JAISCR

Journal of Artificial Intelligence and Soft Computing Research

2011

ISBN 978-83-62916-14-6

Volume 1, Number 1

Polish Neural Network Society
Academy of Management in Lodz (SWSPiZ)

JOURNAL OF ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING RESEARCH (JAISCR)
is published quarterly by the Polish Neural Network Society and the Academy of Management (SWSPiZ)
in Lodz, Poland.

PUBLISHING AND EDITORIAL OFFICE:
Academy of Management (SWSPiZ)
Information Technology Institute (ITI)
Sienkiewicza 9
90-113 Lodz
Tel.: +48 42 6646654
Fax.: +48 42 6366251
E-mail: jaiscr@kik.pcz.pl
URL: http://jaiscr.eu

Print: Drukarnia GREEN, pl. Komuny Paryskiej 4, 90-007 Lodz, tel.: +48 42 6322713

AIMS AND SCOPE:

Journal of Artificial Intelligence and Soft Computing Research is a refereed international journal whose focus is on the latest scientific results and methods constituting soft computing. The areas of interest include, but are not limited to:

Artificial Intelligence in Modelling and Simulation
Artificial Intelligence in Scheduling and Optimization
Bioinformatics
Computer Vision
Data Mining
Distributed Intelligent Processing
Evolutionary Design
Expert Systems
Fuzzy Computing with Words
Fuzzy Control
Fuzzy Logic
Fuzzy Optimisation
Hardware Implementations
Intelligent Database Systems
Knowledge Engineering
Multi-agent Systems
Natural Language Processing
Neural Network Theory and Architectures
Robotics and Related Fields
Rough Sets Theory: Foundations and Applications
Speech Understanding
Supervised and Unsupervised Learning
Theory of Evolutionary Algorithms
Various Applications

Contents

Preface

In the nineties of the previous century Professor Lotfi Zadeh coined the term soft computing. This laid the foundation for contemporary set of techniques which using computer hardware deal with uncertainty and imprecision. These techniques include neural networks, fuzzy logic, evolutionary algorithms, rough sets and are mainly biologically inspired by human mind and nature. Recent decades have brought a sustainable development of such methods thanks to rapid computer science progress. Their appearance and quick progress enabled the user to fully utilize newly created computational potential and to obtain knowledge out of increasing wealth of data. The development of soft computing is strictly connected with the increase of available data as well as capabilities of their processing, mutually supportive factors. Without soft computing the development of intelligent systems would be almost impossible, and their applications would be practically marginal. That is why these techniques have been especially developed in recent years.

Based on the issues described herein, a group of enthusiastic research scientists have established this peer-reviewed, international journal, under the strong support of the Polish Neural Network Society. The journal is a platform for international academic information exchange among professionals in the field of soft computing and artificial intelligence community all over the world. The first volume contains six papers. In the paper "Projective nonnegative matrix factorization based on α-divergence" by Zhirong Yang and Erkki Oja, α-Nonnegative Matrix Factorization and Projective Nonnegative Matrix Factorization are combined to produce a sparse factorizing matrix. Karim El-Laithy and Martin Bogdan, in the paper "Synchrony state generation: an approach using stochastic synapses", examine temporal synchrony within a stochastic synaptic network. Judith Redi et al. in the paper "A two-layer neural system for reduced-reference visual quality assessment" propose a soft computing system for image quality assessment. The paper "Optimizing control by robustly feasible model predictive control and application to drinking water distribution systems", by Vu Nam Tran and Mietek A. Brdys, describes a system to control quantity in drinking water distribution systems. Dimitris C. Theodoridis et al. in the paper "Robustifying analysis of the direct adaptive control of unknown multivariable nonlinear systems based on a new neuro-fuzzy method" use new neuro-fuzzy systems to control nonlinear systems. In the paper "Discovering diagnostic gene targets for early diagnosis of acute using methods of computational intelligence on gene expression data", Maurizio Fiasche et al. perform medical diagnosis based on genes using computational intelligence methods.

On behalf of our editorial board, we are pleased to announce the launch of the Journal of Artificial Intelligence and Soft Computing Research, a new non-profit official journal published by the Polish Neural Network Society and the Academy of Management (SWSPiZ) in Łódź, Poland.

Leszek Rutkowski
Editor - in - Chief
and
President of the Polish Neural Network Society

JAISCR, 2011, Vol.1, No.1, pp. 7–16

PROJECTIVE NONNEGATIVE MATRIX FACTORIZATION BASED ON α-DIVERGENCE

Zhirong Yang and Erkki Oja
Department of Information and Computer Science
Aalto University School of Science and Technology
P.O.Box 15400, FI-00076, Aalto, Finland

Abstract

The well-known Nonnegative Matrix Factorization (NMF) method can be provided with more flexibility by generalizing the non-normalized Kullback-Leibler divergence to α-divergences. However, the resulting α-NMF method can only achieve mediocre sparsity for the factorizing matrices. We have earlier proposed a variant of NMF, called Projective NMF (PNMF) that has been shown to have superior sparsity over standard NMF. Here we propose to incorporate both merits of α-NMF and PNMF. Our α-PNMF method can produce a much sparser factorizing matrix, which is desired in many scenarios. Theoretically, we provide a rigorous convergence proof that the iterative updates of α-PNMF monotonically decrease the α-divergence between the input matrix and its approximate. Empirically, the advantages of α-PNMF are verified in two application scenarios: (1) it is able to learn highly sparse and localized part-based representations of facial images; (2) it outperforms α-NMF and PNMF for clustering in terms of higher purity and smaller entropy.

Supported by the Academy of Finland in the project *Finnish Center of Excellence in Adaptive Informatics Research*.

1 Introduction

Nonnegative learning based on matrix factorization has received a lot of research attention recently. After Lee and Seung [11, 12] presented their *Nonnegative Matrix Factorization* (NMF) algorithms, a multitude of NMF variants have been proposed and applied to many areas such as signal processing, data mining, pattern recognition and gene expression studies [3, 5, 6, 9, 14, 21]. NMF is not only applicable to the feature axis for finding sparse and part-based representations (e.g.[10, 13]), but also to the sample axis, e.g. for finding clusters of data items (e.g. [8, 7, 19]).

The original NMF algorithm minimizes one of two kinds of difference measure between the data matrix and its approximate: the least square error or the non-normalized Kullback-Leibler divergence (or I-divergence). When the latter is used, NMF actually maximizes the Poisson likelihood of the observed data [11]. It was recently pointed out that the divergence minimization can be generalized by using the α-divergence [1], which leads to a family of new algorithms [4, 23]. The convergence proof of NMF with α-divergence is given in [4]. The empirical study by Cichocki et al. shows that the generalized NMF can achieve better performance by using suitable α values.

Projective Nonnegative Matrix Factorization (PNMF) [22] is another variant of NMF. It identifies a nonnegative subspace by integrating the nonnegativity to the PCA objective. PNMF has proven to outperform NMF in feature extraction, where PNMF is able to generate sparser patterns which are more localized and non-overlapping [22]. Clustering results of text data also demonstrate that PNMF is advantageous as it provides better approximation

to the binary-valued multi-cluster indicators than NMF [19].

In this paper we combine the above two techniques by using α-divergence instead of I-divergence as the error measure in PNMF. We provide a multiplicative optimization algorithm which is theoretically convergent. Experiments are conducted, in which the new algorithm is shown to outperform α-NMF for feature extraction and clustering on a variety of datasets.

Part of the work can be found in our preliminary paper [18]. As an extension, we propose here a novel multiplicative update rule which monotonically decreases the α-divergence between the data matrix and its approximate, without additional normalization or stabilization steps. The new algorithm is more desirable because it makes the objectives at different iterations and with different initial guesses comparable. The proof uses a novel convex function for α-divergence which has not been used in the previous literature on divergence measures. We also provide the multiplicative update rule for the special case $\alpha \to 0$, which completes these algorithms for the entire family of α-divergences.

The rest of the paper is organized as follows. We first briefly review the NMF and PNMF methods in Section 2. In Section 3, we present the α-PNMF objective, its multiplicative optimization algorithm and convergence proof. The experiments are presented in Section 4, and Section 5 concludes the paper.

2 Related Work

2.1 Nonnegative Matrix Factorization

Given a nonnegative data matrix $\mathbf{X} \in \mathbb{R}_+^{m\times N}$, *Nonnegative Matrix Factorization* (NMF) seeks an approximative decomposition of $\mathbf{X}$ that is of the form:

$$\mathbf{X} \approx \mathbf{WH}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times N}$ with the rank $r \ll \min(m,N)$.

Denote by $\widehat{\mathbf{X}} = \mathbf{WH}$ the approximating matrix. The approximation can be achieved by minimizing two widely used measures: (1) the least square criterion $\varepsilon = \sum_{i,j}\left(X_{ij} - \widehat{X}_{ij}\right)^2$ and (2) the non-normalized *Kullback-Leibler divergence* (or I-divergence)

$$D_{\mathrm{I}}\left(\mathbf{X}||\widehat{\mathbf{X}}\right) = \sum_{i,j}\left(X_{ij}\log\frac{X_{ij}}{\widehat{X}_{ij}} - X_{ij} + \widehat{X}_{ij}\right). \tag{2}$$

In this paper we focus on the second approximation criterion, which leads to the multiplicative updating rules of the form

$$H_{kj}^{\text{new}} = H_{kj}\frac{\left(\mathbf{W}^T\mathbf{Z}\right)_{kj}}{\sum_i W_{ik}}, \tag{3}$$

$$W_{ik}^{\text{new}} = W_{ik}\frac{\left(\mathbf{ZH}^T\right)_{ik}}{\sum_j H_{kj}}, \tag{4}$$

where we use $Z_{ij} = X_{ij}/\widehat{X}_{ij}$ for notational brevity.

2.2 Nonnegative Matrix Factorization with α-divergence

The α-divergence [1] is a parametric family of divergence functionals, including several well-known divergence measures as special cases. NMF equipped with the following α-divergence as the approximation measure was introduced by Cichocki *et al* and called α-NMF [4]:

$$D_\alpha\left(\mathbf{X}||\widehat{\mathbf{X}}\right) = \frac{\sum_{ij}\left(\alpha X_{ij} + (1-\alpha)\widehat{X}_{ij} - X_{ij}^\alpha\widehat{X}_{ij}^{1-\alpha}\right)}{\alpha(1-\alpha)} \tag{5}$$

The corresponding multiplicative update rules are given by the following, where we define $\widetilde{Z}_{ij} = Z_{ij}^\alpha$:

$$H_{kj}^{\text{new}} = H_{kj}\left[\frac{\left(\mathbf{W}^T\widetilde{\mathbf{Z}}\right)_{kj}}{\sum_i W_{ik}}\right]^{\frac{1}{\alpha}}, \tag{6}$$

$$W_{ik}^{\text{new}} = W_{ik}\left[\frac{\left(\widetilde{\mathbf{Z}}\mathbf{H}^T\right)_{ik}}{\sum_j H_{kj}}\right]^{\frac{1}{\alpha}}. \tag{7}$$

α-NMF reduces to the conventional NMF with I-divergence when $\alpha \to 1$. Another choice of α characterizes a different learning principle, in the sense that the model distribution is more inclusive ($\alpha \to \infty$) or more exclusive ($\alpha \to -\infty$). Such flexibility enables α-NMF to outperform NMF with α properly selected.

2.3 Projective Nonnegative Matrix Factorization

Replacing $\mathbf{H} = \mathbf{W}^T\mathbf{X}$ in (1), we get the *Projective Nonnegative Matrix Factorization* (PNMF) approximation scheme [22]

$$\mathbf{X} \approx \mathbf{W}\mathbf{W}^T\mathbf{X}. \tag{8}$$

Denote $\widehat{\mathbf{X}} = \mathbf{W}\mathbf{W}^T\mathbf{X}$ the approximating matrix, $Z_{ij} = X_{ij}/\widehat{X}_{ij}$, and $\mathbf{1}_m$ a column vector of length m and filled with ones. The PNMF multiplicative update rule for I-divergence is given by [22]

$$W'_{ik} = W_{ik}\frac{(\mathbf{A}\mathbf{W})_{ik}}{(\mathbf{B}\mathbf{W})_{ik}} \tag{9}$$

where $\mathbf{A} = \mathbf{Z}\mathbf{X}^T + \mathbf{X}\mathbf{Z}^T$ and $\mathbf{B} = \mathbf{1}_m\mathbf{1}_n^T\mathbf{X}^T + \mathbf{X}\mathbf{1}_n\mathbf{1}_m^T$.

In practice, iterations with only the update rule (9) are sensitive to the initial guess of $\mathbf{W}$ and often have a very zigzag learning path, where the overall scaling of $\mathbf{W}$ fluctuates between odd and even iterations. This is overcome in practice by using an additional normalization step [22]

$$\mathbf{W}^{\text{new}} = \frac{\mathbf{W}'}{\|\mathbf{W}'\|} \tag{10}$$

or a stabilization step [19]

$$\mathbf{W}^{\text{new}} = \mathbf{W}'\sqrt{\frac{\sum_{ij}X_{ij}}{\sum_{ij}\left(\mathbf{W}'\mathbf{W}'^T\mathbf{X}\right)_{ij}}}. \tag{11}$$

The name PNMF comes from another derivation of the approximation scheme (8) where a projection matrix $\mathbf{P}$ in $\mathbf{X} \approx \mathbf{P}\mathbf{X}$ is factorized into $\mathbf{W}\mathbf{W}^T$. This interpretation connects PNMF with the classical *Principal Component Analysis* subspace method except for the nonnegativity constraint [22]. Compared with NMF, PNMF is able to learn a much sparser matrix $\mathbf{W}$ [19, 22, 23]. This property is especially desired for extracting part-based representations of data samples or finding cluster indicators.

3 PNMF with α-divergence

In this section, we combine the flexibility of α-NMF and the sparsity of PNMF into a single algorithm. We call the resulting method α-*PNMF* which stands for Projective Nonnegative Matrix Factorization with α-divergence.

3.1 Multiplicative Update Rule

α-PNMF solves the following optimization problem:

$$\underset{\mathbf{W}\geq\mathbf{0}}{\text{minimize}}\ \mathcal{J}(\mathbf{W}) = D_\alpha(\mathbf{X}||\mathbf{W}\mathbf{W}^T\mathbf{X}). \tag{12}$$

The gradient of the objective with respect to $\mathbf{W}$ is given by

$$\frac{\partial\mathcal{J}(\mathbf{W})}{\partial W_{ik}} = \frac{1}{\alpha}\left[-\left(\widetilde{\mathbf{A}}\mathbf{W}\right)_{ik} + (\mathbf{B}\mathbf{W})_{ik}\right],$$

where $\widetilde{Z}_{ij} = Z_{ij}^\alpha$, $\widetilde{\mathbf{A}} = \widetilde{\mathbf{Z}}\mathbf{X}^T + \mathbf{X}\widetilde{\mathbf{Z}}^T$ and again $\mathbf{B} = \mathbf{1}_m\mathbf{1}_n^T\mathbf{X}^T + \mathbf{X}\mathbf{1}_n\mathbf{1}_m^T$.

Denote Λ_{ik} the Lagrangian multipliers associated with the constraint $W_{ik} \geq 0$. The Karush-Kuhn-Tucker (KKT) conditions require

$$\frac{\partial\mathcal{J}(\mathbf{W})}{\partial W_{ik}} = \Lambda_{ik} \tag{13}$$

and $\Lambda_{ik}W_{ik} = 0$ which indicates $\Lambda_{ik}W_{ik}^{2\alpha} = 0$. Multiplying both sides of (13) by $W_{ik}^{2\alpha}$ leads to $\frac{\partial\mathcal{J}(\mathbf{W})}{\partial W_{ik}}W_{ik}^{2\alpha} = 0$. This suggests a multiplicative update rule:

$$W_{ik}^{\text{new}} = W_{ik}\left[\frac{\left(\widetilde{\mathbf{A}}\mathbf{W}\right)_{ik}}{(\mathbf{B}\mathbf{W})_{ik}}\right]^{\frac{1}{2\alpha}}. \tag{14}$$

for all $\alpha \neq 0$. For the special case $\alpha = 0$, the update rule is given by

$$W_{ik}^{\text{new}} = W_{ik}\exp\left(\frac{1}{2}\frac{\left(\widetilde{\mathbf{A}}^{(0)}\mathbf{W}\right)_{ik}}{(\mathbf{B}\mathbf{W})_{ik}}\right), \tag{15}$$

where $\widetilde{Z}_{ij}^{(0)} = \log Z_{ij}$ and $\widetilde{\mathbf{A}}^{(0)} = \widetilde{\mathbf{Z}}^{(0)}\mathbf{X}^T + \mathbf{X}\widetilde{\mathbf{Z}}^{(0)T}$.

3.2 Convergence Proof

In this Section, we prove that iteratively applying (14) or (15) monotonically decreases the objective function $D_\alpha(\mathbf{X}||\mathbf{W}\mathbf{W}^T\mathbf{X})$.

The convergence of NMF and most of its variants, including α-NMF, to a local minimum of the cost function is analyzed by using an auxiliary function as its tight upper-bound. This is achieved in α-NMF [4] by using the Jensen inequality based on the convex function

$$h(z) = \frac{\alpha + (1-\alpha)z - z^{1-\alpha}}{\alpha(1-\alpha)}. \tag{16}$$

This convex function is however not applicable to the α-PNMF case because it is not decomposable, i.e. not fulfilling $h(xy) \propto h(x)h(y)$ or $h(xy) = h(x) + h(y) + \text{constant}$.

Here we overcome this problem by using a novel convex function

$$g(x,y) = -\frac{x^{\alpha}y^{1-\alpha}}{\alpha(1-\alpha)}. \tag{17}$$

We further introduce

$$f(y) = g\left(X_{ij}, y\right) \tag{18}$$

for notational brevity. Notice that $f(y)$ is convex with respect to y,

$$f(by) = b^{1-\alpha} f(y), \tag{19}$$

$$f(yz) = -\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}} f(y)f(z). \tag{20}$$

Let $\mathbf{W}$ be the current estimate, $\widehat{\mathbf{X}} = \widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^T\mathbf{X}$, and

$$\gamma_{ijk} = \frac{W_{ik}\left(\mathbf{W}^T\mathbf{X}\right)_{kj}}{\sum_l W_{il}\left(\mathbf{W}^T\mathbf{X}\right)_{lj}} = \frac{W_{ik}\left(\mathbf{W}^T\mathbf{X}\right)_{kj}}{\left(\mathbf{W}\mathbf{W}^T\mathbf{X}\right)_{ij}}, \tag{21}$$

$$\beta_{ajk} = \frac{W_{ak}X_{aj}}{\sum_b W_{bk}X_{bj}} = \frac{W_{ak}X_{aj}}{\left(\mathbf{W}^T\mathbf{X}\right)_{kj}}, \tag{22}$$

$$\widetilde{\mathbf{V}} \equiv \widetilde{\mathbf{V}}(\widetilde{\mathbf{W}}, \mathbf{W}),\ \widetilde{V}_{ik} = \widetilde{W}_{ik}^{1-\alpha} W_{ik}^{\alpha} \tag{23}$$

$$S_{ij} = -\frac{1}{\alpha(1-\alpha)} \widetilde{\mathbf{Z}}^T\mathbf{X} \tag{24}$$

Obviously, $\gamma_{ijk} \geq 0$, $\sum_k \gamma_{ijk} = 1$, $\beta_{ajk} \geq 0$, $\sum_a \beta_{ajk} = 1$, $\mathbf{V} \equiv \widetilde{\mathbf{V}}(\mathbf{W}, \mathbf{W})$, and $V_{ik} = W_{ik}$.

In the derivation below we also employ the following inequality for any symmetric real matrix $\mathbf{M}$ independent of $\widetilde{\mathbf{W}}$ [8]:

$$\frac{1}{2}\mathrm{Tr}\left(\widetilde{\mathbf{W}}^T\mathbf{M}\widetilde{\mathbf{W}}\right) \leq \sum_{ik} \frac{\widetilde{W}_{ik}^2}{2W_{ik}} \left(\mathbf{M}\mathbf{W}\right)_{ik}, \tag{25}$$

where the equality holds if and only if $\widetilde{\mathbf{W}} = \mathbf{W}$.

We can then apply the Jensen inequality twice to obtain the upper bound of $\mathcal{J}_1(\widetilde{\mathbf{W}}) \equiv -\sum_{ij} \frac{X_{ij}^{\alpha}\widehat{X}_{ij}^{1-\alpha}}{\alpha(1-\alpha)}$ by G_1 (see Figure 1).

The gradient of G_1 with respect to W_{ik} using the chain rule is:

$$\frac{\partial G_1}{\partial \widetilde{W}_{ik}} = \sum_{al} \frac{\partial G_1}{\partial V_{al}} \frac{\partial V_{al}}{\partial \widetilde{W}_{ik}} = \frac{\partial G_1}{\partial V_{ik}} \frac{\partial V_{ik}}{\partial \widetilde{W}_{ik}} \tag{37}$$

$$= -\frac{1}{\alpha}\left(\frac{W_{ik}}{\widetilde{W}_{ik}}\right)^{2\alpha-1} \left(\widetilde{\mathbf{A}}\mathbf{W}\right)_{ik} \tag{38}$$

Recall $\mathbf{B} = \mathbf{1}_m\mathbf{1}_n^T\mathbf{X}^T + \mathbf{X}\mathbf{1}_n\mathbf{1}_m^T$. We have

$$\mathcal{J}_2(\widetilde{\mathbf{W}}) \equiv \sum_{ij} \frac{1}{\alpha}\left(\widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{ij} \tag{39}$$

$$= \frac{1}{2\alpha}\mathrm{Tr}\left[\widetilde{\mathbf{W}}^T\mathbf{B}\widetilde{\mathbf{W}}\right] \tag{40}$$

$$\leq \frac{1}{\alpha}\sum_{ik} \frac{\widetilde{W}_{ik}^2}{2W_{ik}} \left(\mathbf{B}\mathbf{W}\right)_{ik} \equiv G_2(\widetilde{\mathbf{W}}, \mathbf{W}) \tag{41}$$

Therefore,

$$\mathcal{J}(\widetilde{\mathbf{W}}) = \sum_{ij} \frac{X_{ij}}{1-\alpha} + \mathcal{J}_1(\widetilde{\mathbf{W}}) + \mathcal{J}_2(\widetilde{\mathbf{W}}) \tag{42}$$

$$\leq \sum_{ij} \frac{X_{ij}}{1-\alpha} + G_1(\widetilde{\mathbf{W}}, \mathbf{W}) + G_2(\widetilde{\mathbf{W}}, \mathbf{W}) \tag{43}$$

$$\equiv G(\widetilde{\mathbf{W}}, \mathbf{W}) \tag{44}$$

Minimization over $\widetilde{\mathbf{W}}$ is implemented by setting $\frac{\partial G}{\partial \widetilde{W}_{ik}} = 0$:

$$-\frac{1}{\alpha}\left(\frac{W_{ik}}{\widetilde{W}_{ik}}\right)^{2\alpha-1} \left(\widetilde{\mathbf{A}}\mathbf{W}\right)_{ik} + \frac{1}{\alpha}\frac{\widetilde{W}_{ik}}{W_{ik}} \left(\mathbf{B}\mathbf{W}\right)_{ik} = 0. \tag{45}$$

The factor $1/\alpha$ cancels when $\alpha \neq 0$, which leads to the update rule (14). When $\alpha \to 0$, we can apply L'Hôpital's rule to the both sides of (45) and obtain

$$\frac{\widetilde{W}_{ik}}{W_{ik}}\left[2\log\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right)\left(\mathbf{B}\mathbf{W}\right)_{ik} - \left(\widetilde{\mathbf{A}}^{(0)}\mathbf{W}\right)_{ik}\right] = 0. \tag{46}$$

Notice that the sequence of W_{ik} remains positive given a positive initialization. Thus we can safely remove the factor $W_{ik}/\widetilde{W}_{ik}$, resulting the update rule (15) for $\alpha \to 0$ or the inverse I-divergence. In summary,

$$\mathcal{J}(\mathbf{W}^{\text{new}}) = G(\mathbf{W}^{\text{new}}, \mathbf{W}^{\text{new}}) \tag{47}$$

$$\leq G(\mathbf{W}^{\text{new}}, \mathbf{W}) \tag{48}$$

$$\leq G(\mathbf{W}, \mathbf{W}) = \mathcal{J}(\mathbf{W}), \tag{49}$$

where the first inequality comes from the upper bound and the second by the minimization. Iteratively applying (14) thus monotonically decreases $D_{\alpha}(\mathbf{X}||\mathbf{W}\mathbf{W}^T\mathbf{X})$. □

Remark 1:

Theoretically, convergent update rules for PNMF based on the non-normalized KL-divergence

$$\mathcal{J}_1(\widetilde{\mathbf{W}}) \equiv \sum_{ij} -\frac{X_{ij}^{\alpha}\widehat{X}_{ij}^{1-\alpha}}{\alpha(1-\alpha)} = \sum_{ij} g\left(X_{ij},\widehat{X}_{ij}\right) = \sum_{ij} f\left(\widehat{X}_{ij}\right) \tag{26}$$

$$= \sum_{ij} f\left(\sum_k \widetilde{W}_{ik}\left(\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{kj}\right) = \sum_{ij} f\left(\sum_k \gamma_{ijk}\frac{\widetilde{W}_{ik}\left(\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{kj}}{\gamma_{ijk}}\right) \tag{27}$$

$$\leq \sum_{ij}\sum_k \gamma_{ijk} f\left(\frac{\widetilde{W}_{ik}\left(\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{kj}}{\gamma_{ijk}}\right) = \sum_{ij}\sum_k \gamma_{ijk} f\left(\frac{W_{ik}}{\gamma_{ijk}}\frac{\widetilde{W}_{ik}}{W_{ik}}\left(\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{kj}\right) \tag{28}$$

$$= -\sum_{ij}\sum_k \gamma_{ijk}\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}}\left[\frac{W_{ik}}{\gamma_{ijk}}\right]^{1-\alpha} f\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right) f\left(\left(\widetilde{\mathbf{W}}^T\mathbf{X}\right)_{kj}\right) \tag{29}$$

$$= -\sum_{ij}\sum_k \gamma_{ijk}\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}}\left[\frac{W_{ik}}{\gamma_{ijk}}\right]^{1-\alpha} f\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right) f\left(\beta_{ajk}\frac{\sum_a \widetilde{W}_{ak}X_{aj}}{\beta_{ajk}}\right) \tag{30}$$

$$\leq -\sum_{ij}\sum_k \gamma_{ijk}\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}}\left[\frac{W_{ik}}{\gamma_{ijk}}\right]^{1-\alpha} f\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right)\sum_a \beta_{ajk} f\left(\frac{\widetilde{W}_{ak}X_{aj}}{\beta_{ajk}}\right) \tag{31}$$

$$= -\sum_{ijak} \gamma_{ijk}\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}}\left[\frac{W_{ik}}{\gamma_{ijk}}\right]^{1-\alpha} f\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right)\beta_{ajk} f\left(\frac{W_{ak}}{\beta_{ajk}}\frac{\widetilde{W}_{ak}}{W_{ak}}X_{aj}\right) \tag{32}$$

$$= \sum_{ijak} \gamma_{ijk}\beta_{ajk}\left[\frac{\alpha(1-\alpha)}{X_{ij}^{\alpha}}\right]^2\left[\frac{W_{ik}W_{ak}}{\gamma_{ijk}\beta_{ajk}}\right]^{1-\alpha} f(X_{aj}) f\left(\frac{\widetilde{W}_{ik}}{W_{ik}}\right) f\left(\frac{\widetilde{W}_{ak}}{W_{ak}}\right) \tag{33}$$

$$= \sum_{aik}\left[\widetilde{W}_{ik}^{1-\alpha}W_{ik}^{\alpha}\right]\left[\widetilde{W}_{ak}^{1-\alpha}W_{ak}^{\alpha}\right]\left[-\frac{1}{\alpha(1-\alpha)}\sum_j Z_{ij}^{\alpha}X_{aj}\right] \tag{34}$$

$$= \sum_{aik} V_{ik}V_{ak}S_{ai} = \mathrm{Tr}\left(\widetilde{\mathbf{V}}^T\mathbf{S}\widetilde{\mathbf{V}}\right) = \frac{1}{2}\mathrm{Tr}\left[\widetilde{\mathbf{V}}^T\left(\mathbf{S}+\mathbf{S}^T\right)\widetilde{\mathbf{V}}\right] \tag{35}$$

$$\leq -\frac{1}{\alpha(1-\alpha)}\sum_{ik}\frac{\widetilde{V}_{ik}^2}{2V_{ik}}\left(\widetilde{\mathbf{A}}\mathbf{V}\right)_{ik} \equiv G_1(\widetilde{\mathbf{V}},\mathbf{V}). \tag{36}$$

Figure 1. Upper-bounding $\mathcal{J}_1(\widetilde{\mathbf{W}}) \equiv -\sum_{ij}\frac{X_{ij}^{\alpha}\widehat{X}_{ij}^{1-\alpha}}{\alpha(1-\alpha)}$.

are unresolved in the previous PNMF literature [19, 22, 23]. This is now given by our proof as a special case ($\alpha \to 1$):

$$W_{ik}^{\text{new}} = W_{ik} \sqrt{\frac{(\mathbf{Z}\mathbf{X}^T\mathbf{W} + \mathbf{X}\mathbf{Z}^T\mathbf{W})_{ik}}{\sum_j (\mathbf{W}^T\mathbf{X})_{kj} + (\sum_j X_{ij})(\sum_b W_{bk})}}. \tag{50}$$

Remark 2:

We have previouly proposed an algorithm that iterates the following two steps [18]:

$$W'_{ik} = W_{ik} \left[\frac{\left(\widetilde{\mathbf{A}}\mathbf{W}\right)_{ik}}{(\mathbf{B}\mathbf{W})_{ik}} \right]^{\frac{1}{\alpha}}, \tag{51}$$

$$W_{ik}^{\text{new}} = W'_{ik} \left(\frac{\sum_{ij} \widehat{X}_{ij} \widetilde{Z}_{ij}}{\sum_{ij} \widehat{X}_{ij}} \right)^{\frac{1}{2\alpha}} \tag{52}$$

The update rule (51) is obtained by turning α-PNMF into a constrained α-NMF with $\mathbf{H} = \mathbf{W}^T\mathbf{X}$. It guarantees the Lagrangian objective decreases in each iteration. However, the definition of such a function varies across different iterations and also across different starting values because the Lagrangian multipliers solved by the K.K.T. conditions are determined by the current $\mathbf{W}$. The resulting objectives are therefore not comparable, which hinders monitoring its convergence and prevents improvement by multiple runs using different initial guesses. By constrast, the update rule (14) assures the monotonic decrease of the original α-PNMF objective whose definition does not depend on the iterations and starting $\mathbf{W}$ values. Therefore, one may easily monitor the convergence, rerun the algorithm several times and select the solution with the best objective.

