes. Energy functions for self-organizing maps. In E. Oja and S. Kaski, editors, *Kohonen Maps*, pages 303–315. Elsevier, Amsterdam, 1999.

[12] H. Ichihashi and K. Honda. Application of kernel trick to fuzzy c-means with regularization by K-L information. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 8(6):566–572, 2004.

[13] R. Inokuchi and S. Miyamoto. Fuzzy c-means algorithms using Kullback-Leibler divergence and Hellinger distance based on multinomial manifold. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 12(5):443–447, 2008.

[14] T. K.ohonen and P. Somervuo. How to make large self-organizing maps for nonvectorial data. *Neural Networks*, 15(8-9):945–952, October-November 2002.

[15] R. Krishnapuram and J. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(4):98–110, 1993.

[16] R. Krishnapuram and J. Keller. The possibilistic c-means algorithm: insights and recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, 1996.

[17] T. M. Martinetz, S. G. Berkovich, and K. J. Schulten. 'Neural-gas' network for vector quantization and its application to time-series prediction. *IEEE Trans. on Neural Networks*, 4(4):558–569, 1993.

[18] S. Miyamoto, H. Ichihashi, and K. Honda. *Algorithms for Fuzzy Clustering*, volume 229 of *Studies in Fuzziness and Soft Computing*. Springer, 2008.

[19] N. Pal, K. Pal, J. Keller, and J. Bezdek. A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems*, 13(4):517–530, 2005.

[20] N. R. Pal, J. C. Bezdek, and E. C. K. Tsao. Generalized clustering networks and Kohonen's self-organizing scheme. *IEEE Transactions on Neural Networks*, 4(4):549–557, 1993.

[21] N. R. Pal, J. C. Bezdek, and E. C. K. Tsao. Errata to Generalized clustering networks and Kohonen's self-organizing scheme. *IEEE Transactions on Neural Networks*, 6(2):521–521, March 1995.

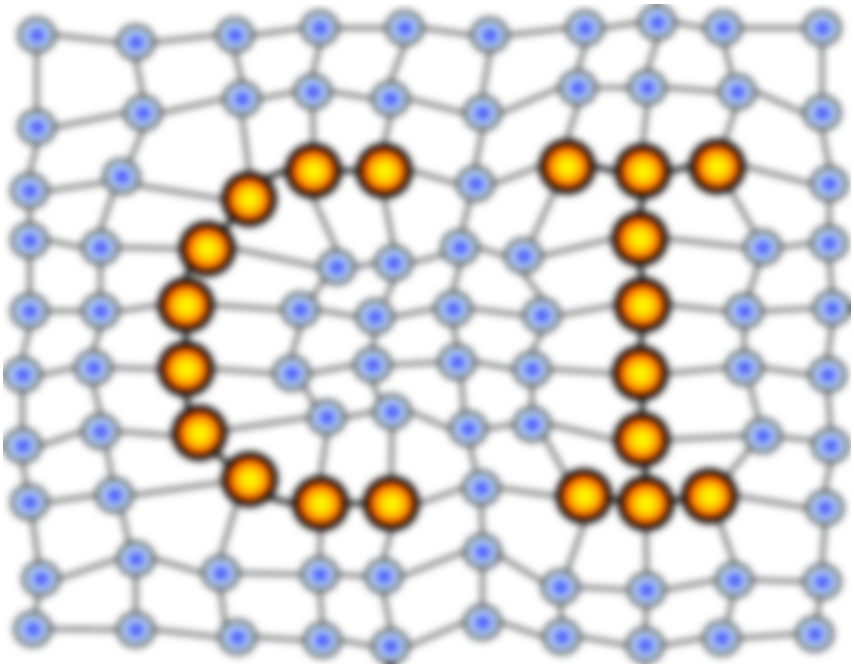
[22] E. Tsao, J. Bezdek, and N. Pal. Fuzzy Kohonen clustering networks. *Pattern Recognition*, 27(5):757–764, 1994.

[23] T. Villmann, T. Geweniger, M. Kästner, and M. Lange. Theory of fuzzy neural gas for unsupervised vector quantization. *Machine Learning Reports*, 5(MLR-06-2011):27–46, 2011. ISSN:1865-3960, http://www.techfak.uni-bielefeld.de/~fschleif/mlr/mlr_06_2011.pdf.

[24] T. Villmann and S. Haase. Divergence based vector quantization. *Neural Computation*, 23(5):1343–1392, 2011.

MACHINE LEARNING REPORTS

Report 01/2012



Impressum

Machine Learning Reports ISSN: 1865-3960

▽ Publisher/Editors

Prof. Dr. rer. nat. Thomas Villmann
University of Applied Sciences Mittweida
Technikumplatz 17, 09648 Mittweida, Germany
• http://www.mni.hs-mittweida.de/

Dr. rer. nat. Frank-Michael Schleif
University of Bielefeld
Universitätsstrasse 21-23, 33615 Bielefeld, Germany
• http://www.cit-ec.de/tcs/about

▽ Acknowledgments

We would like to thank the reviewers for their time and patience.