The new multiplicative algorithm also overcomes another shortcoming of the previous one. The update rule (51) is sensitive to the overall scaling of $\mathbf{W}$ and results in zigzag learning paths. Therefore it must be accompanied with a stabilization step (52) with re-calculated $\widehat{\mathbf{X}}$ and $\widetilde{\mathbf{Z}}$. However, the proof of the consistence of this additional update rule with the original objective $D_\alpha(\mathbf{W}\mathbf{W}^T\mathbf{X})$ is still lacking. In contrast, the new algorithm using (14) does not require any additional normalization or stabilization steps, which facilitates its theoretical analysis.

4 Experiments

Suppose the nonnegative matrix $\mathbf{X} \in \mathbb{R}_+^{m\times N}$ is composed of N data samples $\mathbf{x}_j \in \mathbb{R}_+^m$, $j = 1,\ldots,N$. Basically, α-PNMF can be applied on this matrix in two different ways. Firstly, one employs the approximation scheme $\mathbf{X} \approx \mathbf{W}\mathbf{W}^T\mathbf{X}$ and performs *feature extraction* by projecting each sample into a nonnegative subspace. The second approach approximates the transposed matrix $\mathbf{X}^T$ by $\mathbf{W}\mathbf{W}^T\mathbf{X}^T$ where $\mathbf{W} \in \mathbb{R}_+^{N\times r}$, where α-PNMF can be used for *clustering*, with the elements of $\mathbf{W}$ now indicating the membership of each sample in the r clusters. We conduct benchmark experiments on both cases.

4.1 Feature Extraction

We have used the FERET database of facial images [15] as the training data set. After the face segmentation, 2,409 frontal images (poses "fa" and "fb") of 867 subjects were stored in the database for the experiments. All face boxes were normalized to the size of 32×32 and then reshaped to a 1024-dimensional vector by column-wise concatenation. Thus we obtained a 1024×2409 nonnegative data matrix, whose elements are re-scaled into the region [0,1] by dividing with their maximum. For good visualization, we empirically set $r = 25$ in the feature extraction experiments.

After training, the basis vectors are stored in the columns of $\mathbf{W}$ in α-NMF and α-PNMF. The basis vectors have same dimensionality with the image samples and thus can be visualized as *basis images*. In order to encode the features of different facial parts, it is expected to find some localized and non-overlapping patterns in the basis images. The resulting basis images using $\alpha = 0.5$ (Hellinger divergence), $\alpha = 1$ (I-divergence) and $\alpha = 2$ (χ^2-divergence) are shown in Figure 2. Both methods can identify some facial parts such as eyebrows and lips. In comparison, α-PNMF is able to generate much sparser basis images with more part-based visual patterns.

Notice that two non-negative vectors are orthogonal if and only if they do not have the same non-zero dimensions. Therefore we can quantify the sparsity of the basis vectors by measuring their orthogonalities with the τ measurement [20]:

$$\tau = 1 - \frac{\|\mathbf{R} - \mathbf{I}\|_F}{(r(r-1))}, \tag{53}$$

where $\|\cdot\|_F$ is the Frobenius matrix norm and the element R_{st} of matrix **R**gives the normalized inner product between two basis vectors $\mathbf{w}_s$ and $\mathbf{w}_t$:

$$R_{st} = \frac{\mathbf{w}_s^T \mathbf{w}_t}{\|\mathbf{w}_s\|\|\mathbf{w}_t\|}. \tag{54}$$

Larger τ's indicate higher orthogonality and τ reaches 1 when the columns of **W** are completely orthogonal. The numerical values for the orthogonalities τ using the two compared methods are given under the respective basis image plots in Figure 2. All τ values in the right are considerably larger than their left counterparts, which confirms that α-PNMF is able to extract a sparser transformation matrix **W**.

4.2 Clustering

We have used a variety of datasets, most of which are frequently used in machine learning and information retrieval research. Table 1 summarizes the characteristics of the datasets. The descriptions of these datasets are as follows:

- *Iris*, *Ecoli5*, *WDBC*, and *Pima*, which are taken from the UCI data repository with respective datasets Iris, Ecoli, Breast Cancer Wisconsin (Prognostic), and Pima Indians Diabetes. The *Ecoli5* dataset contains only samples of the five largest classes in the original Ecoli database.
- *AMLALL* gene expression database [2]. This dataset contains acute lymphoblastic leukemia (ALL) that has B and T cell subtypes, and acute myelogenous leukemia (AML) that occurs more commonly in adults than in children. The data matrix consists of 38 bone marrow samples (19 ALL-B, 8 ALL-T and 11 AML) with 5000 genes as their dimensions.
- *ORL* database of facial images [16]. There are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions and facial details. In our experiments, we down-sampled the images to size 46×56 and rescaled the gray-scale values to $[0,1]$.

The number of clusters r is generally set to the number of classes. This work focuses on cases where $r > 2$, as there exist closed form approximations for the two-way clustering solution (see e.g. [17]). We thus set r equal to five times the number of classes for *WDBC* and *Pima*.

Suppose there is ground truth data that labels the samples by one of q classes. We have used the *purity* and *entropy* measures to quantify the performance of the compared clustering algorithms:

$$\text{purity} = \frac{1}{N}\sum_{k=1}^{r} \max_{1\le l\le q} n_k^l, \tag{55}$$

$$\text{entropy} = -\frac{1}{n\log_2 q}\sum_{k=1}^{r}\sum_{l=1}^{q} n_k^l \log_2 \frac{n_k^l}{n_k}, \tag{56}$$

where n_k^l is the number of samples in the cluster k that belong to original class l and $n_k = \sum_l n_k^l$. A larger purity value and a smaller entropy indicate better clustering performance.

The resulting purities and entropies are shown in Table 2, respectively. α-PNMF performs the best for all selected datasets. Recall that when $\alpha = 1$ the proposed method reduces to PNMF and thus returns results identical to the latter. Nevertheless, α-PNMF can outperform PNMF by adjusting the α value. When $\alpha = 0.5$, the new method achieves the highest purity and lowest entropy for the gene expression dataset *AMLALL*. For the other five datasets, one can set $\alpha = 2$ and obtain the best clustering result using α-PNMF. In addition, one can see that Nonnegative Matrix Factorization with α-divergence works poorly in our clustering experiments, much worse than the other methods. This is probably because α-NMF has to estimate many more parameters than those using projective factorization. α-NMF is therefore prone to falling into bad local optima.

5 Conclusions

We have presented a new variant of NMF by introducing the α-divergence into the PNMF algorithm. Our α-PNMF algorithm theoretically converges to a local minimum of the cost function. The resulting factor matrix is of high sparsity or orthogonality, which is desired for part-based feature extraction and data clustering. Experimental results with various datasets indicate that the proposed algorithm can be considered as a promising replacement for both α-NMF and PNMF.

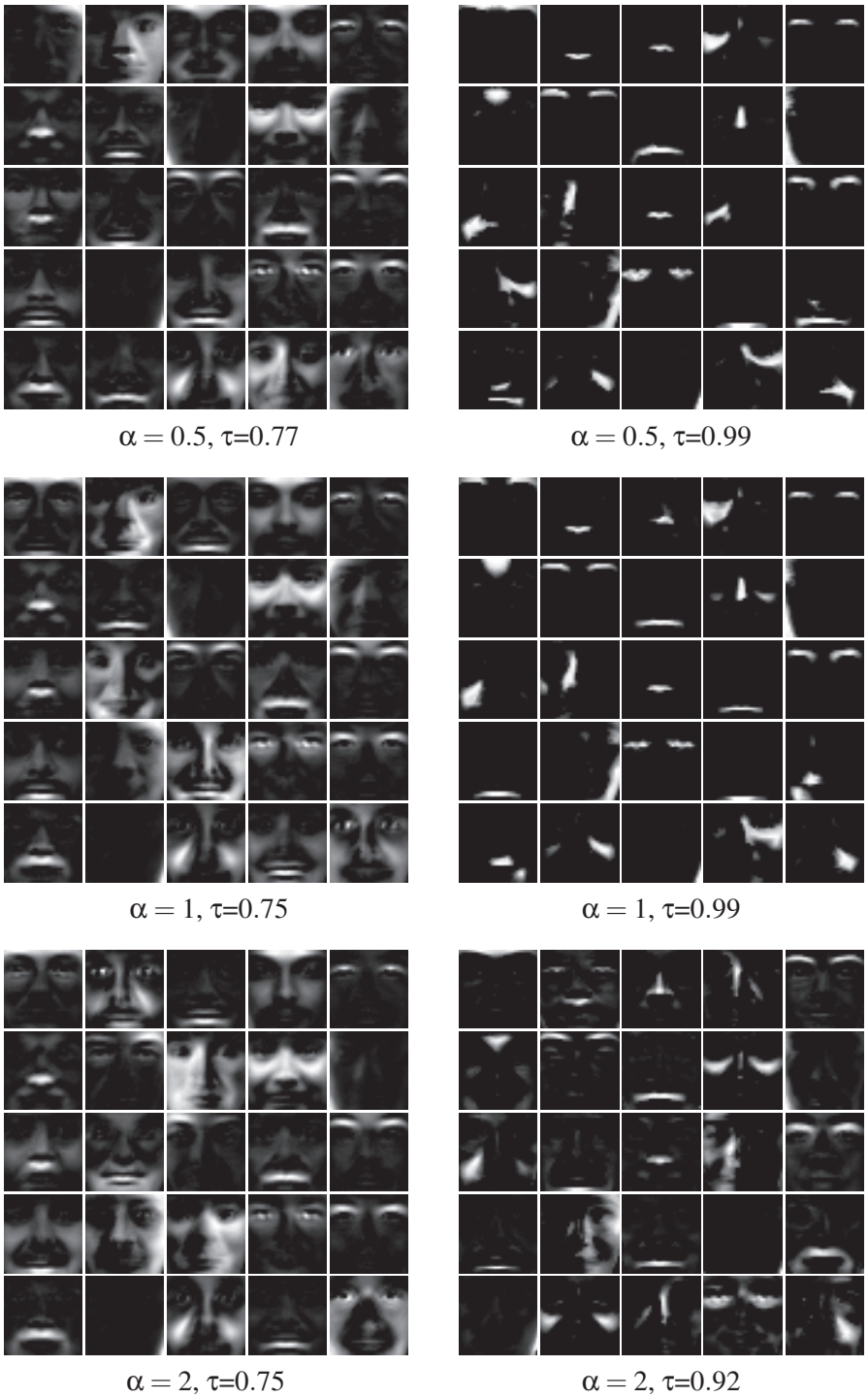


Figure 2. The basis images of (left) α-NMF and (right) α-PNMF.

Table 1. Dataset descriptions

datasets	#samples	#dimensions	#classes	r
Iris	150	4	3	3
Ecoli5	327	7	5	5
WDBC	569	30	2	10
Pima	768	8	2	10
AMLALL	38	5000	3	3
ORL	400	2576	40	40

Table 2. Clustering (a) purities and (b) entropies using α-NMF, PNMF and α-PNMF. The best result for each dataset is highlighted with boldface font.

(a)

	α-NMF			PNMF	α-PNMF		
datasets	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$	-	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$
Iris	0.83	0.85	0.84	0.95	0.95	0.95	**0.97**
Ecoli5	0.62	0.65	0.67	0.72	0.72	0.72	**0.73**
WDBC	0.70	0.70	0.72	0.87	0.86	0.87	**0.88**
Pima	0.65	0.65	0.65	0.65	0.67	0.65	**0.67**
AMLALL	0.95	0.92	0.92	0.95	**0.97**	0.95	0.92
ORL	0.47	0.47	0.47	0.75	0.76	0.75	**0.80**

(b)

	α-NMF			PNMF	α-PNMF		
datasets	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$	-	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$
Iris	0.34	0.33	0.33	0.15	0.15	0.15	**0.12**
Ecoli5	0.46	0.58	0.50	0.40	0.40	0.40	**0.40**
WDBC	0.39	0.38	0.37	0.16	0.17	0.16	**0.14**
Pima	0.92	0.90	0.90	0.91	0.90	0.91	**0.89**
AMLALL	0.16	0.21	0.21	0.16	**0.08**	0.16	0.21
ORL	0.35	0.34	0.35	0.14	0.14	0.14	**0.12**

References

[1] S. Amari. *Differential-geometrical methods in statistics*, volume 28 of *Lecture Notes in Statistics*. Springer-Verlag, New York, 1985.

[2] J.-Ph. Brunet, P. Tamayo, T.R. Golub, and J.P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences*, 101(12):4164–4169, 2004.

[3] Seungjin Choi. Algorithms for orthogonal nonnegative matrix factorization. In *Proceedings of IEEE International Joint Conference on Neural Networks*, pages 1828–1832, 2008.

[4] A. Cichocki, H. Lee, Y.-D. Kim, and S. Choi. Nonnegative matrix factorization with α-divergence. *Pattern Recognition Letters*, 29:1433–1440, 2008.

[5] A. Cichocki, R. Zdunek, A.-H. Phan, and S. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis*. John Wiley, 2009.

[6] I.S. Dhillon and S. Sra. Generalized nonnegative matrix approximations with bregman divergences. In *Advances in Neural Information Processing Systems*, volume 18, pages 283–290, 2006.

[7] Chris Ding, Tao Li, and M.I. Jordan. Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):45–55, 2010.

[8] Chris Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135, 2006.

[9] K. Drakakis, S. Rickard, R. de Fréin, and A. Cichocki. Analysis of financial data using nonnegative matrix factorization. *International Mathematical Forum*, 3:1853–1870, 2008.

[10] C. Févotte, N. Bertin, and J.-L. Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. *Neural Computation*, 21(3):793–830, 2009.

[11] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[12] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13:556–562, 2001.

[13] W. Liu, N. Zheng, and X. Lu. Non-negative matrix factorization for visual coding. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, volume 3, pages 293–296, 2003.

[14] A. Pascual-Montano, J.M. Carazo, Kieko Kochi, Dietrich Lehmann, and R. D. Pascual-Marqui. Nonsmooth nonnegative matrix factorization (nsNMF). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):403–415, 2006.

[15] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face recognition algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22:1090–1104, October 2000.

[16] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of 2nd IEEE Workshop on Applications of Computer Vision*, pages 138–142, Sarasota FL, December 1994.

[17] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[18] Zhirong Yang and Erkki Oja. Projective nonnegative matrix factorization with α-divergence. In *Proceedings of 19th International Conference on Artificial Neural Networks (ICANN)*, pages 20–29, Limassol, Cyprus, 2009. Springer.

[19] Zhirong Yang and Erkki Oja. Linear and nonlinear projective nonnegative matrix factorization. *IEEE Transaction on Neural Networks*, 2010. In press.

[20] Zhirong Yang, Zhijian Yuan, and Jorma Laaksonen. Projective non-negative matrix factorization with applications to facial image processing. *International Journal on Pattern Recognition and Artificial Intelligence*, 21(8):1353–1362, December 2007.

[21] S.S. Young, P. Fogel, and D. Hawkins. Clustering scotch whiskies using non-negative matrix factorization. *Joint Newsletter for the Section on Physical and Engineering Sciences and the Quality and Productivity Section of the American Statistical Association*, 14(1):11–13, 2006.

[22] Zhijian Yuan and Erkki Oja. Projective nonnegative matrix factorization for image compression and feature extraction. In *Proc. of 14th Scandinavian Conference on Image Analysis (SCIA 2005)*, pages 333–342, Joensuu, Finland, June 2005.

[23] Zhijian Yuan and Erkki Oja. A family of modified projective nonnegative matrix factorization algorithms. In *Proceedings of the 9th International Symposium on Signal Processing and Its Applications (ISSPA)*, pages 1–4, 2007.

JAISCR, 2011, Vol.1, No.1, pp. 17–25

SYNCHRONY STATE GENERATION: AN APPROACH USING STOCHASTIC SYNAPSES

Karim El-Laithy and Martin Bogdan
Faculty of Mathematics and Computer Science, University of Leipzig
Johannisgasse 26, 04103 Leipzig, Germany

Abstract

In this study, the generation of temporal synchrony within an artificial neural network is examined considering a stochastic synaptic model. A network is introduced and driven by Poisson distributed trains of spikes along with white-Gaussian noise that is added to the internal synaptic activity representing the background activity (neuronal noise). A Hebbian-based learning rule for the update of synaptic parameters is introduced. Only arbitrarily selected synapses are allowed to learn, i.e. update parameter values. Results show that a network using such a framework is able to achieve different states of synchrony via learning. Thus, the plausibility of using stochastic-based models in modeling the neural process is supported. It is also consistent with arguments claiming that synchrony is a part of the memory-recall process and copes with the accepted framework in biological neural systems.

1 Introduction

Temporal coherence in the firing activity of groups of neurons is widely observed as a common feature throughout the neocortex [40]. The analysis of the responses of stimulated neurons of cat's visual cortex [18] confirmed that activated neurons can reliably produce synchronous discharge with the precision of a few milliseconds. Investigating the key factors in exhibiting such synchronous activity [13] related these observations to both the pure excitatory and the intrinsic time course of synaptic interactions. This coherence is believed to play an important role in neural coding and computational principles [34]. Synaptic background activity (namely the noise) was reported through the theoretical and experimental studies of synchronous activities as a key feature and it was emphasized that such background activity can affect driving coherent spiking activity [10, 23, 19, 7].

Although the real mechanism underlying neuronal synchrony (or temporal correlation) is not completely investigated [40], the issue gained more importance in the research since it has been viewed as a plausible solution to the "binding problem" [35, 12, 32] discussed in [41]. These studies argued that such neuronal temporal synchrony could allow the information about stimuli to be conveyed as temporal relations between neural sites and provide the basis for integrating different features representing the same object. Thus, binding can be defined as the ability of the biological neural system, in terms of inherited flexibilities, to construct higher level symbols by combining more elementary symbols [37].

States of synchrony are involved when any group of neurons realize a degree of synchronous activity, consequently this group of neurons exhibits a state of mental activity [41]. By entertaining such conceptual assumption and in order to observe this state, this temporal synchrony (or temporal correla-

[1] In the referred study, the mathematically known cross-correlation was not meant explicitly, instead the general sense of temporal correlation was meant

tion)[1] is defined over a time period T_{sync}. With this period of time (or Psychological Moment) a brain, mental or neurological state is defined. At times greater than T_{sync} one sees only a sequence of states (state history). Below this time window, a state cannot be defined. The need to maintain this state of temporal correlation for periods greater than a few milliseconds was supported by the argumentation in [37] in order to confine the behavioral conditions fitting the higher brain functions and difficult tasks that require sustained level of activity. In this sense, states of synchrony are believed to be involved in the processing of sensory inputs [11]. Moreover, it is argued in [38] that attention and awareness emerge from the interacting dynamics in terms of synchrony states among distributed neural ensembles. Von der Malsburg stated that plausible values for this time window to define such state could be in the range of 50 - 200 millisecond and may be also extended to involve minutes if other mental aspects are in concern [41]. Within this time window, the actual signal fluctuations are not relevant (for a complete review please refer to [32, 37, 1]).

The generation of synchrony in artificial neural networks (ANN) is addressed in many theoretical and numerical studies, e.g. [13, 4, 31, 2, 42, 26]. These studies confirmed the ability of an ANN to realize the temporal synchrony on the time scales of a few milliseconds even with sharp synchronization on the time scale of single spikes. In general, these studies simulated a population of integrate-and-fire (IAF) neurons with adequate interconnectivity. Their discussions highlighted the major role of excitatory interconnections to achieve a certain degree of synchronous activity. Tsodyks et al presented a notable study in [40]. They considered the non-linear (frequency dependent) synapses for the generation of synchronous firing. Their results showed that the incorporation of nonlinear synapses in recurrent networks provide the basis for the emergence of short-time synchronous activity.

However, it has been shown that deterministic representation of the neural actions does not *model the biological neural behavior realistically* [24, 28, 29]. In addition, Kröger showed in [20] that probabilistic option in regard to neuroscience offers advantages over the deterministic one. A stochastic pulsed-coupled neural network was presented in [5] showing synchronous and asynchronous behavior. A reduced stochastic dynamic model of an interconnected population of excitatory neurons with activity-dependent synaptic depression was proposed in [15], the discussion was focused on the bistability of voltage activities as up and down states. This is believed to be also related to the states of temporal synchrony within the neural ensemble. However, these studies and other did not consider the potential effects of *stochastic dynamic synapses* on synchronization of neural activity in ANN.

In the general case of modeling a biologically-observed neural aspect, we proposed that the stochastic modeling of neuronal and synaptic activity are better than the deterministic one; Because the stochastic approaches are able to account for the essence of neural variability [8]. This was clearly elucidated in [16] trying to predict the exact spike timing of a thalamic neuron. This statement was further supported by our results in the international Quantitative Single-Neuron Modeling 2009[2] (succeeding one of the 2008-challenge [17]). In this challenge and using our stochastic synaptic model [8], we have defined the new benchmark in predicting the spike timing of a single post-synaptic neuron in the lateral geniculate nucleus knowing the spike train on the pre-synaptic side (i.e. in a retinal ganglion cell) [9, 30].

In [8] we have proposed a basic framework with a modified version of the basic stochastic synaptic model (presented in [24]) coupled to a leaky IAF neuron. Our preliminary results showed that an ANN with the introduced framework was able to realize special regimes of activity with synchronous discharge over biologically tenable periods of time. Here, we report the detailed description and analysis of this framework along with the involved results. Specifically, in this paper, we investigate the ability of a network comprising IAF neurons and stochastic synapses to realize the concept of synchrony (the temporal correlations) between the signals of grouped neurons as states of synchronous activity. The goals are: a) to construct an ANN so that when driven by trains of spikes should be able to transform input signals combined with background synaptic activity (here introduced as synap-

[2] http://incf.org/community/competitions/spike-time-prediction/2009/

tic noise) into correlated outputs and b) to show the ability to sustain such level of synchrony over a considerable time course T_{sync}. For this task, a Hebbian-based Reinforcement-like learning algorithm is introduced as well.

2 The Model

Neuron Model:

Neurons are modeled as leaky-IAF neurons usually used in such type of simulations [40]. Each neuron is described by its voltage membrane potential V, with the following dynamics:

$$\tau_V \frac{dV}{dt} = -V + E_{\text{psp}} + \varepsilon, \tag{1}$$

where τ_V is the membrane time constant set at 20 msec, and E_{psp} is the total observed excitatory post-synaptic potential from all pre-synaptic terminals. ε is the added white-Gaussian noise[3] representing the background synaptic activity, with $\langle \varepsilon \rangle = 0$. When V exceeds a certain threshold V_{th}, a spike is generated and V is reset to a resting value, $V_{\text{rest}} = -70\text{mV}$.

Synaptic Model:

We have first introduced the modified stochastic synaptic model (MSSM) in [8, 9]. According to this model, each synaptic connection is modeled as a stochastic activity-dependent connections. This model estimates the transmission probability of an arriving action potential, i.e. spike, from a presynaptic neuron via a synapse to a postsynaptic neuron. The probability-of-release involved is governed by two counteracting mechanisms: facilitation and depression. Facilitation reflects the Ca^{2+} concentration in the presynaptic neuron, while depression represents the effect of the concentration of ready-to-release vesicles in the pre-synaptic neuron. The probability that the ith spike in the spike train triggers the release of a vesicle at time t_i at a given synapse is given by:

$$P(t_i) = 1 - e^{(-C(t_i)\cdot V(t_i))}, \tag{2}$$

where $C(t_i)$ and $V(t_i)$ represent the facilitation and depression mechanisms respectively at t_i. $C(t)$ and $V(t)$ are expressed as [29]:

$$C(t) = C_o + \sum_{t_i} \alpha e^{-(t-t_i/\tau_C)} \tag{3}$$

$$V(t) = max(0,\ V_o - \sum_{t_i} e^{-(t-t_i/\tau_C)}) \tag{4}$$

In eq. 3, τ_C and α represent the decay constant and the magnitude of the response respectively. C_o represents the initial concentration of Ca^{2+} in the pre-synaptic terminal. In eq. 4, $V(t)$ is the expected number of vesicles of neurotransmitter molecules (Nt) in the ready-for-release pool at time t. V_o is the max. number of vesicles that can be stored in the pool. τ_V is the time constant for refilling the vesicles. For the simulation, a discrete version of equations 3 and 4 adopted from [28] is used. Thus, these equations read:

$$\begin{aligned} C(n) &= \alpha \cdot \theta(n-1) \\ &\quad + k_C \cdot (C(n-1) - C_o) + C_o, \end{aligned} \tag{5}$$

$$\begin{aligned} V(n) &= -P(n-1) \cdot \theta(n-1) \\ &\quad + k_V \cdot (V(n-1) - V_o) + V_o. \end{aligned} \tag{6}$$

In eq. 5, k_C corresponds to the decay time constant, τ_C, of the response to a single incoming spike. k_V corresponds to the time constant, τ_V, for refilling the vesicles. $\theta(n)$ represents the instantaneous input firing rate observed at the synapse at time instant n; it equals then Δ_{isi}^{-1}, where Δ_{isi} is the last observed inter-spike-interval (ISI).

Recalling that the binding process of Nt on the postsynaptic membrane induce E_{psp}. Thus, E_{psp} is related to this process through the following dynamics [27]:

$$\tau_{epsp} \frac{dE_{\text{psp}}}{dt} = -E_{\text{psp}} + N_t, \tag{7}$$

where τ_{epsp} is a decay time-constant. N_t is the concentration of the Nt in the synaptic cleft. We assume that the latter can be estimated by tracing the amount of vesicles of Nt that remains in the presynaptic neuron, $V(t)$, over time. We introduce the following equation to estimate $N_t(n)$ and consequently couple the SSM with the IAF neuron model:

$$\begin{aligned} N_t(n) &= max(0,\ V(n) - V(n-1)) \\ &\quad + N_t(n-1) e^{-\Delta_{\text{isi}}/\tau_{N_t}} \end{aligned} \tag{8}$$

[3] The MATLAB function (wgn) is used with overall 0 db power over 1 ohm load impedance

In eq. 8, N_t at any time instant n is the summation of: a) the estimated amount of Nt added with each release at any time step n (or the decrease in $V(n)$ over the time step); where the $max(\ldots)$ avoids negatives and b) the amount of Nt that remains in the cleft from previous releases. The decay with τ_{N_t} reflects the biological cleaning action, or the removal of the Nt from the cleft. Equations 1 and 7 are implemented as discrete forms introduced by [27] using Impulse Invariant Transform in order to facilitate the computations (Please refer to the articles [21, 27] for the derivation).

3 Network and Simulation

Up to our knowledge, the biologically accepted network size, in which temporal correlation can be observed and effectively utilized, is not precisely specified [40]. However, there are some hypothetical suggestions discussing the tenability of the network size. Singer, for example, analyzed the major factors affecting the ability of a group of neurons to exhibit synchronous activity [37]. He pointed out that the network size could be as small as two mutually coupled neurons and may be up to 100 neurons. Herzog and Gerstner argued that if synchrony is an essential feature for the brain activities, it should also be feasible in small networks [14]. They called this "the small network argument", or the new benchmark for consciousness. Thus, they reported, there is a minimal model or a small network that satisfies the criteria underlying consciousness, e.g. temporal synchrony, but is not conscious itself. They stated that groups of up to seven neurons are sufficient to realize memory, learning, or synchrony. Based on the analysis done in [6], a network of two neurons should be able to achieve spike-to-spike synchrony when enough mutual conductance is available.

Thus, two network structures are used in this study. A schematic of the first introduced network is in Fig. 1(a) with the input being fed only to the first neuron, N_1. The network consists of 3 mutually interconnected neurons with only excitatory synapses. Also a bigger network is used with 8 neurons as in Fig. 1(b). In the latter one, the input is also fed to N_1, and similar to the smaller network, the feed back to the input neuron is only possible from the neighboring neurons; in this case from N_2, N_3 and N_4. All synaptic connections are MSSM synapses as described in section 2 supported with white Gaussian noise generators.

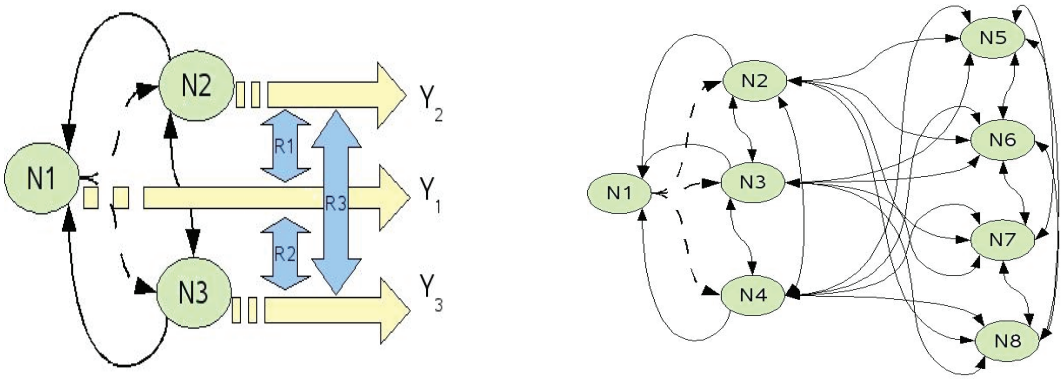
The input is a set of 200 trains of spikes, each with a Poisson distributed inter-spike intervals for an epoch of 150 and 100 msec at 1 msec discretization for the 3 neuron and the 8 neuron networks respectively. This time epoch is arbitrarily used as a median value for the proposed time scale of T_{sync} over which synchrony is plausible. The spike generator is adjusted to generate spikes with a maximum overall firing-rate of 300 Hz. Meanwhile, at each synapse a white Gaussian noise is added locally to the induced postsynaptic potential from this synapse. The level of the noise is modulated via simulated linear amplification.

For representing synchrony, the cross-correlation based measures are accepted in the detection of similarities in responses and for synchrony [40, 26]. Correlograms are not considered in this study based on reviews to the analytical reliability of its results [3]. A correlation-based measure is introduced in [33] that calculates the cross-correlation coefficient between neural responses after applying a Gaussian filtration on the responses. Here, the max. of cross-correlation coefficients between the filtered signals is used to indicate the degree of synchrony. The width of the Gaussian filter is chosen to be equal to the chosen neuronal refractory period of 2 msec.

4 Learning Rule

Generally, if no learning is implemented, the input signal and noise are fed to the network. At the end of each epoch, the mean R_m of the max. cross-correlation coefficients is calculated from all possible combinations between the responses from the three neurons. For example in the case of the network with 3 neuron $R_m = mean(R_1, R_2, R_3)$, where R_1 is the max. cross correlation coefficient between the Gaussian filtered versions of Y_1 and Y_2. Similarly R_2 and R_3, as in Fig. 1(a). An analogous approach is used with the bigger network.

A Hebbian-based learning rule is introduced in [22, 27] showing how both the timing parameters and constants can be updated based on the spiking activity of pre- and postsynaptic neurons. Here,



(a) (b)

Figure 1. Network schematic. a) 3 neuron network. The dashed lines are those synapses permitted to be trained. Double arrowed connections represent a mutual connection. Horizontal-right oriented arrows: output signals from each neuron: Y_1, Y_2 and Y_3. Vertical-two-headed arrows: the corresponding cross-correlation coeff.: R_1, R_2 and R_3. The trained synapses are from N_1 to N_2 and N_3. b) 8 neuron network. The dashed lines are those synapses permitted to be trained. Double arrowed connections represent a mutual connection. The details of the outputs and the calculation of cross correlation are omitted for clarity. The trained synapses are the synapses from N_1 to N_2, N_3 and N_4

this rule is extended to MSSM parameters. Specifically, the dynamics of synaptic or neural activities are governed through the contribution of electrochemical mechanisms. Each of them is represented via a value, m, i.e. α in eq. 3 represents the max. allowed incurrent of Ca^{+2} ions to the presynaptic terminal [24]. A mechanism m could be either excitatory or inhibitory. According to the pre- and postsynaptic activity, the value of m is either increased or decreased following the Hebbian approach [22]. The update of the contribution values could be basically mathematically formed as:

$$m_{\text{new}} = (1 \pm r) m_{\text{current}}, \tag{9}$$

where r is the learning rate. In the proposed MSSM, such parameters are for example $\tau_{N_t}, \tau_V, \tau_C, \alpha$, C_o and V_o.

We introduce a feedback parameter, K, that represents the advance in the direction of getting both more and stable synchrony between the responses (i.e. a higher cross-correlation coefficient). Thus, it is the difference in the observed synchrony R_m from the current run and the previous one, let

$$K = R_{m_{\text{current}}} - R_{m_{\text{previous}}} \tag{10}$$

K is used as a modulator to the learning rate. Thus, the learning rule can be rewritten:

$$m_{i_{\text{new}}} = (1 \pm r \cdot K) \cdot m_{i_{\text{current}}} \tag{11}$$

K can reverse the direction of the updating process of the parameters since it is a signed value, and can either accelerate or decelerate the learning process. This learning rule has the implicit objective of correlating the outputs corresponding to the same input *properties*. It emphasizes the sensitivity of the network to temporal and statistical properties embedded in input signals [27]. In this study, only forward MSSM synapses are allowed to update their parameters via this rule as illustrated in Fig. 1(a) and 1(b).

5 Results

Figures 2(a) and 2(b) illustrate the performance of the networks during the simulation with and without learning. In each subfigure, the two different traces can be taken as an indicator for two *neuronal states of synchrony*: Ground-State (Learning Off) and New-State (Learning On). The network needs about 20 - 40 learning runs until it reaches the New-State. The introduced networks with MSSM are able to show two states of synchrony over a time window of 150 and 100 msec in 3 neuron and 8 neuron network respectively. In this case the noise intensity is held constant and this noise intensity is referred as the zero-level white Gaussian noise, $wgn^{(0)}$.

There may be no clear analytical evidence that

biological neural systems can turn learning on and off. However, the logical analysis of cognitive control tasks in [7] still accepts that the biological neural circuitry can perform something similar to the control of task sharing and non-physical rewiring of neural ensembles, e.g. the control over the flow of information. This makes the idea of using the on/off learning rule acceptable. In the following, however, the use of switching learning On/Off is avoided by using the intensity of the noise as a controlling factor. This is a direct consequence of the experimental and theoretical analysis mentioned before in the introduction.

In order to view the realization of the synchrony states in parallel with the involved synaptic dynamics, using only the 3 neuron network, a higher intensity of noise is fed to the network, $wgn^{(1)}$, where $wgn^{(1)} > 2wgn^{(0)}$. Furthermore, one more randomly chosen synapse is allowed to be tunned. This is meant to allow for non planned transition between the internal available states of the network, if exists, to take place. In other words, the objective of this setup is to observe how the network behaves in general and to test if it is able to accomplish any other available stable state of synchrony than the first reachable one. In Fig. 3(a), the observed synchrony level is shown. The network is able to sustain both of the states defined before keeping learning on along the entire simulation. The vertical dashed lines show when the network decided to change its state from the upper synchrony level to the lower one. Within the dashed lines, there is a finite overshoot in the synaptic strength value. We interpret this disturbance to be the search for the new stable state as the network is pushed out of the first stable state.

However, a network that achieves only synchrony is not so useful unless it can desynchronize its activity [4]. Hence, the ability of the network to desynchronize itself is also investigated. By setting the input to zero, the only remaining input is the noise which is equal for all synapses. Thus, the neurons start with typical firing patterns, this can be (obviously) seen in Fig. 3(b), since the two traces of the cross-correlation coefficient start almost with a value of 1. Since learning is on, and while the network is trying to reach a general stable level of synchrony, the networks desynchronized itself reaching a middle level near those upper ones achieved as in Fig. 3(b).

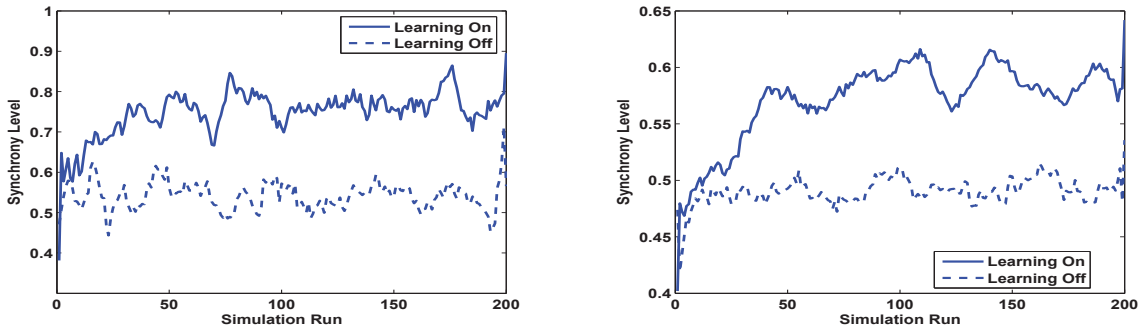
6 Discussion and Conclusion

The simulations presented here demonstrate that networks of neurons interconnected with stochastic synapses have a real tendency to realize special regimes of activity with synchronous discharge over biologically tenable periods of time. The simulation here is restricted to excitatory connections based on the mentioned discussion in the introduction section, however, the role of synaptic depression or specific inhibitory connections in case of using stochastic synapses needs further investigation.

Considering the network size in this study, it cannot represent a cortical minicolumn [40] consisting of neurons, however, it could be viewed, in the context of synchronous activity, to have similar receptive field properties [1]. Other possibilities are issues of further study, e.g. it is expected that in larger simulated networks, such as a cortical mini- and hyper-column, where interconnections between neurons reflect their receptive field properties, other profiles of activity, may be with multiple internal states or with shorter time course, could exist.

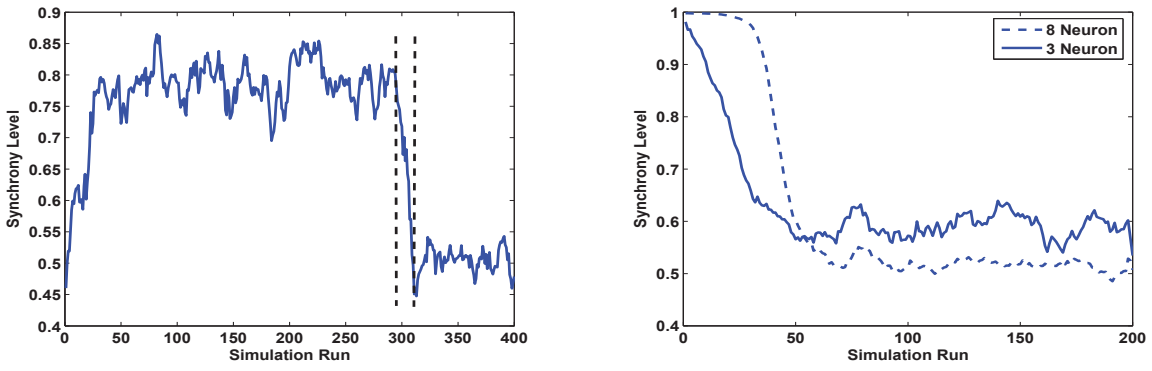
As for the learning algorithm, and up to our knowledge, there may be no clear analytical evidence that biological neural systems can turn learning on and off. However, the logical analysis as in [7] still accepts that the biological neural circuitry can perform something similar to control, e.g. the control over the flow of information, the task sharing and non-physical rewiring of neural ensembles. This makes the idea of using the on/off learning rule acceptable.

In this paper, an ANN of IAF neurons coupled via MSSM synapses is introduced. In case of the evoked simulations, the network is driven by Poisson distributed trains of spikes and white-Gaussian noise. The latter is fed to synaptic activities. Considering that the Poisson distributed input represents a neural activity that carries certain information, the change in the level of synchrony could be seen as if the network, is likely to be, *memorizing* or *internally recalling* this input by pushing all its activities to *sync* with it. On the other hand, and in the case of spontaneous activity as the in-



(a) (b)

Figure 2. Simulation result. a) The detected level of internal synchrony of the 3 neuron network in two cases: when learning is allowed, and when not allowed. T_{sync} = 150 msec. b) The detected level of internal synchrony of the 8 neuron network in two cases: when learning is allowed, and when not allowed. T_{sync} = 100 msec.



(a) (b)

Figure 3. a) The observed behavior of the network with 3 neuron with MSSM in case of allowing more than the feedforward synapses to be tuned. The level of fed noise is higher than the standard one. b) Desynchronization. The networks reach a stable level of synchrony lower than the starting level. Results from both networks, 3 and 8 neuron, are illustrated.

put is set to zero, the networks are able to find a lower stable state of synchrony, since they desynchronize their firing pattern. Both synchrony and desynchrony cope with the conceptual postulations discussed in the context of synchrony [41] and the role of noise in neural information processing [10]. Hence, the proposed framework achieved successfully the general sense of sustaining a defined state of synchronous activity within a group of neurons over a considerable time course of 100 - 150 milliseconds.

It is worth mentioning that the dynamics of facilitation and depression in this MSSM model are conceptually similar to those presented in the famous synaptic model by Markram et al [25], since the facilitation and depression dynamics in our model replace the efficacy and utilization parameters respectively (Please review [39, 25] for more details). The main difference between the two models, however, are mainly seen in two perspectives: a) MSSM realizes the stochastic nature of the release process of Nt, and b) It accounts explicitly for the activity of two synaptic resources which are Ca^{2+} and Nt. However, in Markram's model, the input synaptic current is deterministically evaluated by tracing both the utilization and recovery of a single synaptic resource [25].

It remains, however, that the expressive power of the proposed dynamics in terms of the number of achievable states is to be tackled. On simulation basis, there are some obstacles standing against defining more stable states for the whole network. Basically, although there are some analytical methods, the exact definition of a stable state of synchronous activity in the network is still questionable [36]. The proposed means in this paper as a cross correlation coefficient may lack the fine resolution that is probably needed to differentiate between different states or even sub-states of neural activity. Secondly, advances in the way to define a stable state of activity should be directly combined with dramatic changes in the learning rules.

The stochastic based synaptic model MSSM demonstrated high sensitivity to the change in the noise level, or background activity level. This deeply agrees with the experimental results and the logical considerations reported in [10] that the background activity *controls* the input output characteristics. This envisages the stochastic-based synaptic models as more plausible for modeling biological neural activities. Moreover, it reflects the importance of considering *both* the concentration of Nt and Ca^{+2} as key players in the synaptic dynamics. This allows MSSM to meet a rich repertoire of realistic dynamical behaviors and features.

References

[1] B. Averbeck, P. Latham, and A. Pouget. Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, 7:358–366, 2006.

[2] P. Bressloff and P. Coombes. Dynamics of strongly coupled spiking neurons. *Neural Comput.*, 12(1):91–129, 2000.

[3] C.. Brody. Correlations without synchrony. *Neural Comput.*, 11(7):1537–1551, 1999.

[4] S. Campbell, D. Wang, and C. Jayaprakash. Synchrony and desynchrony in integrate-and-fire oscillators. *Neural Computation*, 11(7):1595–1619, 1999.

[5] R. DeVille and C. Peskin. Synchrony and Asynchrony in a Fully Stochastic Neural Network. *Bulletin of Mathematical Biology*, 70(6):1608–1633, 2008.

[6] R. Dodla and C. Wilson. Synchrony-asynchrony transitions in neuronal networks. *BMC Neuroscience*, 9(Suppl 1):P9, 2008.

[7] D. Durstewitz. Self-Organizing Neural Integrator Predicts Interval Times through Climbing Activity. *J. Neuroscience*, 23(12):5342–5353, 2003.

[8] K. El-Laithy and M. Bogdan. Synchrony state generation in artificial neural networks with stochastic synapses. In *Artificial Neural Networks–ICANN 2009*, volume 5768 of *LNCS*, pages 181–190. Springer, 2009.

[9] K. El-Laithy and M. Bogdan. Predicting spike-timing of a thalamic neuron using a stochastic synaptic model. In *ESANN Proceedings*, pages 357–362, 2010.

[10] J. Fellous, A. Rudolph, B. Destexhe, and T.Sejnowski. Synaptic background noise controls the input/output characteristics of single cells in an in vitro model of in vivo activity. *Neuroscience*, 122:811–829, 2003.

[11] C. Gilbert and M. Sigman. Brain states: Top-down influences in sensory processing. *Neuron*, 54(5):677–696, 2007.

[12] C. Gray. The temporal correlation hypothesis of visual feature integration: Still alive and well. *Neuron*, 24(1):31–47, September 1999.

[13] D. Hansel, G. Mato, and C. Meunier. Synchrony in excitatory neural networks. *Neural Comput.*, 7(2):307–337, 1995.

[14] M. Herzog, M. Esfeld, and W. Gerstner. Consciousness & the small network argument. *Neural Networks*, 20(9):1054–1056, 2007.

[15] D. Holcman and M. Tsodyks. The Emergence of Up and Down States in Cortical Networks. *Science*, 2(3):174–181, 2006.

[16] R. Jolivet, A. Rauch, H. Loscher, and W. Gerstner. Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *J. Computational Neuroscience*, 21:35–49, 2006.

[17] R. Jolivet, F. Schürmann, T. Berger, R. Naud, W. Gerstner, and A. Roth. The quantitative single-neuron modeling competition. *Biological Cybernetics*, 99(4):417–426, November 2008.

[18] P. König, A. Engel, P. Roelfsema, and W. Singer. How precise is neuronal synchronization? *Neural Comput.*, 7(3):469–485, 1995.

[19] T. Kreuz, J. Haas, A. Morelli, H. Abarbanel, and A. Politi. Measuring spike train synchrony and reliability. *BMC Neuroscience*, 8(Suppl 2):P79, 2007.

[20] H. Kröger. Why are probabilistic laws governing quantum mechanics and neurobiology? *Solutions and Fractals*, 25:815, 2005.

[21] J. Liaw and T. Berger. Dynamic synapse: A new concept of neural representation and computation. *Hippocampus*, 6:591–600, 1996.

[22] J. Liaw and T. Berger. Computing with dynamic synapses: A case study of speech recognition. In *Proc. IEEE Int. Conf. Neural Networks*, pages 352–355, 1997.

[23] B. Lindner, L. Schimansky-Geier, and A. Longtin. Maximizing spike train coherence or incoherence in the leaky integrate-and-fire model. *Phys. Rev. E Stat. Nonlinear Soft Matter Phys.*, 66(3 Pt 1):031916, 2002.

[24] W. Maass and A. Zador. Dynamic stochastic synapses as computational units. *Neural Computation*, 11:903–917, 1999.

[25] H. Markram, Y. Wang, and M. Tsodyks. Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. of the Nat. Academy of Sciences of the USA*, 95(9):5323–5328, 1998.

[26] S. Mikula and E. Niebur. Rate and synchrony in feedforward networks of coincidence detectors: Analytical solution. *Neural Comput.*, 17(4):881–902, 2005.

[27] H. Namarvar, J. Liaw, and T. Berger. A new dynamic synapse neural network for speech recognition. In *Proc. IEEE Int. Conf. Neural Networks*, volume 4, pages 2985–2990, 2001.

[28] T. Natschläger. *Efficient Computation in Networks of Spiking Neurons Simulations and Theory*. PhD thesis, Institute of Theoretical Computer Science, Austria, 1999.

[29] T. Natschläger, W. Maass, and A. Zador. Efficient temporal processing with biologically realistic dynamic synapses. *Computation in neural system*, 12:75–78, 2001.

[30] R. Naud, T. Berger, L. Badel, A. Roth, and W. Gerstner. Quantitative single-neuron modeling: competition 2008. In *Neuroinformatics 2008*, 2008.

[31] L. Neltner and D. Hansel. On synchrony of weakly coupled neurons at low firing rate. *Neural Comput.*, 13(4):765–774, 2001.

[32] E. Salinas and T. Sejnowski. Correlated neuronal activity and the flow of neural information. *Nature Reviews Neuroscience*, 2:539–550, 2001.

[33] S. Schreiber, J. M. Fellous, D. Whitmer, P. Tiesinga, and T. J. Sejnowski. A new correlation-based measure of spike timing reliability. *Neurocomputing*, 52-54:925–931, 2003.

[34] T. Sejnowski and O. Paulsen. Network Oscillations: Emerging Computational Principles. *J. Neuroscience*, 26(6):1673–1676, 2006.

[35] W. Singer. Neuronal synchrony: a versatile code for the definition of relations. *Neuron*, 24:49–65, 1999.

[36] W. Singer. Understanding the brain. *European Molecular Biology Org.*, 8:16 –19, 2007.

[37] W. Singer and C. Gray. Visual feature integration and the temporal correlation hypothesis. *Annu. Rev. Neuroscience*, 18:555–586, 1995.

[38] C. Tallon-Baudry. Attention and awareness in synchrony. *Trends in Cognitive Sciences*, 8(12):523–525, 2004.

[39] M. Tsodyks, K. Pawelzik, and H. Markram. Neural networks with dynamic synapses. *Neural Computation*, 10:821–835, 1998.

[40] M. Tsodyks, A. Uziel, and H. Markram. Synchrony generation in recurrent networks with frequency-dependent synapses. *J. Neuroscience*, 20:50, 2000.

[41] C. von der Malsburg. The what and why of binding: The modeler's perspective, September 1999.

[42] C. Van Vreeswijk and D. Hansel. Patterns of synchrony in neural networks with spike adaptation. *Neural Comput.*, 13(5):959–992, 2001.

JAISCR, 2011, Vol.1, No.1, pp. 27–41

A TWO-LAYER NEURAL SYSTEM FOR REDUCED-REFERENCE VISUAL QUALITY ASSESSMENT

Judith Redi, Paolo Gastaldo, Rodolfo Zunino
Department Biophysical and Electronic Engineering
University of Genoa
Via Opera Pia 11/, 16145 Genoa, Italy

Abstract

Real-time assessment of visual quality can be efficiently supported by reduced-refe-rence paradigms, which require a very limited amount of information on the original signal, easily embeddable in the signal itself. In this paper, a reduced-reference system for image quality assessment is proposed, based on a small sized numerical description of images encoding the luminance distribution and its variations due to visual distortions. The assessment paradigm is implemented exploiting machine learning tools and articulates in two phases: first, a Support Vector Machines-based classifier identifies the kind of distortion affecting the image; then, the actual quality level of the distorted image is computed by a specifically trained SVM regressor. The general validity of the approach is supported by experimental validations based on subjective quality data.

1 Introduction

As the fruition of video and multimedia contents becomes wider, exploiting new media technologies such as the internet, electronic imaging systems are more than ever required to guarantee the high quality of the displayed signal, regardless to the distortions originated during the transmission and/or the displaying processes. Such a scenario calls for the need of accurate, on-board signal post-processing systems, able to detect the artifacts brought about by distortions, to estimate the perceived quality of the received images and to apply enhancement algorithms, in order to appropriately correct and enhance the finally displayed signal. The study and modeling of visual quality perception covers then a crucial role in the development of cutting-edge video technology applications.

Up to now, the most reliable tool for estimating perceived quality is subjective testing, directly involving humans and their judgment [5, 31, 20, 1]. Real-time post-processing chains cannot rely on such expensive and time-consuming techniques, hence requiring automated (objective) quality assessment systems (OQAs) [6, 21, 36]. OQAs rely on the computation of objective metrics, which can either exploit low to high level Human Visual System salient features, or relevant statistical descriptions of the signal. Full-Reference (FR) paradigms [33, 11, 37, 3, 7, 8] perform quality assessments by computing features as a comparison of the original with the distorted image, hence requiring full access to both signals. Such techniques find several applications (i.e. coding algorithms performance tests) but cannot be exploited in real-time contexts. Reduced Reference (RR) methods provide instead a good trade-off between blind (No Reference, [39, 35, 19, 9, 18, 40, 16]) and FR quality assessment, only involving a limited amount of numerical features characterizing the original signal [30, 34, 38, 17, 15, 23, 24]. This overcomes a common drawback of No Reference algorithms, namely their limited applicability to a single type of distortion. Thus, the RR paradigm can provide a successful approach for supporting real-time modeling of perceived quality.

Several existing FR and RR algorithms [37, 3,

34, 38] can be defined as "general-purpose" (as opposed to distortion-specific), and offer the great advantage of using a single feature-based description for assessing the quality losses due to different kinds of distortions (e.g. noise, compression artifacts, reduced sharpness). Usually, this non-linear mapping is computed distortion-dependent, adapting the general-purpose metric to the specific effects of one distortion on quality. Such a strategy is proved to be successful; however, it requires some a priori knowledge on the kind of distortion affecting the signal, which is often considered as given. The research proposed in this paper addresses such central issue by introducing a two-layer OQA system that can automatically identify the kind of distortion affecting the signal, and then apply the most effective objective metric in the quality assessment phase.

The present work further investigates on assessment models based on connectionist paradigms [24]. A two-layer RR system based on Support Vector Machines (SVMs) [32] is designed to map a numerical description of the image into quality scores. To handle different distortions, first a classifier determines the type of distortion affecting the image. According to the output of the classifier, a regression machine is chosen among a bank of predictors trained to evaluate the effects on the quality of different distortions. This machine eventually performs the non linear mapping required to quantify the loss in quality of the image with respect to its original version.

The proposed system is fed with a feature-based representation of the distorted image and its original version. Luminance distribution information supports image representation, as artifacts brought about by digital processing affect the original luminance content of the image, each in a peculiar way. Hence, the rationale of the present approach is that by comparing the statistics of the original and distorted image one can identify both the kind and the extent of the distortion. Previous works [23, 24] showed that second order statistics can apply successfully toward that end; therefore, this research adopts a set of features derived from the co-occurrence matrix [25].

In this paper, bandwidth and computational constraints are also considered as parameters to evaluate the effectiveness of the approach. Hence, with respect to [24], a different strategy, involving changes in the metric computation and in the system setup, is used. Experimental validation is provided on the LIVE database [10], including three types of distortion: White noise, Gaussian Blur and JPEG compression. Empirical results confirm the validity of the connectionist paradigm and the effectiveness of luminance statistics for predicting the image quality. Furthermore, the changes applied to [24] further prove the flexibility of the system and the robustness of the overall approach.

2 System Overview

Reduced reference OQAs represent a promising solution for on-board, real time image quality assessment in consumer multimedia systems. The major advantage that such methods offer is the possibility of assessing quality by exploiting some information about the original image, provided that such information is sufficiently small-sized with respect to the video signal to be transmitted. As a major consequence, that information can be included in the signal as metadata without affecting the bandwidth occupation. In this regard, RR approaches improves over full-reference approaches, which are actually unsuitable for most applications, as they require full access to the original signal.

This study proposes a Reduced Reference OQA, based on a double-layer approach that allows handling the effects of different distortions. The first layer tackles the task of distortion identification. The second layer is made of a set of dedicated predictors, specifically trained to understand the quality losses in the image due to the presence of the detected distortion.

In this research, emphasis is indeed put on limiting the computational cost of the objective metric and the amount of information required to be computed from the original signal and set along the transmission channel. Three main factors allow to tackle this purpose effectively, namely : (1) the suitability of luminance distribution derived features to describe quality losses; (2) a feature selection procedure designed to discard non informative features, which in turn would inflate the size of the metadata vector to be transmitted; (3) the non-linear modeling power of Machine Learning tools (i.e. Support Vector Machines), which precisely

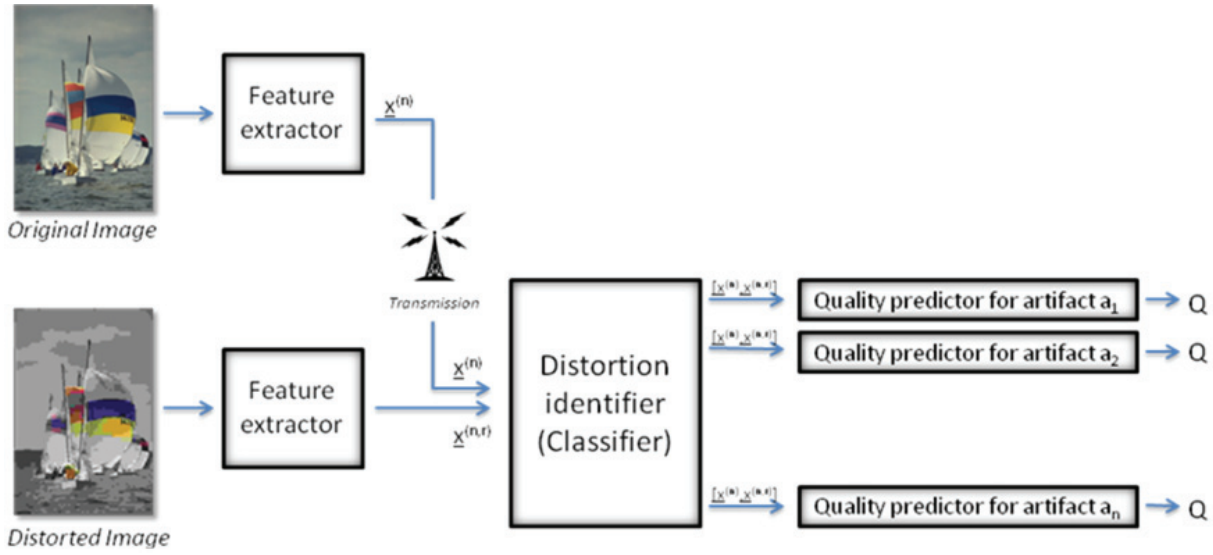


Figure 1. Overview of the Reduced Reference General Purpose Objective Qulaity Assessment System.

mimic human visual perception without requiring a detailed model of the HVS.

2.1 Outline of the Reduced-Reference Quality Prediction Framework

Let $I^{(n)}$ be the reference image, and $\bar{I}^{(n,r)}$ the image resulting from the insertion of some distortion to$I^{(n)}$, being r the distortion level. Let $x^{(n)}$ and $x^{(n,r)}$ be the numerical representations of $I^{(n)}$ and $\bar{I}^{(n,r)}$, respectively. Finally, let $q^{(n)}$ and $q^{(n,r)}$ be the quality levels for $I^{(n)}$ and $\bar{I}^{(n,r)}$, respectively, determined via subjective testing. The proposed system (see figure 1) compares the numerical descriptors, $\{x^{(n)},x^{(n,r)}\}$, and estimates the discrepancy,$d_S\left(q^{(n)},q^{(n,\,r)}\right)$, between the subjective scores associated with the images. At runtime, vector $x^{(n)}$ is worked out at the signal source, while $x^{(n,r)}$ is computed on the receiver side, and the two vectors are eventually processed to obtain quality estimates.

Having defined the set A=$\{a_1, a_2, \ldots, a_l\}$ of distortions of interest, the system relies on a dedicated quality predictor for each of them. As no a-priori knowledge can be assumed on the nature of the distortion affecting the input signal, a distortion detector is first needed to identify the distortion ($a_i \in$A) applied to the image$I^{(n)}$. This first module forwards the image descriptors to the appropriate quality predictor $\Omega(a_i)$, which finally provides an estimation for the difference in quality $d_S\left(q^{(n)},q^{(n,\,r)}\right)$ between the reference and incoming signal, distorted by a_i.

3 Objective Quality Metric

The color distribution across one image can be consistently altered due to the distortion impact. To describe such changes, second order histograms [25, 13] are powerful tools, representing the joint occurrence of a pair of colors throughout the image. Hence, the present research exploits features derived from the co-occurrence matrix to construct an objective description of the image. To represent the color, the luminance (Y) layer of the YC_bC_r colorspace is chosen. The relevance of luminance in quality assessment has been already extensively proved [11]. Furthermore, video streams are usually encoded in the YC_bC_r colorspace, hence, in a real time perspective, the luminance channel would be immediately available for computation.

The luminance distribution-based objective metric is computed in three steps: first locally, on a block-by-block basis, then gathering the obtained values into a single statistical global descriptor, to reduce the number of relevant values to be processed by the Machine Learning based assessment system. Finally, a feature selection procedure is applied, to limit information redundancy in the global descriptor.

3.1 Co-occurrence Matrices

The co-occurrence matrix measures local correlation among gray tones within an image subregion. Defining a region a, including $H_a \times W_a$

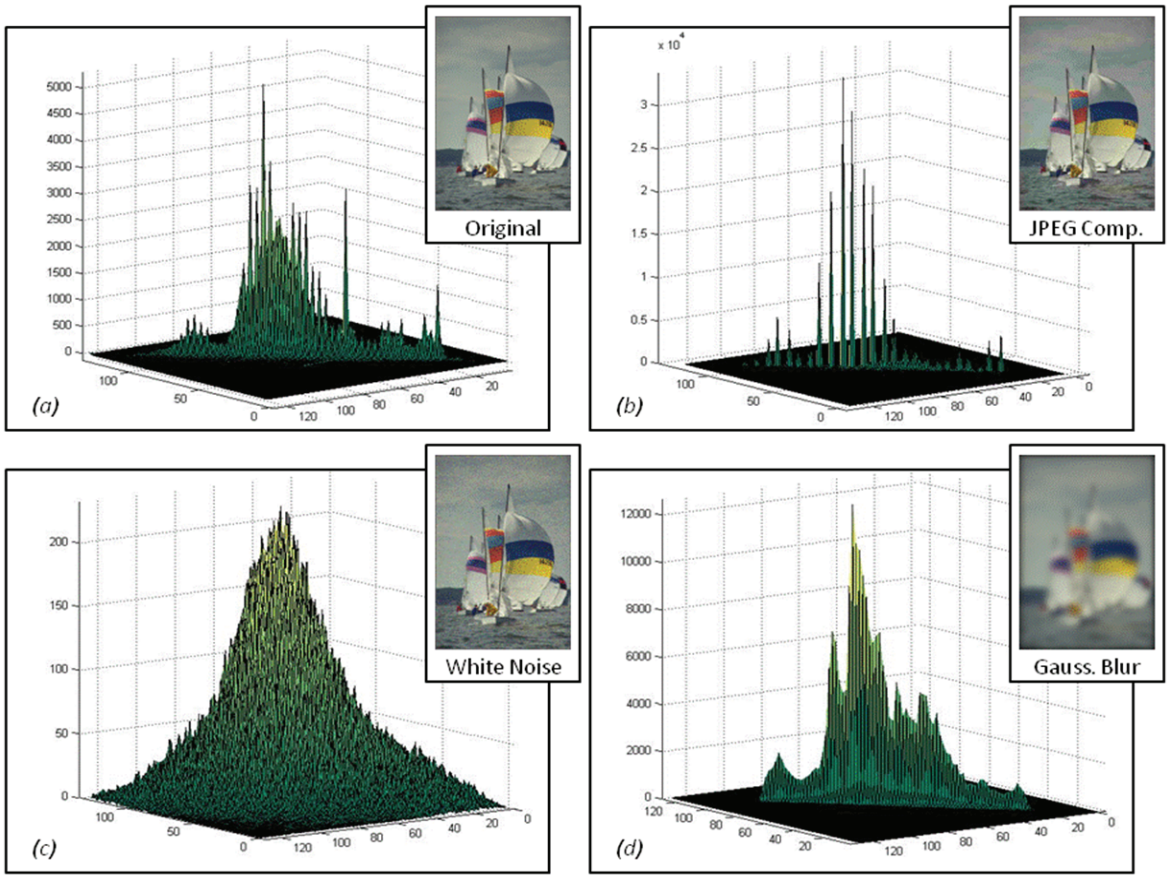


Figure 2. Effects of distortions on the color distribution. The co-occurrence matrix is shown for the original image (a), and the same image with Jpeg Compression (b), White Noise (c) and Gaussian Blur (d) applied, respectively.

pixels, each matrix element $C_{i,j}^{(a)}(\lambda,\theta)$ of the co-occurrence matrix $\mathbf{C}_a$ describes a joint occurrence of luminance levels, and counts the pairs of pixels in a that: (1) have gray levels i and j, and (2) are separated by λ radial units at an angle q to the horizontal axis. Formally, C_a is defined as:

$$C_{i,j}^{(a)}(\lambda,\theta)=\left|\left\{(m,n),\ \begin{matrix}0\le m\le W_a-1\\ 0\le n\le H_a-1\end{matrix},\ \text{s.t.}\ \begin{matrix}a[m,n]=i\\ a[m+\Delta h,n+\Delta v]=j\end{matrix}\right\}\right|$$

$$with\ i,j\ =0,\ldots,N_g \qquad (1)$$

where Dh and Dv are the horizontal and vertical displacements in the q direction, respectively:

$$\Delta h=\lambda\lceil\cos\theta\rceil,\Delta v=\lambda\lceil\sin\theta\rceil,0\circ<q\le 90\circ \quad (2)$$

$$\Delta h=\lambda\lfloor\sin\theta\rfloor,\Delta v=\lambda\lceil\cos\theta\rceil,90\circ<q\le 180\circ \quad (3)$$

$\mathbf{C}_a$ is eventually a $N_g\times N_g$ matrix, where N_g is the number of luminance quantization levels.

Figure 2 illustrates the Co-occurrence matrix extracted from the original image *Sailing_2* of LIVE database and from three differently distorted versions of it. Axis X and Y are indexed by color bins, while the Z axis indicates the joint occurrence of the given pair of colors at a predefined distance 1 pixel in the horizontal direction (i.e., $l=1$ and $q=0$). It is easily noticeable how different distortions cause peculiar changes to the second-order histogram when compared with the original signal (figure 2(a)). For example, JPEG compression (figure 2(b)) causes a severe quantization of the histogram, while white noise (figure 2(c)) spreads the informative content across the whole matrix. Such a marked behavior suggests the possibility that a metric based on the analysis of the co-occurrence matrix variations could efficiently reveal both the distortion affecting the image and its amount.

3.2 Local Metric Computation

Luminance distribution information is collected locally, on adjacent non-overlapping subregions of the image sized N_axN_a pixels. For each region, a Co-occurrence matrix is computed, and, from it, local features values.Indeed, for optimization purposes, co-occurrence matrices can also be characterized by using a set of scalar descriptors (features), statistically-based and image-description oriented [33]. All features are implicitly indexed by the image region, a, from which the matrix is calculated. To minimize computational cost the present research adopts a subset, $\Phi = \{f_u; u = 1,..,N_f\}$, of N_f=10 features that have already been tested to be successful [23, 22].

The block size, N_a, plays a crucial role in the reliability of the image description, and the image size should be taken into account to ensure proper sampling by a sufficient number of measures. Moreover, the features derived from the co-occurrence matrix $C^{(a)}(\lambda,\theta)$ may suffer from border effects. The percentage of pixels that do not enter the computation of $C^{(a)}_{i,j}(\lambda,\theta)$ decreases as N_a increases, hence one should avoid to use small block sizes; a typical setting is N_a>8.

3.3 Distortion Distribution Representation at a Global Level

The second step compresses the information obtained, and aggregates block-level data into one objective vector that characterizes the whole image. This procedure is performed consistently with the actual measuring procedure, since human assessors usually generate one overall quality score per image. Also, for computational and limited bandwidth reasons, block-based information must be reduced into a single, small-sized vector per image. To accomplish this, a percentile-based description of the distribution of each co-occurrence matrix feature is taken. As each feature reports on the effects of distortions on color distribution, the percentile based global vector can be considered an expression of the distortion action across the image.

For a parameter setting (l*,q*), the image $\bar{I}^{(n,r)}$ is represented by a set of N_f objective vectors, $x^{(n,r)}_{u,(\lambda*,\theta*)}$, u=1,..,N_f, which contain detail-related information. In the remainder of this paper, the index pairs (n,r) and (l*,q*) will be omitted wherever possible, in order to simplify the notation. The procedure to construct the objective vectors can be summarized as follows:

Inputs: a picture $\bar{I}^{(n,r)}$, a descriptive feature f_u and a the set of values $\mathrm{X}_u = \{x_{u,m};\ m = 1,..,N_b\}$, computed for each block m of the N_b blocks in the image.

- Compute a percentile-based description of the sample set X_u; let p_a be the a-th percentile:

$$\varphi_{\alpha,u} = p_\alpha(\mathrm{X}_u)$$

- Assemble the objective descriptor vector, x_u, for the feature f_u on the image $\bar{I}^{(n,r)}$ as

$$x_u = \{\varphi_{\alpha,u}; \alpha = 0,20,40,60,80,100\} \quad (4)$$

3.4 Feature Selection

The paper considers the statistical approaches to feature selection proposed in [22], which exploits Kolmogorov-Smirnov's test.

The proposed method tackles feature selection empirically; thus, the data set is obtained by applying the image-processing algorithm, $\zeta_q(\cdot)$, at different settings to a library of training images, Ω = $\{I^{(s)}$, s=1,...,$n_p\}$ and collecting the sample of processed images, $\bar{\Omega} = \{\bar{I}^{(s,q)}; s = 1,..,n_p; q = q_1,..,q_n\}$. Applying the feature-extraction process (as per Sect. 3.2) to each element in $\bar{\Omega}$, gives the eventual sample set V, which holds n_s=$q_n n_p$ elements and is given by:

$$V = \left\{x^{(s,q)}; s = 1,..,n_p; q = q_1,..,q_n\right\}. \quad (5)$$

The analysis selects from the complete set of candidate features, Φ, only the 'active' ones, i.e., those whose statistical properties depart significantly from their original values after applying a processing algorithm, $\zeta_q(\cdot)$. Thus, for each objective feature $f_k \in \Phi$, the analysis compares statistically two samples: one contains the values of f_k for a set of original images, the other holds the values of f_k for a set of processed images. To guarantee the independence of the two samples, the two sets of pictures are disjoint. The feature values are worked out on non-overlapping blocks of pixels randomly extracted from each image.

Table 1. Co-occurrence Matrix Descriptive Features

Feature name	Definition	Feature name	Definition
Absolute value	$f_1 = \sum_z z P_z(\lambda,\theta)$	Inverse Difference	$f_2 = \sum_{i,j} \frac{C_a(i,j,\lambda,\theta)}{1+(i-j)^2}$
Correlation	$f_3 = \left[\sum_{i,j} ij C_a(i,j,\lambda,\theta) - \mu_i^2\right] / \sigma_i^2$	Autocorrelation	$f_4 = \sum_{i,j} ij C_a(i,j,\lambda,\theta)$
Energy	$f_5 = \sum_{i,i} [C_a(i,j,\lambda,\theta)]^2$	Diagonal Energy	$f_6 = \sum_{i,j}^{i=j} [C_a(i,j,\lambda,\theta)]^2$
Entropy	$f_7 = -\sum_{i,j} C_a(i,j,\lambda,\theta) \log_2 C_a(i,j,\lambda,\theta)$	Differential Variance	$f_8 = \left[\sum_z (z-f_1)^2 P_z(\lambda,\theta)\right]^{1/2}$
Contrast	$f_9 = \sum_z z^2 P_z(\lambda,\theta)$	Differential Entropy	$f_{10} = -\sum_z P_z(\lambda,\theta) \log_2 P_z(\lambda,\theta)$
IMC	$f_{11} = \left(\sum_{i,j} C_a(i,j,\lambda,\theta) \log C_a(i,j,\lambda,\theta) - \sum_{i,j} C_a(i,j,\lambda,\theta) \log\left[\sum_j C_i^{(a)}\right]^2\right) \Big/ C_i^{(a)}$		

With $C_i^{(a)} = \sum_j C_a(i,j,\lambda,\theta)$, μ_I *and* σ_I *mean and standard deviation of* $C_i^{(a)}$, *respectively, and* $P_z(\lambda,\theta) = \sum_{i,j,|i-j|=z} C_a(i,j,\lambda,\theta)$

The mutual independence of the data sets allows one to use the Kolmogorov-Smirnov test [12] to disprove the null hypothesis, that is, to determine whether the two data sets for f_k have not been drawn from the same distribution. In that case, f_k is selected as an 'active' feature. KS has been preferred over parametric tests because one usually cannot assume a known distribution of the data involved.

The full pseudo-code of the feature selection algorithm is outlined in [22].

4 Connectionist Paradigms for Objective Quality Assessment

The system described in section 2 consists of two steps. Firstly, the distortion affecting the image has to be identified; secondly, the numerical representation of the image has to be mapped into a quality score by a dedicated predictor, which is specifically trained to assess image quality for a given distortion. The first layer is required to solve a classification problem. When aiming to detect the distortion a_i affecting the sample $I^{(n)}$, the set A=$\{a_1, a_2, \ldots, a_l\}$, the system is required to relate the input vector $\mathbf{x}_u$to a discrete value, representing a_i. On the other hand, the second layer maps the numerical descriptor $\mathbf{x}_u$ into a quality score, which cannot be expressed by discrete values to achieve acceptable accuracy. Therefore, this module can be designed to solve a regression problem.

In both cases, the use of connectionist paradigms is appealing. The machine learning world provides excellent tools able to handle both classification and regression supervised problems. Moreover, from a modeling point of view, such methods appear particularly suitable to model a highly non-linear context such as perception. Among others, Support Vector Machines (SVM) proved to be effective both for classification and regression tasks.

In a most general setting, one has a data sample, X, holding n patterns: each pattern includes a data vector, $\mathbf{x}\in R^m$, and its associate 'target' label, y. Classification problems involve a binary setting $y\in\{-1,+1\}$, whereas a regression problem is tackled when target values are expressed by continuous values, e.g., $y\in[-1,1]$. The learning phase uses

both **x** and y to build up a decision rule; at run-time, the trained machine processes unseen data and associates every input with a prediction of its target,$\hat{y}$.

The regression strategy implements the decision function, $\hat{y}= f(\mathbf{x})$ as a weighted series, whose basic terms, $f(\mathbf{x})$, typically embed nonlinear functions:

$$\hat{y} = f(x) = \sum_i \beta_i \phi_i(x) + \beta_0 \qquad (6)$$

Classification machines just yield a binary output by applying the operator $sign(.)$ to $f(\mathbf{x})$.

In the practical design of any estimator, the training set TG = $\{(\mathbf{x}_i,\mathrm{y}_i);\ i$=1,..,$n_p\}$ gives a sample-based formulation of the desired input-output mapping. For any empirical paradigm, the training procedure implements that mapping by fitting the degrees of freedom of the supported nonlinear estimator as per (6).

SVMs, in particular, tackle the pattern recognition problem within the Statistical Learning Theory [32] framework. A crucial element of it is the so-called kernel trick [29]: the kernel function $K(\cdot,\cdot)$ allows inner products of patterns in a higher dimensional, transformed space, though not involving the specific mapping of each single pattern. Given the points $\phi(x_1)$ and $\phi(x_2)$ in the feature space that are associated with $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively, then their dot product can be written as $\langle \phi(x_1), \phi(x_2) \rangle = K(x_1, x_2)$.

4.1 Support Vector Machines for Classification

In the case of binary classification problems, SVM relies on the solution of the following Quadratic Programming problem to set the free parameters in (6):

$$\min_{\alpha} \left\{ \frac{1}{2} \sum_{l,m=1}^{np} \alpha_l \alpha_m y_l y_m K(x_l, x_m) - \sum_{l=1}^{np} \alpha_l \right\}$$
$$subject\ to: \begin{cases} 0 \le \alpha_l \le C, \forall l \\ \sum_{l=1}^{np} y_l \alpha_l = 0 \end{cases} \qquad (7)$$

In (7), α_l are the SVM parameters setting the class-separating surface and C is a fixed regularization term that rules the trade-off between accuracy and complexity.

Problem setting (7) has the crucial advantage of involving a quadratic-optimization problem with linear constraints, ensuring that the solution is unique. Actually, the specific choice for the kernel parameters $\{C, \sigma\}$ has an impact on the eventual generalization performance of the SVM. Both theoretical [32] and empirical [2] approaches can be adopted to determine the generalization limits. The present research follows an empirical approach involving k-fold cross validation [2].

4.2 Support Vector Machines for Regression

A SVM is used in the proposed system also to map feature-based image descriptions into scalar values that represent the perceived image quality. As such, the objective quality assessment model can be regarded as a regression problem, in which learning evolves according to an empirical sample.

In the present application, the vector, **x**, contains a feature-based description of an image, while the value y_i represents the (normalized) quality score associate with that image. SVMs regression models approximate the target function for an input vector, **x**, as

$$\hat{\gamma}_{SVM}(x) = \sum_{i=1}^{n_{SV}} (\alpha_i - \alpha_i^*) y_i K(\tilde{x}_i, x) + b \qquad (8)$$

where a_i, $a^*{}_i$ are positive parameters and b is a bias. The patterns $\{(\tilde{x}_i, y_i), i = 1, ..., n_{SV}\}$ used are a subset of the training set and are called 'support vectors'. Expression (8) shows that $\hat{\gamma}_{SVM}(x)$ is a series expansion having the kernel function $K(\cdot,\cdot)$ as a basis and involving part or all of the training patterns. Inner products can still be handled in the transformed space independently of the mapping of the original patterns; therefore, the use of the kernel trick also remains valid in regression problems.

The coefficients α_i, α_i^* and b in expression (8) must be adjusted in compliance with the input sample distribution so as to minimize some cost function measuring the deviation resulting from the approximation. To this end, Vapnik [32] suggested the use of ε-insensitive loss functions, which penalize the error whenever the absolute approximation error remains smaller than ε.

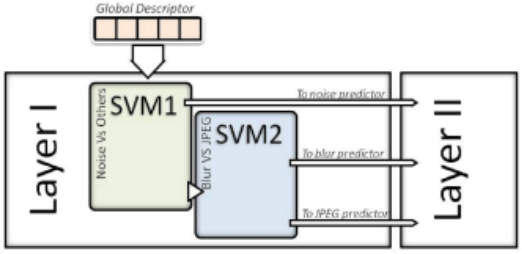


Figure 3. First Layer scheme. A first SVM detects Noisy images, while the second discerns between compressed and blurred ones.

5 Practical Implementation

In this section, a possible implementation of the double-layer system is proposed. A SVM-based classifier provides the distortion identification problem, while a SVM-based architecture maps feature-based description of images into quality scores.

As compared with [24], the present paper proposes a RR system that saves bandwidth and improves the computational cost of the assessment tool. Such goals are achieved 1) by exploiting the co-occurrence matrix for the feature-based representation of the image and 2) by developing an effective objective metric based only on two features.

5.1 Objective Metric Settings and Feature Selection

The following settings are applied to the metric described in section 3 to compute the objective descriptors $x_u^{(n)}$and$x_u^{(n,\,r)}$, corresponding to the reference image $I^{(n)}$ and the target image $\bar{I}^{(n,r)}$ respectively. The input image is divided into square sub-regions of 32x32 pixels and the co-occurrence matrix is computed on the luminance component (Y-layer) of the blocks, with settings λ=1 (neighboring pixels) and θ = 0 (horizontal direction). The set of features Φ defined in table (1) is then extracted for each block. Finally, the global descriptor is assembled by computing 6 percentiles of the distribution of each feature f_u, and combining them in the vector $x_u^{(n,r)} = \left\{\varphi_{\alpha,u}^{(n,r)}; \alpha = 0,20,40,60,80,100; u = 0,1,...,11\right\}$.

The feature selection procedure presented in section 3.3 is applied to Φ to select the two most significant features to be elaborated by the SVMs in layers I and II, independently on the specific problem to be treated. This choice is made in order not to inflate the bandwidth required by the RR model. In practice, the procedure described in 3.3 is applied to each of the learning tasks the system is supposed to tackle: distortion identification, quality mapping for Noise, Blur and JPEG. The features f_k resulting as active in the majority of the tasks are finally selected as the most effective for the whole system performance.

Eventually, the features Entropy and IMC (see table I) are selected. For each feature, the input of the SVM-based quality assessment system is obtained simply by combining the descriptors of the original and the distorted image:

$$z_u^{(n,\,r)} = \left[x_u^{(n)}, x_u^{(n,\,r)}\right] \tag{9}$$

As a result, the system processes 24 values in total, of which only twelve are required to be sent through the transmission channel together with the signal.

5.2 SVM-Based Quality Loss/Gain Quantification

The Support Vector machine technology is exploited for the prediction system implementation. In the first stage, the system is required to recognize which distortion is affecting the image under test. Hence, the role of the distortion identifier module is to solve a multiclass problem, associating each $\mathbf{z}_u$ (as per eq. 8) to distortion $a_i \in$A. We propose to implement a multiclass machine by connecting binary predictors in series, adopting a one-vs.-all strategy. The first SVM is trained to identify images distorted by a_1, and forward them to the a_1 distortion quantifier in layer II. The second SVM module recognizes images affected by a_2, and so on. Eventually, layer I will be made of l-1 SVMs, given l distortions of interest. In the present implementation, distortion caused by White Noise, Gaussian Blur and Jpeg Compression were considered (l

= 3). Layer I is implemented as in figure 3, including a first SVM handling noisy images and a second one dividing blurred from compressed samples. To privilege the system simplicity, a single feature is elaborated by layer I. The outcome of the feature selection is restricted to the better performing feature among the two classification tasks, resulting in feature Entropy.

The layer delegated for objective quality prediction (layer II) instead replicates as many independent predictors as the number of considered distortion (figure 4). Following an approach already experimented in [23], each module (also trained independently) is made of several SVM regressors, each fed by a different feature. These estimators are further integrated within an *ensemble* structure [14, 27]. Based on previous research [23, 24, 27] a simple output averaging strategy is chosen, being the most effective method for integrating the predictions of all estimators:

$$\hat{d}_S\left(q^{(n)}, q^{(n,r)}\right) = \frac{1}{U}\sum_u \hat{d}_S\left(q^{(n)}, q^{(n,r)}\right)\Big|_{f=f_u} \quad (10)$$

For the ensemble strategy to be successful, a basic requirement is to build independent estimators, based on the receptive fields theory [26]: the input space is divided into several, lower-dimensional subspaces, and a predictor is dedicated to each subspace.

Applying the coordinate-partitioning principle to the quality assessment domain leads to the specialization of each predictor on a single feature of the considered set. This setting not only validates the hypothesis of disjoint subspaces required for ensembles effectiveness, but also decreases the dimensionality of the input space, enhancing the SVM generalization ability.

Layer II involves therefore in its final configuration l quality prediction modules made each of U single SVMs gathered in an ensemble, being U the number of features selected for the task. In this research, layer II is made of three quality prediction modules (one for quantifying the effects of White noise, one for Gaussian Blur and one for JPEG compression), each including two SVMs fed by $\mathbf{z}_{Entropy}$ and $\mathbf{z}_{IMC}$(thus, $U = 2$), respectively, consistently with the feature selection procedure output.

5.3 Comparison with a Previous Implementation

The study presented in this paper represents an extension and improvement of a previously proposed work [24]. The previous implementation of the system (from now on 2LQA$_{old}$, as opposed to the one here presented, indicated with 2LQA$_{new}$) will be taken as term of comparison for the experimental validation. Nonetheless, the proposed setup already presents some peculiar advantages with respect to 2LQA$_{old}$, of which a short description will be given in the following.

2LQA$_{old}$ was characterized by an objective metric based on the color-correlogram [13] features. The color-correlogram is a second-order histogram which still measures the joint occurrence of colors at a given offset, but instead of considering only the co-occurrences along direction θ, involves in the computation color pairs in every possible direction, according to a predefined norm. By adopting the co-occurrence matrix, a very reliable distortion description is still available (see figure 2); nonetheless, one can lower the overall computational cost of the feature-extraction procedure, since at least 50% of the pixel pairs are excluded from the computation (when comparing with a correlogram computed with offset=1 and norm L1, being the less computationally heavy configuration).

The second relevant difference can be found in the feature selection procedure, which was previously performed empirically. For 2LQA$_{old}$ color-correlogram features were selected in order to maximize the generalization ability of each of the distortion-dedicated quality assessors in layer II. That choice led to the use of 4 features, corresponding to a 48-dimensional input vector for the system and to a double transmission overhead for the original image description.

Finally, while the implementation of the classification layer holds, the second layer of 2LQA$_{old}$ is based on CBP neural networks [28] ensembles. The effectiveness of this choice will have to be confirmed evaluating experimental results.

6 Experimental Validation

The second release of the LIVE database [11] was used as a testbed for the performance evalu-

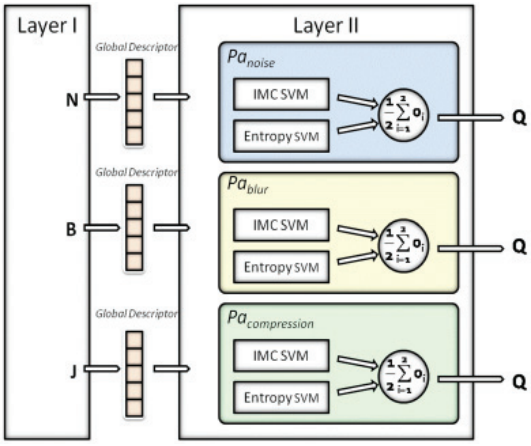


Figure 4. Layer II implementation. Three different modules are built to be specialized on the effect of one of the considered distortions. Each module includes an ensemble of 2 SVMs, each trained on a different descriptive feature.

ation of the proposed model, being a recognized benchmark in the image quality assessment field. LIVE database is based on 29 original images (from now on "image contents"). Each of this content is altered with different levels of five tipologies of distortions, originating then five subsets including both the original images and their impaired versions. For each of the 729 images in the dataset, a subjective score is provided, originated from panel sessions directly involving humans. Subjective scores, namely DMOSs, Differential Mean Opininon Scores, express the difference in quality between each impaired picture and its undistorted equivalent. Such values are the targets for layer II. For layer I, manual annotation of the distortion affecting the samples was sufficient, being independent from subjective evaluation.

The three datasets including images impaired with White noise, Gaussian Blur and JPEG compression were considered for validation.

To ensure robustness and avoid image content-related learning effects, a k-fold-like testing strategy was adopted. Five groups of images were created, each containing all the distorted versions of disjoint subsets of image contents. Both layers were then tested performing 5 runs, in each of which alternatively 4 of the 5 folds were used as training data and the remaining one was used as test data. In this way, the machines were tested on image contents never processed during the training.

Experimental results are presented as follows. First, details concerning the set up of the layer I are presented. Then, the development of the distortion-oriented assessment modules is discussed. Finally, the performance of the eventual quality assessment system combining the two layers is compared with other approaches proposed in the literature.

6.1 Precision in Distortion Identification

The two SVMs of the first layer were trained independently. The *k*-fold cross-validation technique was applied to tune the kernel parameters. For the first task, noisy images recognition, the SVM was trained on a dataset resulting from the merge of the three LIVE sets. A linear kernel handled the problem successfully, as shown in table II. The parameter C was finally set to $2.8*10^2$. The second SVM was trained on a subset of the previous dataset, including only blurred and compressed images. A normalized second order polynomial kernel as formulated in [4] was preferred for this task. Based on the cross-validation output, the parameter C was set to 1.310^5.

Table II reports the classification errors for each run and for both SVM classifiers. While the performance of the first classifier is almost perfect, the second SVM lacks in precision, due to the intrinsically more complex problem. The perceptual overlap between compression and blurring artifacts (JPEG compression causes also blur) is reflected also in the model: in this case a non-linear kernel was necessary, and the setting of the parameter C indicates increased complexity. Nonetheless, on average the percentage of misclassified images

is less than 6.5%, gaining more than 1% in accuracy with respect to the 2LQA$_{old}$ system. This result validates, at least for layer I, the choice of a co-occurrence matrix-based metric and of the feature selection output. Also, the use of a polynomial kernel seems to be more appropriate for the second task tackling.

6.2 Accuracy in Quality Loss Prediction

The three ensembles of SVMs implementing the second layer were each trained on a different dataset. The datasets for White Noise and Blurred images contained 145 patterns; the remaining testbed included 159 JPEG compressed images. DMOSs, originally ranging between [0,100] were remapped for computational reasons into the range [-1, +1]. The six SVMs were all equipped with a RBF kernel; thanks to the intrinsic flexibility of the system it was possible to select optimum models independently for every predictor. The final settings for the Noise predictor were $\{ C^{Noise}_{Entropy} = 100, \quad \sigma^{Noise}_{Entropy} = 2\}$ for the Entropy based SVM and $\{ C^{Noise}_{IMC} = 100, \quad \sigma^{Noise}_{IMC} = 1\}$ for the IMC-based SVM. Following the previous notation, the setting used for the Blur predictor were $\{ C^{Blur}_{Entropy} = 10, \quad \sigma^{Noise}_{Entropy} = 1\}$, $\{ C^{BLur}_{IMC} = 10, \quad \sigma^{Blur}_{IMC} = 0.5\}$; and for the JPEG Compression predictor $\{ C^{JPEG}_{Entropy} = 10, \quad \sigma^{JPEG}_{Entropy} = 1\}$, $\{ C^{JPEG}_{IMC} = 100, \quad \sigma^{JPEG}_{IMC} = 0.5\}$.

To evaluate the second layer performance, we report several parameters which measure the discrepancy between the estimated change in quality, $\hat{d}_S\left(q^{(n)}, q^{(n,\,r)}\right)$, and the actual variation provided by the LIVE database, $d_S\left(q^{(n)}, q^{(n,\,r)}\right)$. Four quantities are considered:

- The Pearson's Correlation Coefficient, ρ;
- The Spearman's Rank Order Correlation Coefficient, SROCC;
- The mean percentage prediction error, $\%\mu_{|err|}$, where $\mu_{|err|}$ is the value of the absolute prediction error between d_s and $\hat{d}_S$.
- The Root Mean Square prediction Error, RMSE, between d_s and $\hat{d}_S$.

The first two indicators are recommended by the VQEG committee for objective metric performance evaluation, being a measure of prediction monotonicity, i.e. of the consistency between the rank ordering of the samples given by the OQA and that provided by humans in subjective tests. Complimentary, $\mu_{|err|}$ and RMSE are given as measures of the prediction accuracy. Tables 3 to 5 show the output of the second layer of the proposed system compared to the performance of 2LQA$_{old}$.

In general, the system achieves considerably high accuracy, particularly when dealing with artifacts caused by Gaussian Noise, for which the percentage prediction absolute error is lower than 6% and the Correlation of the assessments with the subjective scores is 0.96. Dealing with compressed images, the RMSE is lower than 0.18 on a two points scale, which is acceptable for real-world applications. Finally, as for 2LQA$_{old}$, some lack in precision is presented by the Blur Predictor.

With respect to 2LQA$_{old}$, the proposed system gains in prediction monotonicity, for both Blur and Compression effects prediction, while a slight decrease in accuracy occurs for the three quality assessors. The differences in the systems setup should be taken in account: although the proposed system consistently simplifies the metric computation and the system requirements, it still allows obtaining an increase in correlation of the predicted scores, just slightly loosing in accuracy.

6.3 Comparison with Other Approaches

As a further validation of the system, table 6 and 7 compare the proposed approach with several well known OQAs. General Purpose RR OQAs are actually very few [34, 15], hence further comparison is provided with the two well-known FR metrics MSSIM and PSNR. The proposed system compares satisfactorily with the metric proposed by Li and Wang (yet not including distortion identification), and outperforms the method proposed in [34] for all distortion types except for Gaussian Blur Prediction, already recognized as a weak point of the model. In the comparison with MSSIM and PSNR, two details should be taken into account, namely (1) the difference in the original image availability for the computation of FR and RR metrics and (2) the fact the quantities reported were computed after a non-linear regression of all patterns for each dataset and the generalization ability of the obtained models was tested only using images that had been used

Table 2. Performance of each SVM machine for distortion identification in terms of % of misclassified patterns

	Proposed system		**2LQA$_{old}$ [27]**	
	Noise vs. All	**Blur vs. JPEG**	**Noise vs. All**	**Blur vs. JPEG**
Run #1	0.00%	1.49%	0.00%	1.49%
Run #2	1.04%	3.18%	1.07%	1.58%
Run #3	0.00%	3.22%	1.09%	4.84%
Run #4	0.00%	14.51%	0.00%	19.35%
Run #5	0.00%	10.00%	0.00%	12.76%
Average	**0.26%**	**6.48%**	**0.26%**	**7.56%**
Kernel	Linear	Polynom.	RBF	RBF
C	$2.8 * 10^2$	$1.3 * 10^5$	10^4	10^5

Table 3. Performance of the quality estimator for noisy images in terms of correlation (Pearson and Spearman Coefficients and errors (absolute and RMSE) between predicted and subjective quality scores.

	Proposed system				**2LQA$_{old}$ [27]**			
	ρ	SROCC	$\mu_{\|err\|}\%$	rmse	ρ	SROCC	$\mu_{\|err\|}\%$	rmse
Run #1	0.954	0.921	6.376	0.143	0.981	0.971	3.119	0.082
Run #2	0.942	0.951	5.864	0.149	0.937	0.944	5.216	0.163
Run #3	0.979	0.969	4.591	0.115	0.989	0.984	2.677	0.070
Run #4	0.954	0.956	5.777	0.142	0.985	0.982	3.122	0.079
Run #5	0.965	0.963	4.439	0.116	0.976	0.964	4.656	0.110
Average	**0.959**	**0.952**	**5.409**	**0.133**	**0.974**	**0.969**	**3.758**	**0.101**

Table 4. Performance of the quality estimator for blurred images in terms of correlation (Pearson and Spearman Coefficients and errors (absolute and RMSE) between predicted and subjective quality scores.

	Proposed system				**2LQA$_{old}$ [27]**			
	ρ	SROCC	$\mu_{\|err\|}\%$	rmse	ρ	SROCC	$\mu_{\|err\|}\%$	rmse
Run #1	0.933	0.897	7.867	0.189	0.946	0.946	5.365	0.136
Run #2	0.915	0.894	7.724	0.183	0.668	0.643	12.328	0.334
Run #3	0.839	0.842	8.653	0.213	0.966	0.972	3.758	0.100
Run #4	0.936	0.933	7.139	0.169	0.914	0.912	6.469	0.154
Run #5	0.812	0.834	14.270	0.392	0.892	0.868	10.244	0.240
Average	**0.887**	**0.880**	**9.130**	**0.229**	**0.877**	**0.868**	**7.633**	**0.193**

Table 5. Performance of the quality estimator for JPEG Compressed images in terms of correlation (Pearson and Spearman Coefficients) and errors (absolute and RMSE) between predicted and subjective quality scores.

	Proposed system				**2LQA$_{old}$ [27]**			
	ρ	SROCC	$\mu_{\|err\|}\%$	rmse	ρ	SROCC	$\mu_{\|err\|}\%$	rmse
Run #1	0.914	0.902	7.078	0.181	0.944	0.922	5.367	0.138
Run #2	0.886	0.872	7.522	0.190	0.857	0.864	7.186	0.207
Run #3	0.944	0.894	6.217	0.165	0.920	0.860	7.094	0.179
Run #4	0.934	0.923	6.287	0.151	0.939	0.912	4.579	0.121
Run #5	0.910	0.908	7.992	0.207	0.882	0.868	8.712	0.225
Average	**0.917**	**0.900**	**7.019**	**0.179**	**0.908**	**0.885**	**6.588**	**0.174**

in the training process.

7 Conclusions

A Reduced-reference, double layer system for objective image quality assessment is proposed. This general purpose system is designed to first recognize which distortion is affecting the image, and then to quantify the quality loss caused by the presence of such distortion. Both layers are supported by Support Vector Machines, trained in the first case to correctly classify images according to the distortion affecting them, and in the second case to understand the mapping between a numerical representation of the image and the quality impairment brought about by the applied distortion. The numerical description of the image (Objective Metric) is designed to minimize both the computational cost and the transmission bandwidth requirements.

The proposed implementation of the system allows consistent savings in computational time and bandwidth. With respect to the system presented in [24], a co-occurrence matrix is used in place of the most expensive Color Correlogram, reducing the computational time up to 50%. The stricter feature selection, applied offline, selects only two values to be computed from the original and distorted signals, compared to the four required before. This brings a twofold benefit. Firstly, it decreases the computational effort necessary for the numerical description of the images. Secondly, it allows characterizing one image with 24 values, i.e., with 96 bytes, on common 32-bits architectures. As a consequence, on the sender side, only two quantities have to be computed, and an overhead $rr_{info} <$ 0.05 KB is sent throughout the channel as metadata. This is of the major importance for emerging multimedia technologies, such as video streaming on mobile phones, for which not always large bandwidth availability can be assumed.

The performance of the system is not compromised with respect to its less efficient version [24], also well comparing with the state of the art of available Reduced-Reference quality assessment metrics.

A possible limitation of the proposed system is the current inability of handling the combined effect of different distortions. A heavily blurred image presenting traces of noise would be at present processed as if no noise was applied. However, different artifacts contribute in a different way to the final quality evaluation; therefore, all of them should be taken into account. Focusing on predicting the annoyance of different types of visual artifacts, would be greatly beneficial, allowing the abstraction of their perceptual impact from the actual distortion producing them. Further effort should then be put in understanding how to combine the impact of different distortions in estimating the overall quality of the picture. Needless to say, intensive subjective studies are required to make this development concrete.

References

[1] V. Baroncini. New tendencies in subjective video quality evaluation. *IECIE Transactions on Fundamentals*, 11(89):2933–2937, 2006.

[2] Lugosi G. Bartlett P., Boucheron S. Model selection and error estimation. *Machine Learning*, 48(1–3):85–113, 2002.

[3] H.R. Sheikh.and A.C. Bovik. Image information

and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006.

[4] A. Boni D. Anguita, S. Ridella, F. Rivieccio, and D. Sterpi. *Theoretical and Practical Model Selection Methods for Support Vector Classifiers*, pages 159–179. Springer, 2005.

[5] P. Engeldrum. *Psychometric scaling: a toolkit for imaging systems development*. Imcotek Press, Winchester, 2000.

[6] A.M. Eskicioglu and P.S. Fisher. Image quality measures and their performance. *IEEE Trans. on Communications*, 43(12):2959–2965, 1995.

[7] D.D. Giusto G. Ginesu, F. Massidda. A multifactors approach for image quality assessment based on a human visual system model. *Signal Processing: Image Communication*, 21:316–333, 2006.

[8] Z. Gao and Y.F. Zheng. Quality constrained compression using dwt-based image quality metric. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(7):910–922, 2008.

[9] A.C. Bovik H.R. Sheikh and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *IEEE Trans. Image Process*, 14(11):1918–1927, 2005.

[10] L. Cormack H.R Sheikh, Z. Wang and A.C. Bovik. Live image quality assessment database at http://live.ece.utexas.edu/research/quality. Technical report.

[11] M.F. Sabir H.R. Sheikh and A.C. Bovik. A statistical evaluation of recent full reference image quality assessment algorithm. *IEEE Trans. Image Processing*, 15(11):3441–3452, 2006.

[12] R.G. Laha I.M. Chakravarti and J. Roy. *Handbook of Methods of Applied Statistics*, volume I. John Wiley, 1967.

[13] S. Ravi Kumar J. Huang, M. Mitra, W.J. Zhu, and R. Zabih. Image indexing using color correlograms. In *Proc. IEEE CVPR '97*, pages 762–768, 1997.

[14] R.P.W. Duin J. Kittler, M. Hatef and J. Matas. On combining classifiers. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20:226–239, 1998.

[15] Q. Li and Z. Wang. General-purpose reduced-reference image quality assessment based on perceptually and statistically motivated image representation. In *IEEE International Conference on Image Processing*, San Diego, CA, 2008.

[16] H. Liu and I. Heynderickx. A perceptually relevant no-reference blockiness metric based on local image characteristics. *EURASIP Journal on Advances in Signal Processing*, 2009.

[17] D. Barba M. Carnec, P. Le Callet. Objective quality assessment of color images based on a generic perceptual reduced reference. *Signal Processing: Image Communication*, 23:239–256, 2008.

[18] I. Van Zyl Marais and W.H. Steyn. Robust defocus blur identification in the context of blind image quality assessment. *Signal Processing: Image Communication*, 22:833–844, 2007.

[19] S. Winkler P. Marziliano, F. Dufaux and T. Ebrahimi. Perceptual blur and ringing metrics: application to jpeg2000. *Signal Processing: Image Communication*, 19:163–172, 2004.

[20] ITU-T Recommendation P.911. *Subjective audiovisual quality assessment methods for multimedia applications*. Geneva, 1998.

[21] T.N. Pappas and R.J. Safranek. *Perceptual criteria for image quality evaluation*. Stateplace New York: Academic, 2000.

[22] R.Muijs P.Gastaldo, R. Zunino and I. Heynderickx. Building neural systems for no-reference quality assessment. In *Proc. of the First International Workshop on Video Processing and Quality Metrics(VPQM'05)*, 2005.

[23] Zunino R. Redi J., Gastaldo P. and Heynderickx I. Co-occurrence matrixes for the quality assessment of coded images. In *International Conference on Artificial Neural Networks (ICANN'08)*, 2008.

[24] Zunino R. Redi J., Gastaldo P. and SnplaceHeynderickx SnI. Reduced reference assessment of perceived quality by exploiting color information. In *Proc. International Conference on Artificial Neural Networks (ICANN'09)*, 2009.

[25] K. Shanmugam R.M. Haralick and I. Dinstein. Textural features for image classification. *IEEE Trans. On Systems, Man and Cybernetics SMC-3*, pages 610–621, 1973.

[26] D.E. Rumelhart and J.L. McClelland. *Parallel distributed processing*. MIT Press, Cambridge, MA, 1986.

[27] E. Bienenstock S. Geman and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–48, 1992.

[28] S. Rovetta S. Ridella and R. Zunino. Circular back-propagation networks for classification. *IEEE Trans. on Neural Networks*, 8(1):84–97, 1997.

[29] B. Schlkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[30] H.-J. Zepernick T.M. Kusuma. On perceptual objective quality metrics for in-service picture quality monitoring. In *Third ATcrc Telecommunications and Networking Conference and Workshop*, Melbourne, Australia, 2003.

[31] International Telecommunication Union. *Methodology for the subjective assessment of the quality of television pictures ITU-R BT.500*. 1995.

[32] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[33] S. Winkler. Issues in vision modeling for perceptual video quality assessment. *Signal Processing*, 78:231–252, 1999.

[34] E.P. Simoncelli Z. Wang. Reduced-reference image quality assessment using a wavelet-domain natural image statistic model. In *Proceedings of SPIE Human Vision and Electronic Imaging X*, volume 5666, pages 149–159, San Jose, CA, 2005.

[35] H.R. Sheikh Z. Wang and A.C. Bovik. No-reference perceptual quality assessment of jpeg compressed images. In *IEEE Int. Conf. Image Processing*, pages 477–480, Rochester, NY, 2002.

[36] H.R. Sheikh Z. Wang. and A.C. Bovik. *Objective video quality assessment*. CRC Press, Boca Raton, FL, 2003.

[37] H.R. Sheikh Z. Wang, A.C. Bovik and E.P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 2004.

[38] H.R. Sheikh Z. Wang, G. Wu, E.P. Simoncelli, E.H. Yang, and A.C. Bovik. Quality-aware images. *IEEE Transactions on Image Processing*, 15(6):1680–1689, 2006.

[39] S.Winkler Z. Yu, H.R. Wu and T. Chen. Vision-model-based impairment metric to evaluate blocking artifact in digital video. In *Proc. IEEE*, volume 90, pages 154–169, 2002.

[40] Y. Horita Z.M. Parvez Sazzad, Y. Kawayoke. No reference image quality assessment for jpeg2000 based on spatial features. *Signal Processing: Image Communication*, 23:257–268, 2008.

JAISCR, 2011, Vol.1, No.1, pp. 43–57

OPTIMIZING CONTROL BY ROBUSTLY FEASIBLE MODEL PREDICTIVE CONTROL AND APPLICATION TO DRINKING WATER DISTRIBUTION SYSTEMS

Vu Nam Tran[1], Mietek A. Brdys[1,2]

[1] *School of Electronic, Electrical and Computer Engineering*
College of Engineering and Physical Sciences.
University of Birmingham, Birmingham, B15 2TT, U.K.
e-mail: tvngachduoith@gmail.com

[2]*Department of Control Systems Engineering,*
Faculty of Electrical and Control Engineering,
Gdansk University of Technology, ul. Narutowicza 11/12, 80 – 233 Gdansk, Poland
e-mail:m.brdys@bham.ac.uk

Abstract

The paper considers optimizing Model Predictive Control (MPC) for nonlinear plants with output constraints under uncertainties. Although the MPC technology can handle the constraints in the model by solving constraint model based optimization task, satisfying the plant output constraints under the model uncertainty still remains a challenge. The paper proposes Robustly Feasible MPC (RFMPC), which achieves feasibility of the outputs in the controlled plant. The RFMPC which is applied to control quantity in Drinking Water Distribution Systems (DWDS) is illustrated by application to the DWDS example. In the simulation exercise, Genetic Algorithm is selected as the optimization solver and the reduced search space methodology is applied in the implementation under MATLAB-EPANET environment.

1 Introduction

Model Predictive Control has been an advanced technology and widely used in process control industry due to its ability to control multivariable systems with the presence of constraints. MPC actually belongs to a class of model based controller design concepts. The basic idea of the MPC algorithm remains unchanged regardless whatever kind of plant models are considered. It determines the optimal control actions by minimizing the user-defined objective function, or performance index. The current control actions are determined on-line, at each control step, by solving a finite-horizon open loop optimization problem, using the current state of the plant process as the initial state. However, only the first part of the optimized control input sequence is applied to the plant in the next time step. At the next control step, the prediction horizon moves forward and the same procedure repeats [2, 10].

Due to its operation on a receding horizon, MPC is also referred as receding control horizon or moving horizon optimal control. There are two significant factors that determine how effective an MPC is. The first factor is the accuracy of the plant model since it is explicitly used to predict the plant outputs. The second factor is how effective optimization solvers are. Although with the best plant models, MPC technology is still challenged by the uncertainty existing in the system such as a model structure error, a state estimation error, and disturbances. Fulfilling constraints is essential in many process plants for reason of safety, productivity, and environment protection. The controller out-

puts, which are based on the plant model, may not meet the plant output constraint due to the model-reality mismatch. The mismatch is often caused by the difference between predicted disturbance and actual disturbance. Feasible control input may become infeasible when they are applied to the plant if there is no robustly feasible controller. The robustness meeting of the output constraints or state constraints under system uncertainties is the main objective of the robustly feasible MPC. In this paper, the optimizing RFMPC is considered. The robust feasibility will be assessed by the robust output prediction over the reduced horizon. Safety zones are employed to tighten the output constraints in order to achieve robustly feasible control input. The control method is applied to control quantity in DWDS.

2 Representation of RFMPC

The structure of the RFMPC [3] consists of several units as illustrated in Fig.1.The MPC optimizer solves the MPC optimization task to produce control inputs. In this task, the plant outputs are predicted basing on the nominal model of the plant. In the nominal model, the disturbance inputs are represented by their predictions, while the internal model uncertainties are represented by a selected scenario. Before the control input is applied to the plant, its robust feasibility is assessed by the "Constraint Violation Checking" unit. The feasibility assessment is based on the robust output prediction that is generated by the "Robust Output Prediction" unit. Given the control input, the corresponding robust output predictions over the prediction horizon are a region in the output space in which all the plant outputs generated by the control input and all possible scenarios of the disturbance inputs are contained. The input robust feasibility is checked by confronting the output constraints with the robust output prediction. If the control feasibility passed its assessment, then the proposed control input is applied to the plant. Otherwise, robust output prediction is fed into the "Safety Zone Generator" unit. The safety zones as such are used to tighten the output constraints. The control actions produced by the MPC optimizer under modified (tighten) output constraints are expected to produce the real plant outputs that satisfy the plant constraints although they still may violate the modified constraints. Such control actions and the corresponding safety zones are called robustly feasible.

3 Robust Output Prediction (ROP)

Consider a continuous time plant, with the piecewise constant inputs, where input-output dynamics is modelled by the following discrete time systems:

$$\begin{cases} x(t+1) = f(x(t),u(t),z(t)) \\ y(t) = F(x(t),u(t)) \end{cases} \quad (1)$$
$$with\ the\ initial\ conditions:\ x(t_o) = x_o$$

where t is discrete time variable, $u(t)$ and $z(t)$ are the control and distuarbance input respectively. The disturbance input $z(t)$ is not known, and only bounds $z^{\min}$ and $z^{\max}$on its instantaneous values are available, that is $z(t) \in [z^{\min}, z^{\max}]$

Given inputs $u(t+k|t)$ for $k \in [1, H_p]$, where H_p is the prediction horizon, the plant output over H_p can be predicted by using the plant model as:

$$\begin{cases} x(t+k+1|t) = \\ \quad f(x(t+k|t),u(t+k|t),z(t+k|t)) \\ y(t+k|t) = F(x(t+k|t),u(t+k|t)) \end{cases} \quad (2)$$
$$with\ the\ initial\ conditions:\ x(t_o) = x_o$$

where $z(t+k|t)$ denotes prediction of the disturbance input at $t+k$ produced at the time instant t.

The vector of control inputs and outputs over the prediction horizon are respectively defined as:

$$\hat{U} = [u(t|t) \ldots\ u(t+H_m-1|t) \underbrace{u(t+H_m|t) \quad \cdots \quad u(t+H_p-1|t)}_{from\ (t+H_m)\ to\ (t+H_p-1)}]^T \quad (3)$$

$$\hat{Y} = [\ y(t+1|t) \quad \cdots \quad y(t+H_p|t)\ ]^T \quad (4)$$

where $u(t+i|t)$,$y(t+i|t)$ are the control inputs and model outputs at time $t+k$ produced at time instant t, and H_m,H_p are the *input horizon* and *prediction horizon* respectively.

Given the $x(t|t)$ and control inputs over the horizon $u(t+k|t)$, $k = 0,\ldots,H_p-1$. Let $y(t+k|t)$, $k = 1,\ldots,H_p$ be the corresponding plant outputs.

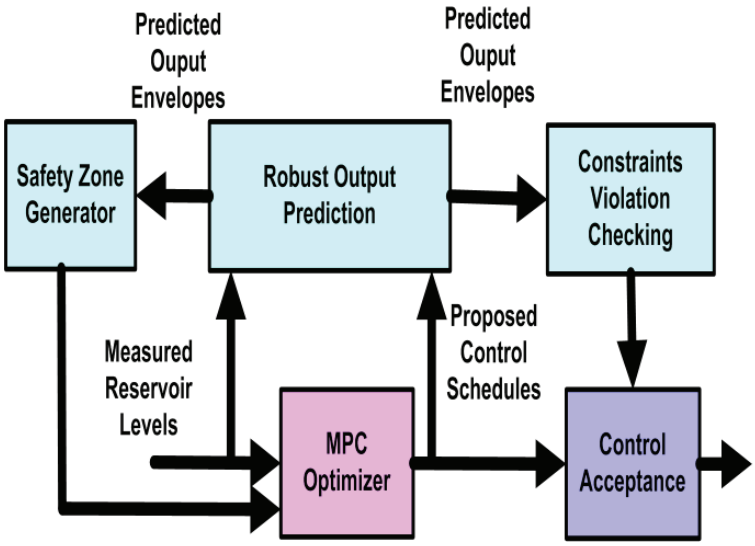


Figure 1. Structure of Robustly Feasible MPC

The robust prediction of $y(t+k|t)$ is composed of two envelopes:

$$Y_p^l = [y_p^l(t+1|t)\ldots y_p^l(t+H_p|t)]^T \tag{5}$$

$$Y_p^u = [y_p^u(t+1|t)\ldots y_p^u(t+H_p|t)]^T \tag{6}$$

where $y_p^l(t+k|t)$ and $y_p^u(t+k|t)$ are the upper and lower limits that robustly bound the plant output at prediction time step k:

$$\begin{array}{l} y_p^l(t+k|t) \le y(t)|_{t=t+k} \le y_p^u(t+k|t) \\ k=1,\ldots,H_p \end{array} \tag{7}$$

The least conservative bounding envelopes $y_p^l(t+k|t)$ and $y_p^u(t+k|t)$ can be determined as:

$$\begin{aligned} y_p^l(t+k|t) &= \min_{z(t|t),z(t+1|t),..,z(t+k-1|t)} y(t+k|t) \\ &= \min_{z(t|t),z(t+1|t),..,z(t+k-1|t)} F(x(t+k|t),u(t+k|t)) \end{aligned} \tag{8}$$

$$\begin{aligned} y_p^u(t+k|t) &= \max_{z(t|t),z(t+1|t),..,z(t+k-1|t)} y(t+k|t) \\ &= \max_{z(t|t),z(t+1|t),..,z(t+k-1|t)} F(x(t+k|t),u(t+k|t)) \end{aligned} \tag{9}$$

where uncertainty at time $t+k$: $z(t+k|t) \in [z^{\min}, z^{\max}]$, $\forall k \in \overline{1:H_p}$; the states $x(t+k|t)$ are obtained from the state space equations (2) with known initial condition $x(t|t)$.

Generating $y_p^l(t+k|t)$ and $y_p^u(t+k|t)$ also produce plant state bounding envelopes $x_p^l(t+k|t)$ and $x_p^u(t+k|t)$, for $k=1,\ldots,H_p$.

Since the robust output prediction is calculated over the horizon H_p, there are H_p optimization problems to be solved to find H_p values of $y_p^l(t+k|t)$ and $y_p^u(t+k|t)$. As k increases from 1 to H_p, the optimization also increases the number of variables from 1 to H_p.

Indeed, when k =H_p, (8) and (9) have H_p variables $z(t), z(2), \ldots, z(t+H_p-1)$. The more variables the optimization has, the more computing time the solvers require. As these computations are carried out online, it is desired to reduce the time computing as much as possible.

3.1 Stepwise Robust Output Prediction (SWROP)

In the previous section, solving optimization problems (8) and (9) give at least conservative solution of robust output prediction (ROP). This approach is so called the exact optimization method. In contrast to the exact optimization method, we propose in this section an approximated optimization method where its advantage is to reduce the optimization process computing time.

Instead of solving the optimization task with respect to k variables $z(t|t), z(t+1|t), \ldots, z(t+$

$k-1|t)$, one could approximate the least conservative robust output prediction (LCROP) by solving the optimization tasks ((8) and (9) with respect to only one variable $z(t+k-1|t)$ while $z(t|t), z(t+1|t), \ldots, z(t+k-2|t)$ are obtained from the optimization in the previous time steps. In other words, instead of simultaneous optimization with respect to all disturbance inputs, a step by step optimization is applied with respect to one disturbance input at the time starting with $x_p^l(t+k|t)$ and $x_p^u(t+k|t)$.

Hence,

$$y_p^l(t+k|t) = \min_{z(t+k-1|t)} y(t+k|t)\Big|_{z(t|t)=z^{\min}(t|t),\ldots,z(t+k-2|t)=z^{\min}(t+k-2|t)} \tag{10}$$

$$y_p^u(t+k|t) = \max_{z(t+k-1|t)} y(t+k|t)\Big|_{z(t|t)=z^{\min}(t|t),\ldots,z(t+k-2|t)=z^{\min}(t+k-2|t)} \tag{11}$$

where $z^{\min}(t+i|t)$ and $z^{\max}(t+i|t)$ can be obtained by solving:

$$z^{\min}(t+i-1|t) = \arg\min_{z(t+i-1|t)} y(t+i|t)\Big|_{z(t|t)=z^{\min}(t|t),\ldots,z(t+k-2|t)=z^{\min}(t+i-2|t)} \quad \forall i \in \overline{1:k} \tag{12}$$

and

$$z^{\max}(t+i-1|t) = \arg\max_{z(t+i-1|t)} y(t+i|t)\Big|_{z(t|t)=z^{\min}(t|t),\ldots,z(t+k-2|t)=z^{\min}(t+i-2|t)} \quad \forall i \in \overline{1:k} \tag{13}$$

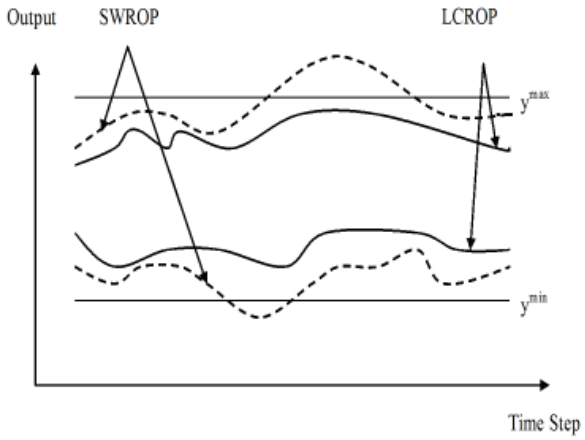


Figure 2. SWROP stays outside LCROP

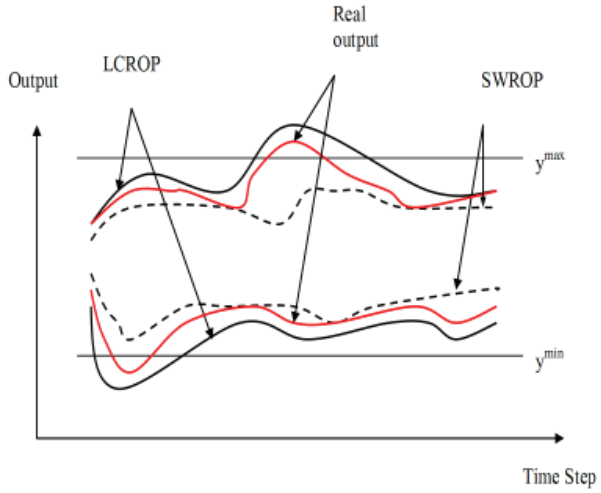


Figure 3. SWROP lies entirely inside LCROP

The resulting bounding envelopes are more conservative but the computing time is vastly reduced. Unfortunately, the expressions (10) and (11) generate the ROP only for some class of systems. The paper objective is to apply RFMPC to DWDS and there such a class has a clear interpretation, hence can clearly be identified. In order to assess the robust feasibility by SWROP, one should ensure that the LCROP entirely remains inside the SWROP as described in Fig.2. Otherwise the real output may

possibly violate the upper or lower constraint even though the SWROP does not, as described in Fig 3.

In practice, there are some classes of the system that have the characteristic as depicted in Fig.2 while some will have the characteristic of Fig.3. Hence, in order to avoid the situation of having robustly infeasible control input, designers in practice should take that into consideration of choosing the appropriate method to calculate the robust output prediction.

3.2 Reduced Robust Feasibility Horizon

So far the ROP has been considered over the whole output prediction horizonH_p set up for the RFMPC. This has been done in order to secure the existence of the robustly feasible safety zones at any control time step. However, as computing of ROP over H_p is computationally very demanding and this may not meet the time constraints set up by online computing requirements. We should consider reducing this demand by shortening the ROP horizon. Clearly the cost to be paid is an increased risk of non existence of robustly feasible safety zones at certain control time steps. As only the first control action out of a whole sequence determined by the RFMPC is applied to the plant, we must secure the robust feasibility over the first time step. This is how far we can go with the reduction of the ROP horizon from H_p to H_r. An attractive outcome of the ROP horizon reduction is that the very attractive computing SWROP method may become applicable over the reduced horizon while may not be applicable over the entire horizon. (see Fig.4)

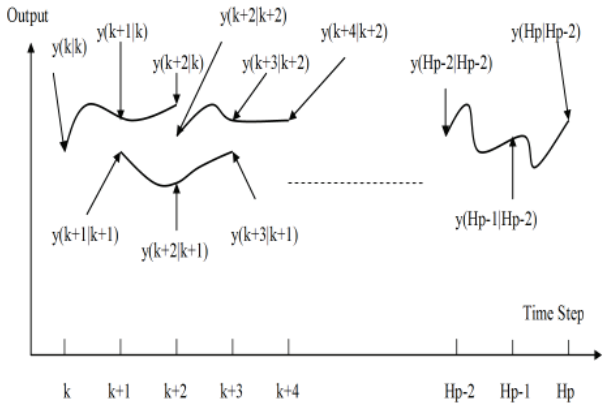


Figure 4. Example of reduced robust feasibility horizon to two time steps - $H_r = 2$

4 Safety Zone Generator

Using safety zones is not a new idea to meet the system constraint under unknown factors. It is widely used in engineering area, such as a conservative design in many electrical devices. When the input from the nominal model base MPC controller is applied to the plant, due to the uncertainties of the system, the output constraints may not be fulfilled and their violations may be unacceptable at certain time instants. If the violation occurs, it is important to correct or modify the constraints that apply to the nominal MPC. The safety zones generator is the unit that modifies the output constraints via the iterative scheme.

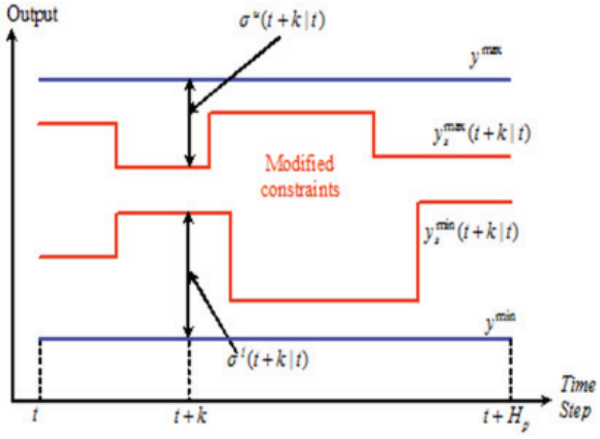


Figure 5. The output constraints modified by safety zones

Consider over the prediction horizon, the vectors of the lower and upper limits on the plant output $Y^{\min} = [\ y^{\min} \ \cdots \ y^{\min}\ ]^T$; $Y^{\max} = [\ y^{\max} \ \cdots \ y^{\max}\ ]^T$ and the vectors of the safety zones $\sigma^l = [\ \sigma_1^l \ \cdots \ \sigma_{H_p}^l\ ]^T$, $\sigma^u = [\ \sigma_1^u \ \cdots \ \sigma_{H_p}^u\ ]^T$ for the lower and upper output constraints, respectively where σ_i^l and σ_i^u are non negative real numbers. The vectors $Y_s^{\min} = Y^{\min} + \sigma^l$ and $Y_s^{\max} = Y^{\max} - \sigma^u$ are composed of the lower and upper bounds of the modified output constraints over H_p, respectively.

The "Safety Zones Generator" produces iteratively robustly feasible safety zones by using the following relaxation algorithm [3]:

(i) Set $x = [\ \sigma^l \quad \sigma^u\ ] = 0$;

(ii) Solve MPC optimization task with modified output constraintsH_p

(iii) A vector Vcomposed of the output constraint violation over the prediction horizon is calculated as:

$$V = [V_1 \cdots V_{2H_p}]^T \triangleq \begin{bmatrix} (Y^{\min} - Y_p^l)^T & (Y_p^u - Y^{\max})^T \end{bmatrix}^T$$

Define $f(V_i) \triangleq \max\{0, V_i\}$ and $C(\sigma^l, \sigma^u) \triangleq [f(V_1) \cdots f(V_{2H_p})]^T$
If

$$C(\sigma^l, \sigma^u) = 0 \tag{14}$$

is satisfied then go to step (vi), Else go to step (iv);

(iv) Calculate the safety zone corrections by using $\delta^{(k)} = -\nu C(x^{(k)})$ where $\nu = \max([diag[\nabla C(0)]]^{-1})$ is called the relaxation gain

(v) $x^{(k+1)} = x^{(k)} + \delta^{(k)}$, go to step (ii)

(vi) The robustly feasible safety zones have now been found and the control inputs $u(t|t)$ are applied to the plant.

5 Optimizing Control of DWDS by RFMPC

In the daily operation of water distribution systems, a period of water demand prediction ahead of current time is usually needed to be the basis for generating optimal pump actions so as to achieve certain control objective, e.g. the least pumping cost. Since too long or infinitive time horizon demand prediction is not accurate or unavailable, a relatively short prediction horizon is more realistic, and this is applied in a receding horizon manner, which forms the key idea of MPC technique. The corresponding optimization that reflects the control strategy is solved under the RFMPC structure and only the first part of the control input is applied. In this section, the formulation of the optimizing control problem and the simulation environment implementation by MATLAB – EPANET are presented. The Genetic Algorithm is selected as the optimization solver and the *reduced search space* methodology that has been used in the implementation will also be explained.

5.1 Formulation of the Optimizing Control Problem

The main goal of DWDS is to supply water to customers and satisfy their quantity and quality demand. There are two major aspects in the control of DWDS: quantity and quality. The quality control deals with water quality parameters. Having disallowed the concentration of the chemical parameter, for instance chlorine, cause serious heath dangers. Maintaining concentrations of the water quality parameters within the prescribed limits throughout the network is a major objective. When the quantity control is considered, the objective is to minimize the electrical energy cost of pumping, while satisfying consumer water demand and physical constraints such as pressure at nodes or reservoir levels, by producing optimized control input such as optimized pump speeds and valve control schedules [5]. The uncertainty is in the demand and structure and parameters of DWDS model. In this paper, only the quantity control aspect is considered by applying RFMPC technique. The quality issues are addressed in [4, 15] for example.

Objective function- pumping cost control: It is a very common control objective to achieve the least pumping cost while satisfying constraints. Moreover, in order to achieve a sustainable operation day after day, it is expected that tank levels can come back to their original states after a certain period. For the DWDS example, the network is operated daily and the prediction horizon is $H_p = 24h$. It is desired that after 24 hours, the tank level could have a similar level.

Hence, the overall objective function at $t = \bar{t}$ reads:

$$J = \sum_{t=\bar{t}}^{\bar{t}+H_p-1} \gamma(t)\Delta t \sum_{j=1}^{G_p} \sum_{i=1}^{U_j} \frac{\xi q_{j,i}(t)\Delta h_j(t)}{\eta_{j,i}(t)} + \rho \sum_{s=1}^{S} |r_s(\bar{t}+H_p) - r_s(\bar{t})|$$

where H_p is the prediction time horizon, ρ is a weighting factors, $\gamma(t)$ is a power unit charge in /kWh for the $(t+1)$ time stage, r_s is the sth reservoir/tank level, $s = 1, ..., S$, $\xi$is a unit conversion factor for electrical power relating water quantities to electrical energies, and η_i is the pump efficiency of the ith pump in the jth pump group, $i = 1, ..., U_j$ and $j = 1, \ldots, G_p$.

The decision variables are the pump speed over

the prediction horizon. They are automatically computed by the optimization solver to minimize the objective junction while satisfying the optimization constraints. The optimization constraints compose of equality constraints and inequality constraints. However, from the implementation point of view, the optimization constraints could be classified into implicit and explicit ones.

Explicit constraints: are often the control input constraints. It can be the sequence of the pump speed schedule or the ON/OFF state of pumps and need to satisfy the physical constraints. They are explicitly embedded into the optimization solver i.e. $u^{\min} \leq u(t+k|t) \leq u^{\max}$.

Implicit constraints: are the ones that have to be implicitly embedded into the optimization solvers. Implicit constraints are nonlinear and consist of all equality constraints defining the hydraulic equilibrium state of the system. They correspond to the set of flow continuity equations, volume/mass balance equations, and energy conservation equations. The output constraint such as tank heads and flows are also considered as implicit constraints. Implicit constraints are composed of:

- Nodal flow continuity equations: $\sum_{j\in J_i^-} q_j - \sum_{j\in J_i^+} q_j - d_i = \begin{cases} 0 & \text{for } i \in M \\ lq_i & \text{for } i \in M_l \end{cases}$
- Water elements head-flow equations: $h_{N_j^+} - h_{N_j^-} = \Delta h_j(q_j, u_j)$ where $q_j$is the flow at arc j (liter/sec); $h_i$is the head at node i (m); d_i is the demand flow at node i (liter/sec); lq_i is the leakage flow at node i (liter/sec); $u_j$is the control variable representing the state of valve or pump at arc j; Δh is the head-flow characteristic function at arc j; $M(M_l)$ is the set of non-leaky (leaky) nodes; $J_i^+(J_i^-)$ is the set of arcs whose start (end) node are i; and $N_j^+(N_j^-)$is the start (end) node j
- Volume mass balance equations of tanks/reservoirs
- Output constraints. They are in the form of lower and upper bounds on certain flows, junction heads, and on all tank heads in order to meet the tank capacity constraints.

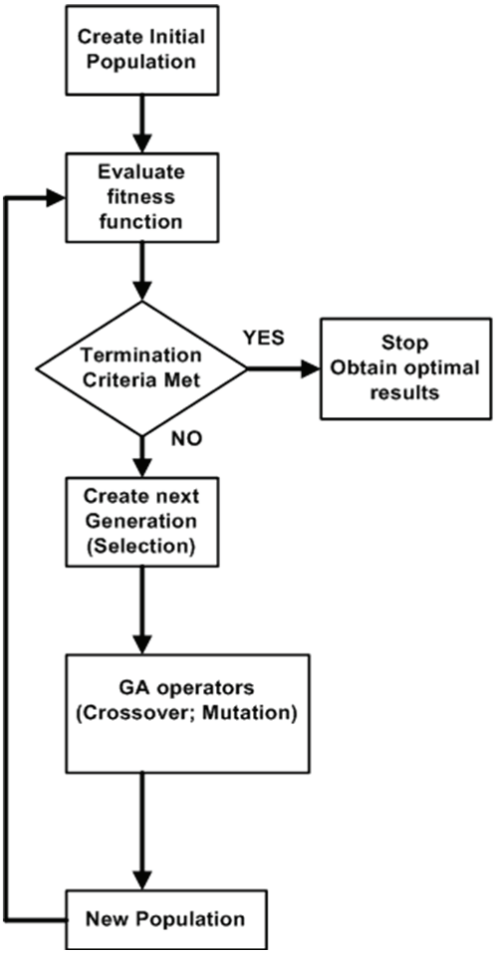


Figure 6. Flowchart of the standard GA

5.2 Genetic Algorithm – the Optimization Solver

GA represents a discrete variable optimization technique based on the principles of genetics and natural selection. The method was originally proposed by Holland in 1975 [9], and further developed by Goldberg in 1989 [7]. GA has been extensively used in many industrial engineering applications and so far represents one of the most commonly employed natural optimization techniques for design of water distribution networks as evidenced by use of GA for sizing of pipes [14, 17], evaluation of system reliability [16], and placement of early warning detection sensors [12]. GA has successfully solved the discontinuous, the non-differentiable, the non-convex, the multiple peaks, or highly nonlinear function optimization problems which traditional analytical and numerical methods are not well suited [8]. Advantages of this method include:

– The GA searches from a population of decision variable sets, not a single decision variable set.
– The GA can optimize discrete or continuous variables.
– The GA uses the objective function itself, not the derivative information. [8]
– The GA algorithm uses probabilistic (not deterministic) search rules.
– GA does not require discretization of state variables.
– GA does not require transition probabilities.
– GA models results in optimal or near optimal solutions.

However, GA has also some disadvantages:

– Cannot efficiently handle large number of constraints.
– Computationally difficult to provide very long string length with binary coding.
– Every iteration needs objective function evaluation.
– Global solutions are not guaranteed.

Briefly, the GA algorithm is initiated with a random population of individuals in which each individual is represented by a binary string (i.e., chromosome) for one possible solution. The standard structure of the GA is illustrated in Fig.6. The major components of the algorithm are explained as below:

1. At the beginning of the computation process, users specify the initial population in advance or GA randomly creates an initial population. For each population generation, a measure of the fitness with respect to an objective function is calculated.
2. If one of the pre-established criteria of the algorithm such as a number of generations, time limit, fitness limit, stall generations, stall time limit, and fitness tolerance are met, then the optimization terminate and the optimal results are obtained. Otherwise, go to step 3.
3. Based on this fitness value, the individuals are selected to create the next generation (Selection). They are called parents.
4. GA use operators to produce children (*offspring*) for the next generation of population. There are different operators e.g. crossover, mutation. Children are produced either by making random changes to a single parent (*mutation)* or by combining the vector entries of a pair of parents (*crossover)*. In the mutation process, all children are mutated with a certain probability, which ensures the probability of searching a particular subspace of the problem, the space is never zero. It is intended to prevent premature convergence and loss of genetic diversity. Whereas in the *crossover* process, the individuals are recombined to produce offspring. The purpose of crossover is to let individuals exchange useful information with each other, and get higher fitness in order to have better individual in the next generation and have preferable performance.
5. Use new generated population for a further run of the algorithm. Go to step 2.

In order to obtain the best result from GA, users usually need to experiment with different options. Selecting the best options for a problem involves trial and error. There are quite a number of different ways/options to improve results. Few of the most important factors that need to be tuned in order to get the good result are described as below:

– Population Size: represents the diversity of the population. It is one of the most important factors that determines the performance of the genetic algorithm performs is the *diversity* of the population. If the average distance between individuals is large, the diversity is high; if the average distance is small, the diversity is low. Getting the right amount of diversity is a matter of trial and error. If the diversity is too high or too low, the genetic algorithm might not perform well. However, the *Population size* should be at least the value of *Number of variables*, so that the individuals in each population span the space being searched.

– Initial range: specifies the range of the vectors in the initial population that is generated by a creation function. If users know or have knowledge approximately where the solution to a problem lies, the initial range should be specified so that the guess for the solution is contained and it can enable GA search more effectively. However, the genetic algorithm can find the solution even if it does not lie in the initial range, provided that the populations have enough diversity.

– Crossover fraction and Mutation function: Crossover fraction effect the ability of the algorithm to extract the best genes from different individuals and recombine them into potentially superior children. Different crossover fractions lead to different performances of GA. Similarly, different mutation function adds to the diversity of a population and thereby increases the likelihood that the algorithm will generate individuals with better fitness values. For certain problems, adjusting the amount of mutation fraction can possibly make significant improvement of the solutions.

– Initial population: has significant impact on GA's performance in the complex nonlinear constrained problem. The optimizing control problem of DWDS consists of many variables and nonlinear constraints that GA cannot directly handle. For GA in order for GA to perform effectively, initial population needs to be provided. Different initial population might lead to different solutions. Without knowledgeable initial population, GA are still able to randomly create initial population itself, but it could lead to the situations of no feasible solution being found, or take tremendous computing time to converge.

Regarding to the application of complex DWDS, the standard GA needs to be enhanced in order to exploit specific features of the optimization task and achieve required computing efficiency [11].

5.3 Simulation Environment Implementation

The computer implementation is based on MATLAB-EPANET environment. The optimization problems are solved by standard GA [6] which can be called through MATLAB Genetic Algorithm Toolbox [1]. The simulation implementation is illustrated in Fig.7. In order to solve the DWDS problem efficiently, the *reduced search space* methodology was employed by means of which the water distribution network simulator (EPANET) [13] is embedded directly to the optimization solver (GA). Specifically, in the Predictive Controller block, starting with the feasible set of control inputs, the optimization solver i.e. GA passes the control input into the EPANET for use in explicitly satisfying the implicit system constraints (equality constraints) and in evaluating the implicit bound constraints (inequality constraints). The water network information (i.e. nodal pressure, tank level, heads, flows) that are obtained by the EPANET are passed back to GA for determining the objective function. This process iteratively runs until the optimal control inputs at the present time step are found. The EPANET here plays as the *model* of the DWDS. The flow path of the optimization procedure within the Model Predictive Controller block is shown in Fig.8.

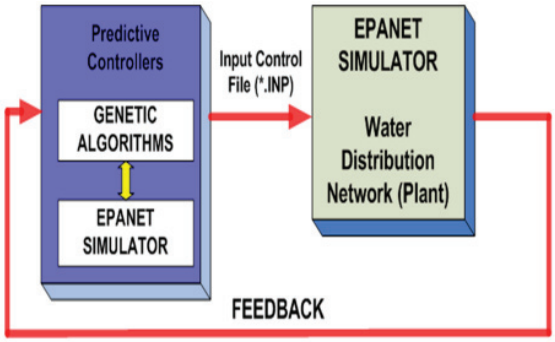


Figure 7. Simulation environment implementation

At each time step only the first part of the control inputs is applied to the Water Distribution Network (plant) block. The EPANET is used again as the *plant* of the DWDS. Note that there is always a model-plant mismatch i.e. disturbances (predicted demand and actual demand). The plant generates the massive output data in which only the states (tank levels) are fed back to the Model Predictive Controllers block for the next time step use.

The EPANET that is embedded into GA takes care of all hydraulic equations of DWDS. In other words, it reduces the number of equality constraints of the optimization problem; hence GA will only have to find the solution in the *reduced search*

space. Consequently, the overall computing time is reduced. It is an important feature of *reduced search space* methodology since the online operation of the RFMPC requires a tremendous computing effort.

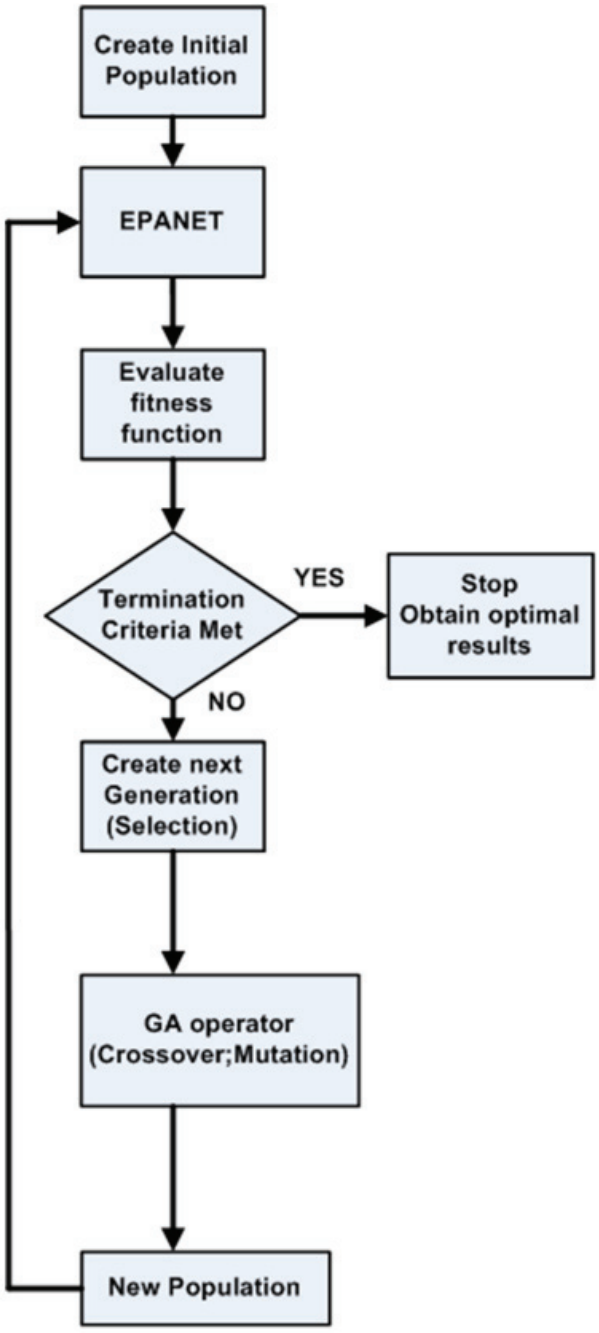


Figure 8. Optimization procedure of MPC

The main specifications of the software and hardware in the simulation:

- EPANET : Version 2.0
- MATLAB: Version 7.6.0.324 (R2008a)
- GA-Toolbox: Version 2.3 (2008a)
- CPU/Memory: 2.00 GHz CPU, 2.00GB RAM.
- Operating System: Windows XP Professional

5.4 Application of RFMPC to Example Case Study DWDS

The DWDS, which is depicted in Figure 9 includes 1 source reservoir and 1 storage tank. Water is pumped from the reservoir source by the pump station and can also be supplied by the storage tank (node 7).

The assumed positive flow is expressed in the way the link ID shows, e.g. link 203 denotes the positive flow direction from Node 2 to Node 3. The negative value of the flow indicates the flow direction is opposite to the assumed positive flow direction, e.g. $q_{402} < 0$ means the flow direction is from Node 3 to Node 2.

The prediction horizon is $H_p = 24$. The interested control input is pump speed sequence over 24 hours period. RFMPC is applied to produce control the input sequence. The tank level limits are: $r_s^{\min} \leq r_s \leq r_s^{\max}$, and these are the output constraints. The demands are predicted with the error of 10% at each consumption node.

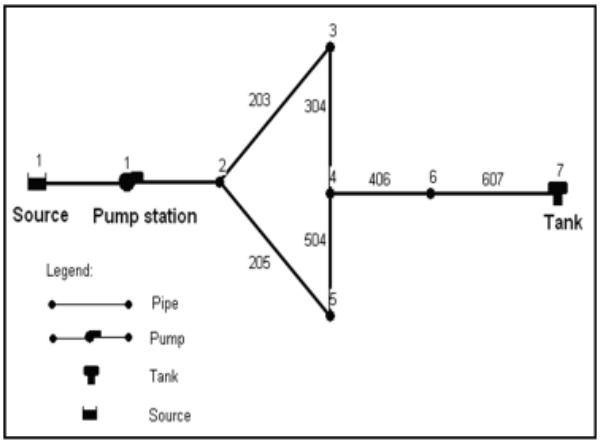


Figure 9. Diagram of an DWDS example

For the operational control purposes, the configuration of the DWDS example is given in the table 1, 2, 3, and 4 which includes the nodal elevation, nodal base demand, the operating constraints for nodal pressure, daily nodal demand profile, tank initial level and level range pipe, pipe and pump installation data, and time dependant electricity tariff.

Table 1. Nodal data for the pipe network

Node ID	Elevation(m)	Minimum head(m)	Maximum head(m)	Base demand(l/s)
2	15.0	18.0	32.0	5.0
3	14.0	16.0	30.0	5.0
4	12.0	12.0	28.0	5.0
5	14.0	14.0	30.0	5.0
6	8.0	10.0	28.0	30.0
1	5.0			
7	—— Tank/Reservoir nodes ——			

Table 2. Tank (Reservoir) data of the example DWDS

Node ID	Elevation(m)	Initial level above bottom(m)	Min level above bottom(m)	Max level above bottom (m)	Tank diameter(m)
7	10.0	5.0	3.0	10	15.0
1	15.0	————			

Table 3. Pipe data of the example DWDS

Pipe ID	Start node	End node	Length(m)	Diameter(mm)	C Value
203	2	3	1000	400.0	100
205	2	5	1000	400.0	100
304	3	4	1000	400.0	100
504	5	4	1000	300.0	100
406	4	6	1000	500.0	100
607	5	7	1000	500.0	100

Table 4. Pump data of the example DWDS

Pump ID	Head node	Tail node	Coefficient $\bar{A}$ (10^{-2})	Coefficient $\bar{B}$	Cut-off head $\bar{C}$ (m)
1	1	2	-0.5419	0	200

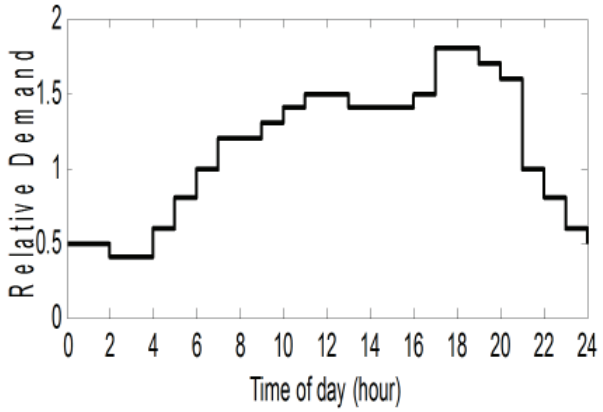


Figure 10. Daily demand profile

Optimization problems are formulated with a moving horizon $H_p = 24$hours ahead of present time and the sampling period is fixed to 1 hour. Electricity tariff difference is taken into account: low tariff is charged at 4.51p/kWh for the night time between 10:00pm–6:00am and high tariff is 9.72p/kWh for the daytime between 10:00pm–6:00am

5.4.1 Designing RFMPC

The MPC task is solved by GA solver with the optimization search in the reduced space. In this search, the GA is coupled to the EPANET simulator solving the DWDS equality constraints.

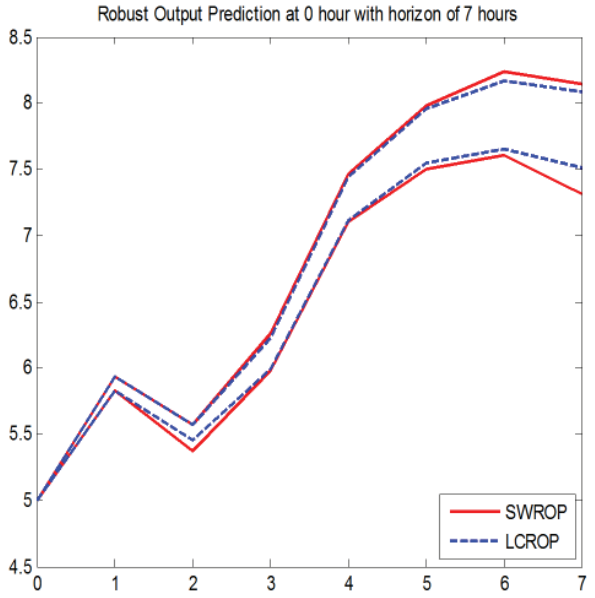


Figure 11. Robust output prediction at t = 0, Hr = 7

A method for generating ROP is chosen by observing the simulation results shown in Fig.11 The SWROP and the LCROP are applied at t = 0 over 7 time steps. It can be seen in Fig.11 that the SWROP method generates envelopes that are outside the region determined by the LCROP method. Hence, the SWROP is applicable to our example DWDS. Moreover, the envelopes calculated by the two methods are very close over the first 6 steps. The ROP horizon therefore is further reduced to 2 steps and the SWROP method is to be applied.

Also the relaxation gained in the algorithm for determining the robustly safety zones (RFSZ) is selected by simulation where several gain values are tried and the results are illustrated in Fig.12 The equality (14) in the step (iii) of the RFSZ relaxation algorithm has more than one solution. Clearly, the smaller safety zones are, the less conservative control actions are, and consequently a better controller performance is achieved. In Fig.12, this is obtained for small gain values. On the other hand, the computing time is essential; hence the number of iterations needed to reach the RFSZ should be minimized. This is obtained for high gain values as described in Fig.8. Therefore, gain $\nu = 0.6$ is chosen in order to trade between the two aspects.

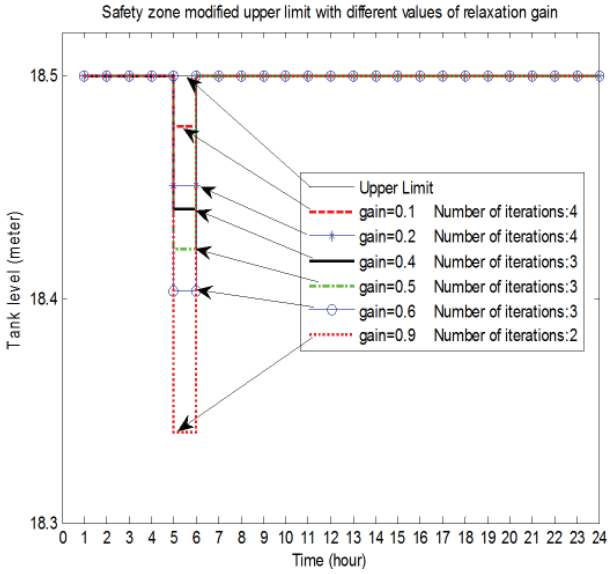


Figure 12. Robustly feasible safety zones and the corresponding modified tank upper limit for different relaxation gain values.

In the considered objective function of the corresponding MPC that has been applied to the DWDS, a long term sustainable operation of the distribution system is guaranteed by a periodical control of the tank level back to its original status. This

issue is embedded into the optimization problem as a suitable penalty term.

There are some parameters such as termination criteria, initial range, and population size in the GA options that have been chosen by experiments in order to obtain good simulation results within a reasonable period of computing time. The initial population is also provided to GA rather than let it create itself. The property settings of GA are listed as below:

```
StallTimeLimit=500;
PopulationSize=48;
StallGenLimit=10;
PopInitRange=[0.2;1.6];
```

5.4.2 Simulation Results

Firstly, the RFMPC is applied to the example DWDS at t = 0. Robust feasibility at the obtained control sequence is checked over the horizon $H_r = 2$ and the first two control inputs are assessed as robustly feasible. Hence, there is no need to activate the "Safety Zone Generator". In Fig.13 two tank trajectories are illustrated: one in a dash line is obtained by applying the control sequence to the model with the demand prediction while the second one in solid is the tank trajectory seen in the real system where the demand may differ from the predicted one up to 10%.

It can be seen in Fig.13 that the upper limit tank constraint is violated during 5 hour to 7 hour time period. Clearly, we are not aware of this violation at t = 0. However, a lesson to be learnt is that applying a whole control sequence obtained at t = 0 to the network is not recommended not only in this case but in general. Therefore, the RFMPC is kept applying to produce the control actions on-line by employing feedback and all its mechanism described in this paper.

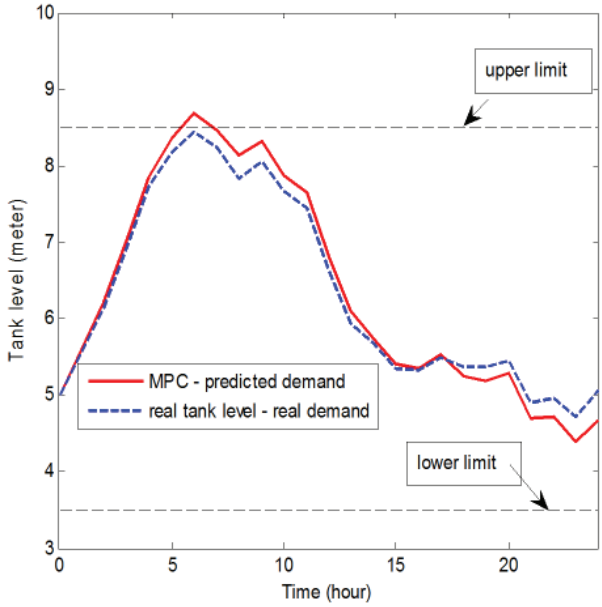


Figure 13. Predicted tank level trajectory by RFMPC over the horizon at time instant t = 0 and Hr = 2

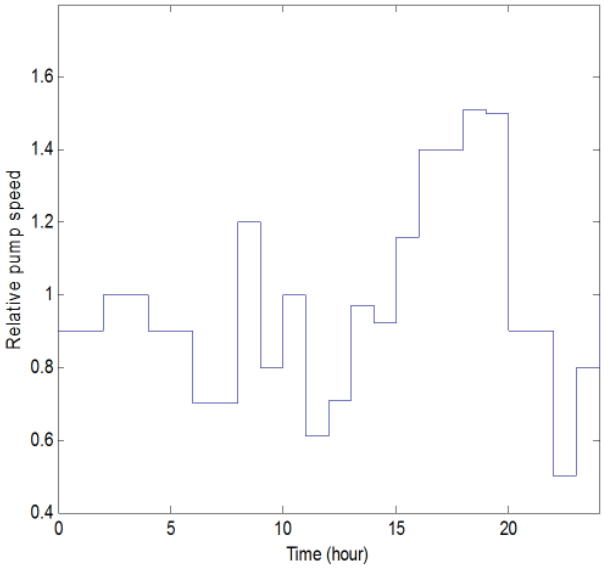


Figure 14. Control actions - relative pump speed

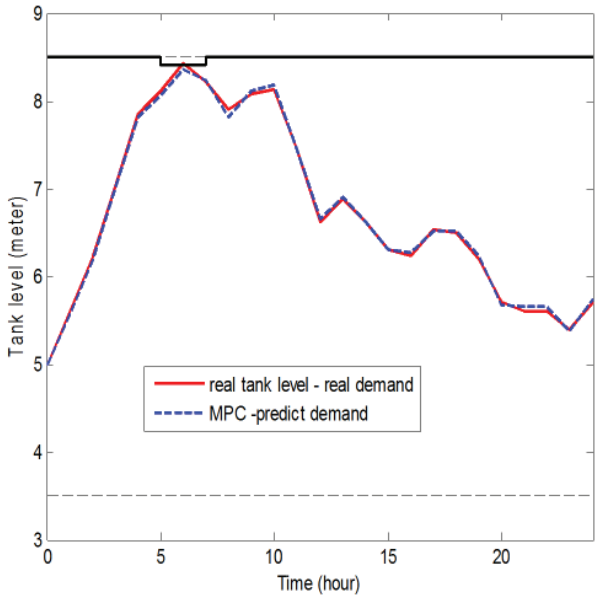


Figure 15. Tank trajectory over the 24 hours

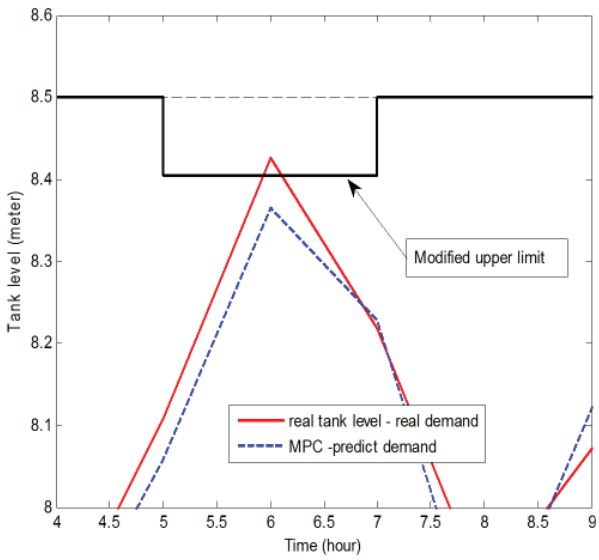


Figure 16. Zoom-in of Fig.10 during 4–9 hours

The results are illustrated in Fig.14–16. It can be seen in Fig.15 that the upper tank level constraint had to be modified by robustly feedback safety zones over 5, 6, and 7 time steps in order to achieve robust feedback of the control action over these time steps. Although the modification does not tighten the constraints excessively, its conservatism would be improved by extending the robust prediction horizon. The details of the situation over 5, 6, and 7 time steps are illustrated in Fig.16.

In order to assess the RFMPC feedback strength, the control actions generated on-line are also applied to the DWDS model. The resulting tank trajectory and the control input are shown in Fig.15 and Fig.14, respectively. The two trajectories are much closer in Fig.15 than in Fig.13. Hence, the possible impact of the feedback in compensating the demand error impact is noticeable.

Lastly, as shown in Fig.16, the modified constraints are satisfied in the model but not in reality. However, the actual constraint is met in reality, showing the effectiveness of the RFSZ mechanism.

6 Conclusions

This paper has further developed Robustly Feasible Model Predictive Control Method for the online optimizing control of nonlinear plants with output constraints under uncertainty. The RFMPC has been applied to quantity control in Drinking Water Distribution Systems. It has been illustrated by simulation based on an example DWDS. The effects of the robust output prediction, shortening the robust output prediction horizon, robustly feasible safety zones, and the feedback strength of RFMPC have been shown.

The combination of GA and EPANET via the reduced space methodology has been applied in the implementation of the DWDS. Nevertheless, meeting the requirement of computing time during online operation is still a big challenge for RFMPC designers. In addition to the reduced space methodology, the gradient-type optimization solvers can be utilized due to its advantage of taking the derivative information. This interesting and important topic is currently under future research.

References

[1] *MATLAB Genetic Algorithm Toolbox Manual.*

[2] A. Bemporad and M. Morari. Robust model predictive control: A survey,. In *Lecture Notes in Control and Information Sciences*, volume 245, pages 207–226. Springer-Verlag, 1999.

[3] M.A. Brdys and T. Chang. Robust model predictive control under output constraints. In *Proc. of the 15th IFAC World Congress*, Barcelona, 2002.

[4] M.A. Brdys, K. Duxinkiewicz, T. Chang, M.Polycarpou, A. Wang, J. Uber, and M. Propato. Hierachical control of integrated quality and quantity in drinking water distribustion systems.

In *Proc. of the IFAC International Conference on Technology, Automation and Control of Wastewater Systems- TiASWiK'02*, Gdansk - Sobieszewo, Poland, June 19–21, 2002.

[5] M.A. Brdys and B. Ulanicki. *Operational Control of Water Systems: Structures, Algoritms and Applications*. 1994.

[6] K. Deb. An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering*, 186:311–338, 2000.

[7] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, 1989.

[8] M. Haestad, T.M. Walski, D.V. Chase, D.A. Savic, W. Grayman, S. Beckwith, and E. Koelle. *Advanced Water Distribution Modeling and Management*. Haestead Press, Waterbury, CT, 2003.

[9] J.H. Holland. *Adaptation in Natural and Artificial Systems*. MI: University of Michigan Press, Ann Arbor, 1975.

[10] J.M.Maciejowski. *Predictive Control with Constraints*. 2000.

[11] W. Kurek and M.A. Brdys. Genetic solver of optimisation task of mpc for optimising control of integrated quantity and quality in drinking water distribution systems. volume 11. IFAC– PapersOnLine, Elsevier, 2008.

[12] A. Ostfeld and E. Salomons. Optimal layout of early warning detection stations for water distribution systems security. *Journal of Water Resources Planning and Management-Asce*, 130:377–385, 2004.

[13] L.A. Rossman. *EANET 2.0 Users manual*. US Environmental Protection Agency, Cincinnati, 2000.

[14] D.A. Savic and G.A. Walters. Genetic algorithms for least-cost design of water distribution networks. *Journal of Water Resources Planning and Management-Asce*, 123:66–67, 1997.

[15] T.Chang and M. A. Brdys. Performance comparison of chlorine controllers based on input-output and state-space models. In *Proc. of the 1st Annual Environmental & Water Resources Systems Analysis (EWRSA) Symposium, A.S.C.E. Environmental & Water Resources Institute (EWRI) Annual Conference*, Roanoke, Virginia, May 19-22, 2002.

[16] B.A. Tolson, H.R. Maier, A.R. Simpson, and B.J. Lence. Genetic algorithms for reliability-based optimization of water distribution systems. *Journal of Water Resources Planning and Management-Asce*, 130:63–72, 2004.

[17] Z.Y. Wu and A.R. Simpson. Competent genetic-evolutionary optimization of water distribution systems. *Journal of Computing in Civil Engineering*, 15:89–101, 2001.

JAISCR, 2011, Vol.1, No.1, pp. 59–79

ROBUSTIFYING ANALYSIS OF THE DIRECT ADAPTIVE CONTROL OF UNKNOWN MULTIVARIABLE NONLINEAR SYSTEMS BASED ON A NEW NEURO-FUZZY METHOD

Dimitris C. Theodoridis[1], Yiannis S. Boutalis[1,2], Manolis A. Christodoulou[3]

[1]*Department of Electrical and Computer Engineering,*
Democritus University of Thrace, Xanthi, Greece, 67100 GR
e-mail: dtheodo@ee.duth.gr

[2]*Department of Electrical, Electronic and Communication Engineering,*
Chair of Automatic Control, University of Erlangen-Nuremberg, Germany
e-mail: ybout@ee.duth.gr

[3]*Department of Electronic and Computer Engineering,*
Technical University of Crete, Chania, Crete, Greece 73100 GR
e-mail: manolis@ece.tuc.gr

Abstract

In this paper, we are dealing with the problem of directly regulating unknown multivariable affine in the control nonlinear systems and its robustness analysis. The method employs a new Neuro-Fuzzy Dynamical System definition, which uses the concept of Fuzzy Systems (FS) operating in conjunction with High Order Neural Networks. In this way the unknown plant is modeled by a fuzzy - recurrent high order neural network structure (F-RHONN), which is of the known structure considering the neglected nonlinearities. The development is combined with a sensitivity analysis of the closed loop in the presence of modeling imperfections and provides a comprehensive and rigorous analysis showing that our adaptive regulator can guarantee the convergence of states to zero or at least uniform ultimate boundedness of all signals in the closed loop when a not-necessarily-known modeling error is applied. The existence and boundedness of the control signal is always assured by employing a method of parameter "Hopping" and "Modified Hopping", which appears in the weight updating laws. Simulations illustrate the potency of the method showing that by following the proposed procedure one can obtain asymptotic regulation despite the presence of modeling errors. Comparisons are also made to simple recurrent high order neural network (RHONN) controllers, showing that our approach is superior to the case of simple RHONN's.

1 Introduction

High order neural network structures can deal with imprecise data and ill-defined activities. However, subjective phenomena such as reasoning and perceptions are often regarded beyond the domain of conventional neural network theory [21]. It is interesting to note that fuzzy logic is another powerful tool for modeling uncertainties associated with human cognition, thinking and perception. Therefore, it has been established that neural networks and fuzzy inference systems are universal approximators [8, 19, 26],i.e., they can approximate any nonlinear function to any prescribed accuracy pro-

vided that sufficient hidden neurons and training data, which have to be distributed in the full operation space of the plant, or fuzzy rules are available. The neural and fuzzy approaches are most of the time equivalent, differing between each other mainly in the structure of the approximator chosen. Indeed, in order to bridge the gap between the neural and fuzzy approaches several researchers introduce adaptive schemes using a class of parameterized functions that include both neural networks and fuzzy systems [3, 10, 14, 16].

Recently, the combination of these two different technologies has given rise to $fuzzy-neural$ or $neuro-fuzzy$ approaches, that are intended to capture the advantages of both fuzzy logic and neural networks. Numerous works have shown the viability of this approach for system modeling [3, 10, 13, 14, 16]. Algorithms based upon this integration are believed to have considerable potential in the areas of expert systems, medical diagnosis, control systems, pattern recognition and system modeling.

Adaptive control theory has been an active area of research over the past years [2, 5, 7, 6, 9, 11, 17, 18, 20, 25, 27]. The identification procedure is an essential part in any control procedure. In the neuro or $neuro-fuzzy$ adaptive control two main approaches are followed. In the indirect adaptive control schemes, first the dynamics of the system are identified and then a control input is generated according to the certainty equivalence principle. In the direct adaptive control schemes [9, 5, 18] the controller is directly estimated and the control input is generated to guarantee stability without knowledge of the system dynamics. Also, many researchers focus on robust adaptive control that guarantees signal boundness in the presence of modeling errors and bounded disturbances [2, 5, 6, 7, 9, 11, 18, 25, 27].

Recently [4, 12], high order neural network function approximators (HONNF's) have been proposed for the identification of nonlinear dynamical systems of the form $\dot{x} = f(x,u)$, approximated by a Fuzzy Dynamical System. This approximation depends on the fact that fuzzy rules could be identified with the help of HONN's. The same rationale has been employed in [24, 1, 22, 23] where a neuro - fuzzy approach for the indirect and direct control of square unknown systems has been introduced assuming only parameter uncertainty.

In this paper HONN's are also used for the $neuro-fuzzy$ direct control of nonlinear dynamical systems with modeling errors and a robustifying analysis of the method is presented. From the neural network aspect, we have the alternative approximation of weighted indicator functions ensured with the help of multi high order neural networks. From the fuzzy logic aspect, the underlying fuzzy model is of Mamdani-type [15]. The structure identification of the fuzzy system is made offline based either on human expertise or on gathered data. However, the required a-priori information obtained by linguistic information or data is very limited. The only required information is an estimate of the centers of the output fuzzy membership functions. Information on the input variable membership functions and on the underlying fuzzy rules is not necessary because this is automatically estimated by the HONN's. This way the proposed method is less vulnerable to initial design assumptions.

We consider that the unknown system is of an affine in the control multivariable form and propose its approximation by a recurrent structure employing two independent fuzzy subsystems. We also assume the existence of disturbance expressed as modeling error terms depending on both input and system states. Every fuzzy subsystem is approximated from a family of HONN's, each one being related with a group of fuzzy rules. Weight updating laws are given and it is proved that when the structural identification is appropriate and the modeling error terms are within a certain region depending on the input and state values, then the error reaches zero very fast. Also, an appropriate state feedback is constructed to achieve asymptotic regulation of the output, while keeping bounded all signals in the closed loop. A novel technique of weight hopping is also introduced to assure the existence and boundedness of the control signal.

The paper is organized as follows. Section 2 presents notation and preliminaries related to the concept of fuzzy systems (FS) and the terminology used in the remaining paper, while Section 3 demonstrate the $neuro-fuzzy$ representation of the proposed algorithm. The direct neuro fuzzy regulation of affine in the control dynamical systems

under the presence of modeling errors and its robustifying analysis is presented in Section 4, where the method of parameter hopping is explained and the associated weight adaptation laws are given. Finally, simulations presented in Section 5 results on the control of a well known system show off that by following the proposed procedure one can obtain asymptotic regulation in a much better way than by just simply using RHONN controllers. Finally, Section 6 concludes the work.

2 Preliminaries

Consider a nonlinear function $f(x) \in R^n$, $x \in X \subset R^n$ approximately described by a Mamdani-type Fuzzy System (FS). Let $\Omega_{j_1,j_2,...,j_n}^{l_1,l_2,...,l_n}$ be defined as the subset of $x \in X$ belonging to the $(j_1,j_2,...,j_n)^{th}$ input fuzzy patch and pointing - through the vector field $f(\cdot)$ - to the subset which belong to the $l_1,l_2,...,l_n{}^{th}$ output fuzzy patch. In other words, $\Omega_{j_1,j_2,...,j_n}^{l_1,l_2,...,l_n}$ contains input values x that are associated through a fuzzy rule with output values $f(x)$.

Furthermore, the FS receiving as input the $n-tuple$ of $x = (x_1,x_2,\ldots,x_n)$ gives as output an approximate of $f(x)$ using fuzzy rules and a well known fuzzy inference procedure.

Definition 1 *According to the above notation the Rule Firing Indicator Function (RFIF) or simply Indicator Function (IF) connected to* $\Omega_{j_1,j_2,...,j_n}^{l_1,l_2,...,l_n}$ *is defined as follows:+*

$$I_{j_1,j_2,...,j_n}^{l_1,l_2,...,l_n}(x(t)) = \begin{cases} \alpha(x(t)) & if\, x(t) \in \Omega_{j_1,j_2,...,j_n}^{l_1,l_2,...,l_n} \\ 0 & otherwise \end{cases} \quad (1)$$

where $\alpha(x(t))$ *denotes the firing strength of the rule.*

According to the standard fuzzy system description, this strength depends on the membership value of each x_i in the corresponding input membership functions μ_{Fj_i} and more specifically [26], $\alpha(x(t)) = \min[\mu_{Fj_1}(x_1(k)),\ldots\mu_{Fj_n}(x_n(k))]$. Then, assuming a standard defuzzification procedure (e.g. weighted average), the functional representation of the fuzzy system can be written as

$$f(x(t)) = \sum(\bar{x}_f)_{j_1,...,j_n}^{l_1,l_2,...,l_n} \times \left(I'\right)_{j_1,...,j_n}^{l_1,l_2,...,l_n}(x(t)) \quad (2)$$

where the summation is carried out over all the available fuzzy rules. $(\bar{x}_f)_{j_1,...,j_n}^{l_1,l_2,...,l_n}$ is any constant vector consisting of the centers of fuzzy partitions of f determined by $l_1,l_2,...,l_n$ and $(I')_{j_1,...,j_n}^{l_1,l_2,...,l_n}(x(t))$ is the weighted IF (WIF) defined in (1) divided by the sum of all IF participating in the summation of 2.

However, in order the approximation problem to make sense the space $y := x$ must be compact. Thus, our first assumption is the following:

Assumption 1 $\mathrm{y} := \mathrm{x}$ *is a compact set.*

Notice that since $y \subset \Re^n$ the above proposition is identical to the proposition that it is closed and bounded. Also, it is noted that even if y is not compact we may assume that there is a time instant T such that $x(t)$ remains in a compact subset of y for all $t < T$; i.e. if $y_T := \{x(t) \in y, t < T\}$ We may replace proposition 1 by the following proposition

Assumption 2 y_T *is a compact set.*

Basing on the fact that functions of high order neurons are capable of approximating discontinuous functions [4] and [12] we use high order neural networks (HONN's) [22] in order to approximate the WIF. The next lemma [12] states that a HONN can approximate the WIF $(I')_{j_1,...,j_n}^{l_1,...,l_n}$.

Lemma 1 *Consider the indicator function* $(I')_{j_1,...,j_n}^{l_1,...,l_n}$ *and the family of the HONN's* $N(x(t);w,L)$. *Then for any* $\varepsilon > 0$ *there is a vector of weights* $w^{j_1,...,j_n;l_1,...,l_n}$ *and a number of* $L^{j_1,...,j_n;l_1,...,l_n}$ *high order connections such that*

$$\sup_{(x(t))\in\bar{\mathrm{y}}} \{(I')_{j_1,...,j_n}^{l_1,...,l_n}(x(t)) - N(x(t);w^{j_1,...,j_n;l_1,...,l_n}, L^{j_1,...,j_n;l_1,...,l_n})\} < \varepsilon$$

where $\bar{\mathrm{y}} \equiv \mathrm{y}$ *if assumption 1 is valid and* $\bar{\mathrm{y}}_T \equiv \mathrm{y}$ *if assumption 2 is valid.*

Let us now keep $L^{j_1,...,j_n;l_1,...,l_n}$ constant, i.e. let us preselect the number of high order connections, and let us define the optimal weights of the HONN with $L^{j_1,...,j_n;l_1,...,l_n}$ high order connections as follows

$$\bar{w}^{j_1,\ldots,j_n;l_1,\ldots,l_n} := \arg \min_{w \in R^{j_1,\ldots,j_n;l_1,\ldots,l_n}} \times$$
$$\left\{ \sup_{(x(t)) \in \bar{y}} \left| (I')^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x(t)) \right. \right.$$
$$\left. \left. -N(x(t); w, L^{j_1,\ldots,j_n;l_1,\ldots,l_n}) \mid \right\} \right.$$

and the modeling error as follows

$$\nu^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x(t)) = (I')^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x(t))$$
$$-N(x(t); w^{j_1,\ldots,j_n;l_1,\ldots,l_n}, L^{j_1,\ldots,j_n;l_1,\ldots,l_n})$$

It is worth noticing that from Lemma 1, we have that $\sup_{(x(t)) \in \bar{y}} \left| \nu^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x(t)) \right|$ can be made arbitrarily small by simply selecting appropriately the number of high order connections.

Following the above notation $(I')^{l_1,l_2,\ldots,l_n}_{j_1,\ldots,j_n}$ in (2) can be approximated by $N^{l_1,l_2,\ldots,l_n}_{j_1,\ldots,j_n}(x) = N(x(t); w^{j_1,\ldots,j_n;l_1,l_2,\ldots,l_n}, L^{j_1,\ldots,j_n;l_1,l_2,\ldots,l_n})$.

So, Eq. (2) can be rewritten as

$$f(x(t)) = \sum (\bar{x}_f)^{l_1,l_2,\ldots,l_n}_{j_1,\ldots,j_n} \times N^{l_1,l_2,\ldots,l_n}_{j_1,\ldots,j_n}(x(t)) \quad (3)$$

From the above definitions and Eq. (3), it is obvious that the accuracy of the approximation of $f(x)$ depends on the approximation abilities of HONN's and on an initial estimate of the centers of the output membership functions. These centers can be obtained by experts or by off-line techniques based on gathered data. Any other information related to the input membership functions is not necessary because it is replaced by the HONN's.

Figure (1) shows the overall scheme of the proposed $neuro-fuzzy$ approximation of a function $f(x)$ depending on measurements of input variables x and a-priori knowledge of the centers of the partitions of the fuzzy output variables. When x is given as inputs to the $neuro-fuzzy$ network (input layer), the output of indicator layer gives the weighted indicator function outputs which activate the corresponding rules around a fuzzy center (rule layer). The summation of all rules at each sampling time instant gives the overall output of the function $f(x)$ (output layer).

3 Neuro-Fuzzy Representation of the Algorithm

We consider affine in the control, nonlinear dynamical systems of the form

$$\dot{x} = f(x) + g(x) \cdot u \quad (4)$$

where the state $x \in R^n$ is assumed to be completely measured, the control u is in R^q, f is an unknown smooth vector field called the drift term and g is a matrix with rows containing the unknown smooth controlled vector fields g_{ij}. The above class of continuous-time nonlinear systems are called affine, because in (4) the control input appears linear with respect to g. The main reason for considering this class of nonlinear systems is that most of the systems encountered in engineering, are by nature or design, affine.

We are using an affine in the control fuzzy dynamical system, which approximates the system in (4) and uses two fuzzy subsystem blocks for the description of $f(x)$ and $g(x)$ as follows

$$\hat{f}(x) = A\hat{x} + \sum (\bar{x}_f)^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times \left(I'\right)_f{}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x) \quad (5)$$

$$\hat{g}_{ij}(x) = \sum (\bar{x}_{g_{ij}})^{l}_{j_1,\ldots,j_n} \times \left(I'\right)_g{}^{l}_{j_1,\ldots,j_n}(x) \quad (6)$$

where A is a $n \times n$ stable matrix which for simplicity can be taken to be diagonal as $A = diag[-a_1, -a_2, \ldots, -a_n]$, a_i positive, and the summation is carried out over the number of all available fuzzy rules. $(I')_f$, $(I')_g$ are appropriate fuzzy rule indicator functions and the meaning of indices $\bullet^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}$ has already been described in Section 2.

According to Lemma 1, every indicator function can be approximated with the help of a suitable HONN. Therefore, every $(I')_f$, $(I')_g$ can be replaced with a corresponding HONN as follows

$$\hat{f}(x|W_f) = A\hat{x} + \sum (\bar{x}_f)^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times N_f{}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(x) \quad (7)$$

$$\hat{g}_{ij}(x|W_g) = \sum (\bar{x}_{g_{ij}})^{l}_{j_1,\ldots,j_n} \times N_g{}^{l}_{j_1,\ldots,j_n}(x) \quad (8)$$

where W_f, W_g are weights that results from adaptive laws which will be discussed later, and N_f, N_g are appropriate HONN's.

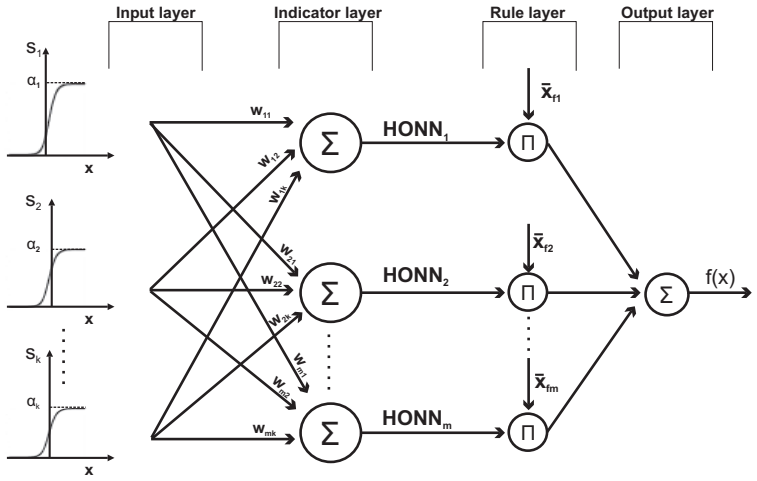


Figure 1. Overall scheme of the proposed $neuro-fuzzy$ representation which approximates function $f(x)$ based on measurements of x and a-priori knowledge of the centers $\bar{x}_f$.

So, the optimal approximation of $f(x)$ and $g(x)$ subfunctions of the dynamical system becomes

$$f(x|W_f^*) = Ax + \sum (\bar{x}_f)_{j_1,\ldots,j_n}^{l_1,\ldots,l_n} \times N_{f\,j_1,\ldots,j_n}^{*l_1,\ldots,l_n}(x) \quad (9)$$

$$g_{ij}(x|W_g^*) = \sum (\bar{x}_{g_{ij}})_{j_1,\ldots,j_n}^{l} \times N_{g\,j_1,\ldots,j_n}^{*l}(x) \quad (10)$$

In order to simplify the model structure, since some rules result in the same output partition, we could replace the NNs associated to the rules having the same output with one NN and therefore the summations in (7),(8) are carried out over the number of the corresponding output partitions. Therefore, the affine in the control fuzzy dynamical system in (5), (6) is replaced by the following equivalent affine Fuzzy - Recurrent High Order Neural Network (F-RHONN), which depends on the centers of the fuzzy output partitions $(\bar{x}_f)_l$ and $(\bar{x}_{g_{ij}})_l$

$$\dot{\hat{x}} = A\hat{x} + \sum_{l=1}^{Npf} (\bar{x}_f)_l \times N_{fl}(x) + \sum_{i=1}^{n} \left(\sum_{j=1}^{q} \left(\sum_{l=1}^{Npg_i} (\bar{x}_{g_{ij}})_l \times N_{g_l}(x) \right) u_j \right) \quad (11)$$

Or in a more compact form

$$\dot{\hat{x}} = A\hat{x} + X_f W_f S_f(x) + X_g W_g S_g(x) u \quad (12)$$

where X_f, X_g are matrices containing the centres of the partitions of every fuzzy output variable of $f(x)$ and $g(x)$, respectively, $S_f(x), S_g(x)$ are matrices containing high order combinations of sigmoid functions of the state x and W_f, W_g are matrices containing respective neural weights according to (11). The dimensions and the contents of all the above matrices are chosen so that $X_f W_f S_f(x)$ is a $n \times 1$ vector and $X_g W_g S_g(x)$ is a $n \times q$ matrix. For notational simplicity we assume that all output fuzzy variables are partitioned to the same number, m, of partitions. Under these specifications X_f is a $n \times n \cdot m$ block diagonal matrix of the form $X_f = diag(X_{f_1}, X_{f_2}, \ldots, X_{f_n})$ with X_{f_i} being an m-dimensional row vector of the form

$$X_{f_i} = \begin{bmatrix} \bar{x}_{f_i}^1 & \bar{x}_{f_i}^2 & \cdots & \bar{x}_{f_i}^m \end{bmatrix}$$

or in a more detailed form

$$X_f = \begin{bmatrix} \bar{x}_{f_1}^1 & \cdots & \bar{x}_{f_1}^m & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \bar{x}_{f_2}^1 & \cdots & \bar{x}_{f_2}^m & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 & \bar{x}_{f_n}^1 & \cdots & \bar{x}_{f_n}^m \end{bmatrix}$$

where $\bar{x}_{f_i}^p$ with $p = 1,2,\ldots,m$, denotes the centre of the p-th partition of f_i. Also, $S_f(x) = \begin{bmatrix} s_1(x) & \ldots & s_k(x) \end{bmatrix}^T$, where each $s_l(x)$ with $l = \{1,2,\ldots,k\}$, is a high order combination of sigmoid functions of the state variables and W_f is a $n \cdot m \times k$ matrix with neural weights. W_f assumes the form $W_f = \begin{bmatrix} W_{f_1} & \cdots & W_{f_n} \end{bmatrix}^T$, where each W_{f_i} is a matrix $\left[w_{f_i}^{pl} \right]_{m \times k}$. X_g is a $n \times n \cdot m \cdot q$ block diagonal matrix of the form $X_g = diag(X_{g_{1j}}, X_{g_{2j}}, \ldots, X_{g_{nq}})$ with each $X_{g_{ij}}$ ($j = 1,2,\ldots,q$, $i = 1,2,\ldots,n$) being an $m-$dimensional raw vector of the form

$$X_{g_{ij}} = \begin{bmatrix} \bar{x}^1_{g_{ij}} & \bar{x}^2_{g_{ij}} & \cdots & \bar{x}^m_{g_{ij}} \end{bmatrix}$$

or in a more detailed form

$$X_g = \begin{bmatrix} X_{g_{11}} & \cdots & X_{g_{1q}} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & X_{g_{21}} & \cdots & X_{g_{2q}} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 & X_{g_{n1}} & \cdots & X_{g_{nq}} \end{bmatrix}$$

where $\bar{x}^k_{g_{ij}}$ denotes the center of the k-th partition of g_{ij}. W_g is a $n \cdot m \cdot q \times n \cdot q$ block diagonal matrix with $W_g = [W_{g_1}, W_{g_2}, \ldots, W_{g_{nq}}]^T$, where each W_{g_k} with $k = 1,2,...,n \cdot q$ is an $m-$dimensional column vector $\left[w^p_{g_k}\right]_{m\times 1}$ of neural weights. Finally, $S_g(x)$ is a $n \cdot q \times q$ matrix of the form $S_g = [S_{g_1}, S_{g_2}, \ldots, S_{g_n}]^T$, where each S_{g_i} is a diagonal $q \times q$ matrix $S_{g_i} = diag(s_i, \ldots, s_i)$ with the diagonal element $s_i(x)$ being a high order combination of sigmoid functions of the state variables.

4 Direct Robust Adaptive Neuro-Fuzzy Control

4.1 Problem Formulation

The state regulation problem is known as our attempt to force the state to zero from an arbitrary initial value by applying appropriate feedback control to the plant input. However, since the plant is considered unknown, we assume that the unknown plant can be described by the following model arriving from the neuro-fuzzy representation of (12), where the weight values W_f^* and W_g^* are unknown.

$$\dot{\hat{x}} = A\hat{x} + X_f W_f^* S_f(x) + X_g W_g^* S_g(x) u \qquad (13)$$

Due to the approximation capabilities of the dynamic neural networks, we can assume with no loss of generality, that the unknown plant (4) can be completely described by a dynamical neural network plus a modeling error term $\omega(x,u)$. In other words, there exist weight values W_f^* and W_g^* such that the system (4) can be written as

$$\dot{\hat{x}} = A\hat{x} + X_f W_f^* S_f(x) + X_g W_g^* S_g(x) u + \omega(x,u) \quad (14)$$

Therefore, the state regulation problem is analyzed for the system (14) instead of (4). Since W_f^* and W_g^* are unknown, our solution consists of designing a control law $u(W_f, W_g, x)$ and appropriate update laws for W_f and W_g to guarantee convergence of the state to zero and in some cases, which will be analyzed in the following sections, boundedness of x and of all signals in the closed loop.

The following mild assumptions are also imposed on (4), to guarantee the existence and uniqueness of solution for any finite initial condition and $u \in U$.

Assumption 3 *Given a class $U \subset R^q$ of admissible inputs, then for any $u \in U$ and any finite initial condition, the state trajectories are uniformly bounded for any finite $T > 0$. Meaning that we do not allow systems processing trajectories which escape at infinite, in finite time T, T being arbitrarily small. Hence, $|x(T)| < \infty$.*

Assumption 4 *The vector fields* f, g_{ij}, $i = 1,2,...,n$ *are continuous with respect to their arguments and satisfy a local Lipchitz condition so that the solution $x(t)$ of (4) is unique for any finite initial condition and $u \in U$.*

4.2 Adaptive Regulation - Complete Matching

In this subsection, we present a solution to the adaptive regulation problem and investigate the modeling error effects. Assuming the presence of modeling error the unknown system can be written as (14), where $x \in \Re^n$ is the system state vector, $u \in \Re^q$ are the control inputs, X_f, X_g are $n \times n \cdot m$ and $n \times n \cdot m \cdot q$ block diagonal matrices, respectively, W_f^* is a $n \cdot m \times k$ matrix of synaptic weights and W_g^* is a $n \cdot m \cdot q \times n \cdot q$ block diagonal matrix. Finally, $S_f(x)$ is a k-dimensional vector and $S_g(x)$ is a $n \cdot q \times n$ block diagonal matrix with each diagonal element $s_i(x)$ being a high order combination of sigmoid functions of the state variables.

Define now ν as

$$\nu \triangleq X_f W_f S_f(x) + X_g W_g S_g(x) u - \dot{x} - Ax \qquad (15)$$

substituting Eq. (14) to Eq. (15) we have

$$\nu \triangleq X_f \tilde{W}_f S_f(x) + X_g \tilde{W}_g S_g(x) u - \omega(x,u) \qquad (16)$$

where $\tilde{W}_f = W_f - W_f^*$ and $\tilde{W}_g = W_g - W_g^*$. W_f and W_g are estimates of W_f^* and W_g^*, respectively and are obtained by update laws which are to be designed in the sequel. ν cannot be measured since $\dot{x}$ is unknown. To overcome this problem, we use the following filtered version of ν

$$\nu = \dot{\xi} + K\xi$$

where $K = \begin{bmatrix} k_1 & 0 & \cdots & 0 \\ 0 & k_2 & 0 & \cdots \\ \cdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & k_n \end{bmatrix}$ is a diagonal positive definite matrix. In the sequel, according to Eq. (15) we have that

$$\dot{\xi} + K\xi = -\dot{x} - Ax + X_f W_f S_f(x) + X_g W_g S_g(x)u \tag{17}$$

and after substituting Eq. (14) we have

$$\dot{\xi} = -K\xi + X_f \tilde{W}_f S_f(x) + X_g \tilde{W}_g S_g(x)u - \omega(x,u) \tag{18}$$

To implement Eq. (18), we take

$$\xi \triangleq \zeta - x \tag{19}$$

Employing Eq. (19), Eq. (17) can be written as

$$\dot{\zeta} + K\zeta = Kx - Ax + X_f W_f S_f(x) + X_g W_g S_g(x)u \tag{20}$$

with state $\zeta \in \Re^n$. This method is referred to as error filtering.

The regulation of the system can be achieved by selecting the control input to be

$$u = -\left[X_g W_g S_g(x)\right]^+ \left[X_f W_f S_f(x) + \upsilon\right] \tag{21}$$

where $[\cdot]^+$ means pseudo-inverse in Moore-Penrose sense and

$$\upsilon = (K - A)x \tag{22}$$

Thus, substituting Eq. (21), Eq. (20) becomes

$$\dot{\zeta} = -K\zeta \tag{23}$$

To continue, consider the Lyapunov candidate function

$$V = \xi^T\xi + \zeta^T\zeta + \frac{1}{2\gamma_1} tr\left\{\tilde{W}_f^T \tilde{W}_f\right\} + \frac{1}{2\gamma_2} tr\left\{\tilde{W}_g^T \tilde{W}_g\right\} \tag{24}$$

If we take the derivative of Eq. (24) with respect to time we obtain

$$\dot{V} = -\xi^T K\xi - \zeta^T K\zeta + \xi^T X_f \tilde{W}_f S_f(x) + \xi^T X_g \tilde{W}_g S_g(x)u - \xi^T \omega(x,u) + \frac{1}{\gamma_1} tr\left\{\dot{W}_f^T \tilde{W}_f\right\} + \frac{1}{\gamma_2} tr\left\{\dot{W}_g^T \tilde{W}_g\right\}$$

Hence, if we choose

$$tr\left\{\dot{W}_f^T \tilde{W}_f\right\} = -\gamma_1 \xi^T X_f \tilde{W}_f S_f(x) \tag{25}$$

$$tr\left\{\dot{W}_g^T \tilde{W}_g\right\} = -\gamma_2 \xi^T X_g \tilde{W}_g S_g(x)u \tag{26}$$

$\dot{V}$ becomes

$$\dot{V} \leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 + \|\xi\|\,\|\omega(x,u)\| \tag{27}$$

It can be easily verified that Eqs. (25) and (26) after making the appropriate operations, can be element wise written as

a) for the elements of W_f

$$\dot{w}_{f_i}^{pl} = -\gamma_1 \bar{x}_{f_i}^p \xi s_l(x) \tag{28}$$

or equivalently $\dot{W}_{f_i}^l = -\gamma_1 \left(X_{f_i}\right)^T \xi s_l(x)$ for all $i = 1,2,...,n$, $p = 1,2,...,m$ and $l = 1,2,...,k$.

b) for the elements of W_g

$$\dot{w}_{g_{ij}}^p = -\gamma_2 \bar{x}_{g_{ij}}^p \xi u_j s_i(x) \tag{29}$$

or equivalently $\dot{W}_{g_{ij}} = -\gamma_2 \left(X_{g_{ij}}\right)^T \xi u_j s_i(x)$ for all $i = 1,2,...,n$, $j = 1,2,...,q$ and $p = 1,2,...,m$.

Equations (28) and (29) can be finally written in a compact form as

$$\dot{W}_f = -\gamma_1 X_f^T \xi S_f^T(x) \tag{30}$$

$$\dot{W}_g = -\gamma_2 X_g^T \xi u^T S_g^T(x) \tag{31}$$

where ξ is a vector $\xi = (\xi_1, \xi_2, ..., \xi_n)$ and u is also a vector $u = (u_1, u_2, ..., u_n)$. Furthermore, we cannot conclude anything about the weight convergence if the existence and boundeness of signal u are not assured. So, the weight updating laws (28), (29) have to be modified by introducing a method of parameter "Hopping" or "Modified Hopping", which is explained below.

4.2.1 Introduction to the Parameter Hopping

The weight updating laws presented previously in Section 4.2 are valid when the control law signal in (21), (22) exists. Therefore, the existence of $[X_g W_g S_g(x)]^+$ has to be assured. Since the sub matrices of $S_g(x)$ are diagonal with the diagonal elements $s_i(x) \neq 0$ and X_g, W_g are block diagonal and

thus linearly independent, the existence of the pseudoinverse is assured when $X_{g_{i,j+q(i-1)}} \cdot W_{g_{j+q(i-1)}} \neq 0$, $\forall i = 1,\ldots,n$ and $\forall j = 1,\ldots,q$. Therefore, $W_{g_{j+q(i-1)}}$ has to be confined such that $\left| X_{g_{i,j+q(i-1)}} \cdot W_{g_{j+q(i-1)}} \right| \geq \theta_{j+q(i-1)} > 0$, with $\theta_{j+q(i-1)}$ being a small positive design parameter (usually in the range of $[0.001, 0.01]$). In case the boundary defined by the above confinement is nonlinear the updating W_g can be modified by using a projection algorithm [9]. For notational simplicity, we can define with $a = i, j+q(i-1)$ and $b = j+q(i-1)$. However, in our case the boundary surface is linear and the direction of updating is normal for it because $\nabla [X_{g_a} \cdot W_{g_b}] = X_{g_a}$. Therefore, the projection of the updating vector on the boundary surface is of no use. Instead, using concepts from multidimensional vector geometry we modify the updating law such that, when the weight vector approaches (within a safe distance θ_b) the forbidden hyperplane $X_{g_a} \cdot W_{g_b} = 0$ and the direction of updating is toward the forbidden hyper-plane, it introduces a *hopping* which drives the weights in the direction of the updating but on the other side of the space, where here the weight space is divided into two sides by the forbidden hyper-plane. For example, let the weight updating hopping occurs at the t_h time instant. Then, if the weights at t_h^- time instant lies in the space determined by $X_{g_a} \cdot W_{g_b} < -\theta$ then, after performing hopping the weights move into the space determined by $X_{g_a} \cdot W_{g_b} > \theta$ and from t_h^+ on they continue their updating direction. This procedure is depicted in Fig. 2, where a simplified 2-dimensional representation is given. Theorem 2 below introduces this *hopping* in the weight updating law.

Lemma 2 *The updating law for the elements of $W_{g_{ij}}$ given by (29) and modified according to the Hopping method:*

$$\dot{W}_{g_b} = \begin{cases} -\gamma_2 (X_{g_a}) \xi u_j s_i(x) & if\ |X_{g_a} \cdot W_{g_b}| > \theta_b \\ & or\ X_{g_a} \cdot W_{g_b} = \pm\theta_b \\ & and\ X_{g_a} \cdot \dot{W}_{g_b} <> 0 \\ -\gamma_2 (X_{g_a}) \xi u_j s_i(x) & otherwise \\ -\frac{2\kappa^{inner}\left(X_{g_a} W_{g_b} (X_{g_a})^T\right)}{tr\{(X_{g_a})^T X_{g_a}\}} & \end{cases}$$

assures the existence of the control signal.

Proof The first part of the weight updating equation is used when the weights are at a certain distance (condition if $|X_{g_a} \cdot W_{g_b}| > \theta_b$) from the forbidden plane or at the safe limit (condition $|X_{g_a} \cdot W_{g_b}| = \pm\theta_b$) but with the direction of updating moving the weights far from the forbidden plane (condition $X_{g_a} \cdot \dot{W}_{g_b} <> 0$). In the current notation, the "$\pm$" symbol has a one to one correspondence with the "$<>$" one, meaning that "$+$" case corresponds to "$<$" case and the "$-$" case corresponds to "$>$" case.

In the second part of $\dot{W}_{g_b}$, term $-\frac{2\kappa^{inner}\left(X_{g_a} W_{g_b} (X_{g_a})^T\right)}{tr\{(X_{g_a})^T X_{g_a}\}}$ determines the magnitude of weight *hopping*, which as explained in the vectorial proof of "hopping" [22], has to be at least two times the distance of the current weight vector to the forbidden hyper-plane. In addition, the constant value κ^{inner} helps the weights to move nearby but outside the forbidden hyper planes in order to avoid the infinite hopping. Therefore, the ***existence*** of the control signal is assured because the weights never reach the forbidden plane. □

The inclusion of weight hopping in the weights updating law guarantees that the control signal does not go to infinity. Apart from that, it is also of practical use to assure that $X_g W_g S_g(x)$ does not approach even temporarily at very large values because in this case the method may become algorithmically unstable driving at the same time the control signal to zero failing to control the system. To assure that this situation does not happen we have again to assure that $|X_{g_a} \cdot W_{g_b}| < \rho_b$ with ρ_b being again a design parameter determining an external limit for $X_{g_a} \cdot W_{g_b}$. Following the same lines of thought with the case of weight hopping introduced above we could again consider the forbidden hyperplanes being defined by the equation $|X_{g_a} \cdot W_{g_b}| = \rho_b$. When the weight vector reaches one of the forbidden hyper-planes $X_{g_a} \cdot W_{g_b} = \rho_b$ and the direction of updating is toward the forbidden hyper-plane, a new *modified hopping* is introduced which moves the weights insight the restricting area. This procedure is depicted in Fig. 4, in a simplified 2-dimensional representation. The magnitude of hopping is $-\frac{\kappa^{outer}\left(X_{g_a} W_{g_b} (X_{g_a})^T\right)}{tr\{(X_{g_a})^T X_{g_a}\}}$ being determined by following again the same vectorial proof [22], with κ^{outer} a small positive number decided appropriately from the designer as will be ex-

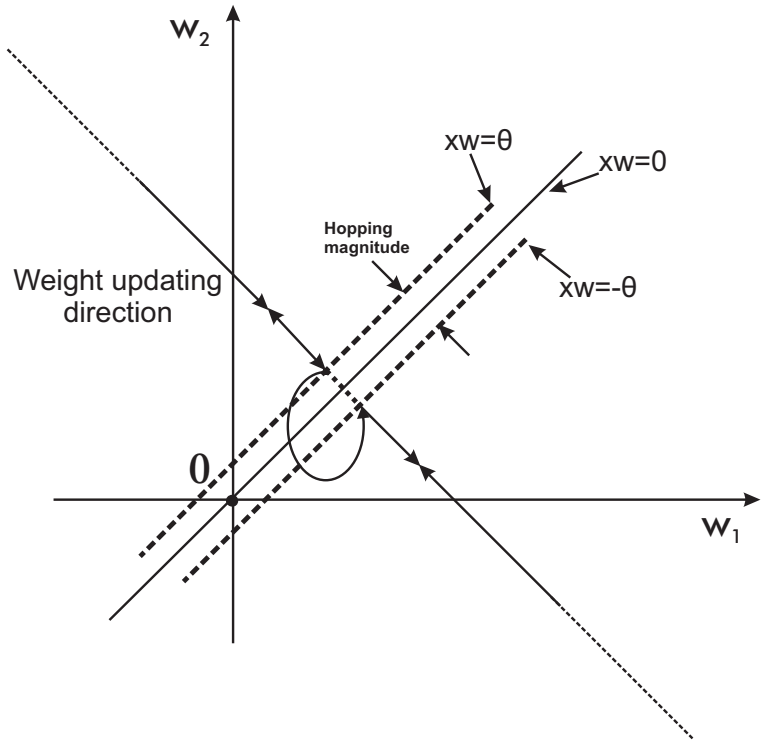


Figure 2. Pictorial Representation of parameter hopping

plained further down.

The adaptation of the weights is perpendicular to the forbidden hyperplanes. This is demonstrated using the first derivative of $X_{g_a} \cdot W_{g_b}$ in respect to the elements of W_{g_b} which is actually equal to the vector X_{g_a} of the fuzzy output centers. When the weights leave the admissible area then the hopping condition is activated and the weights come back to the permissible area as can be seen in Fig. 3. The positive constant values κ^{inner}, κ^{outer} help the designer to avoid the infinite hopping that may occur between the forbidden hyperplanes.

By performing *hopping* when $X_{g_a} \cdot W_{g_b}$ reaches either the inner or outer forbidden planes, $X_{g_a} \cdot W_{g_b}$ is confined to lie in space $P = \{X_{g_a} \cdot W_{g_b} : |X_{g_a} \cdot W_{g_b}| \leq \rho_b \; and \; |X_{g_a} \cdot W_{g_b}| > \theta_b\}$ lying between these hyper-planes. The weight updating law for W_{g_b} incorporating the two hopping conditions can now be expressed as

$$\dot{W}_{g_b} = \begin{cases} -\gamma_2 (X_{g_a}) \xi u_j s_i(x) & if\, X_{g_a} \cdot W_{g_b} \in P \\ & or X_{g_a} \cdot W_{g_b} = (\pm\theta_b \; or \pm \\ & and\, X_{g_a} \cdot \dot{W}_{g_b} <> or \\ & >< 0 \\ -\gamma_2 (X_{g_a}) \xi u_j s_i(x) - & \\ -\frac{2\sigma_i \kappa^{inner} \left(X_{ga} W_{g_b} (X_{ga})^T\right)}{tr\{(X_{ga})^T X_{ga}\}} & otherwise \\ -\frac{(1-\sigma_i) \kappa^{outer} \left(X_{ga} W_{g_b} (X_{ga})^T\right)}{tr\{(X_{ga})^T X_{ga}\}} & \end{cases} \tag{32}$$

where

$$\sigma_i = \begin{cases} 0 & if\, X_{g_a} \cdot W_{g_b} = \pm\rho_l \\ & and\, X_{g_a} \cdot \dot{W}_{g_b} <> 0 \\ 1 & otherwise \end{cases} \tag{33}$$

where again, the " $\pm$ " symbol has a one to one correspondence with the " $<>$ " one, meaning that " $+$ " case corresponds to " $<$ " case and the " $-$ " case corresponds to " $>$ " case.

At this point we can distinguish two possible cases. The complete model matching at zero case and the modeling error at zero case.

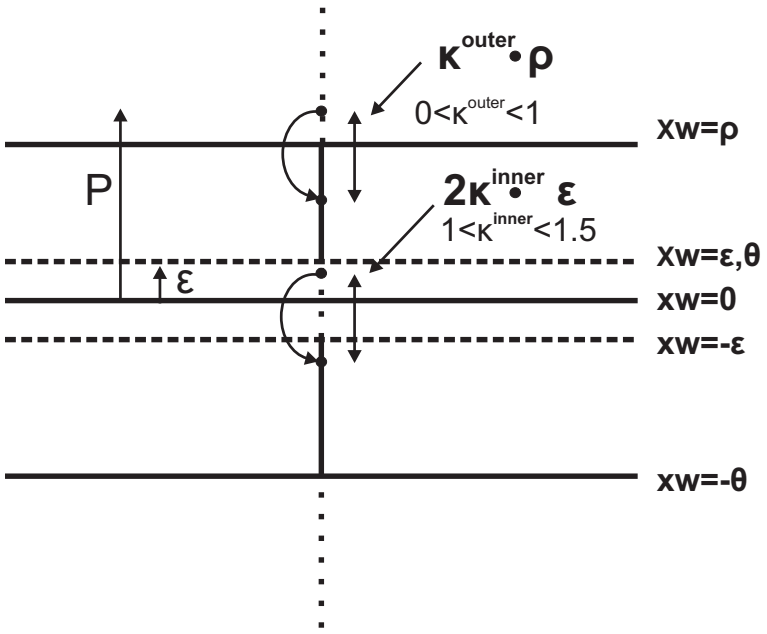


Figure 3. Inner and outer hopping at a distance which depends on an appropriate selection of κ^{inner}, κ^{outer} constant values.

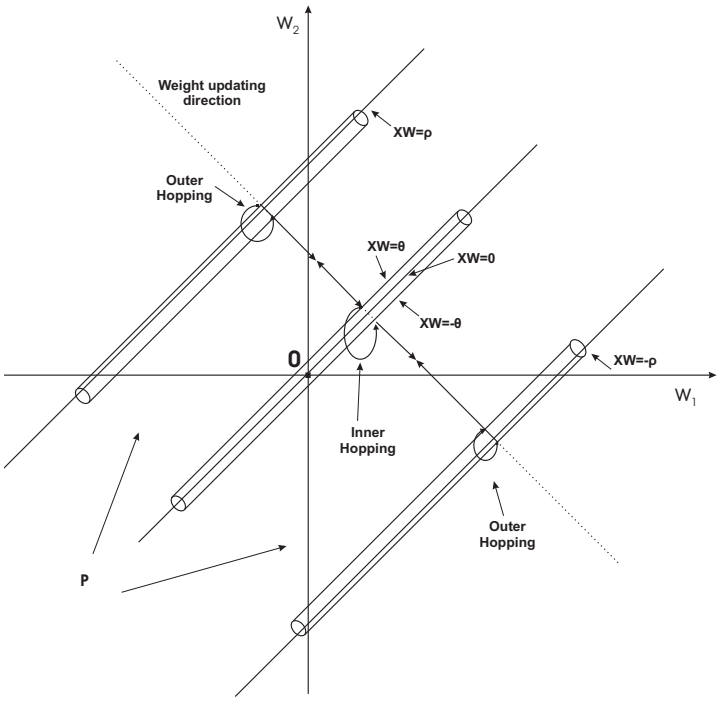


Figure 4. Pictorial Representation of inner and outer parameter hopping

4.2.2 The Complete Model Matching at Zero Case

We make the following assumption.

Assumption 5 The modeling error term satisfies

$$\|\omega(x,u)\| \leq \ell_1' \|x\| + \ell_1'' \|u\|$$

where ℓ_1' and ℓ_1'' are known positive constants.

Also, we can find an *a priori* known constant $\ell_u > 0$, such that

$$\|u\| \leq \ell_u \|x\| \tag{34}$$

and assumption 5 becomes equivalent to

$$\|\omega(x)\| \leq \ell_1 \|x\| \tag{35}$$

where

$$\ell_1 = \ell_1' + \ell_1'' \ell_u \tag{36}$$

is a positive constant.

One can easily verify that (34) is valid provided that $X_f W_f$ is uniformly bounded by a known positive constant ε_i so $X_f W_f(t)$ is confined to the set $P_2 = \{X_{f_i} \cdot W_{f_i}^l : \left| X_{f_i} \cdot W_{f_i}^l \right| \leq \varepsilon_i\}$ through the use of a hopping algorithm. In particular, the standard update law (26) is modified to

$$\dot{W}_{f_i}^l = \begin{cases} -\gamma_1 (X_{f_i})^T \xi s_l(x) & \begin{array}{l} if\, X_{f_i} \cdot W_{f_i}^l \in P_2 \\ or\, X_{f_i} \cdot W_{f_i}^l = \pm\varepsilon_i \\ and\, X_{f_i} \cdot \dot{W}_{f_i}^l >< 0 \end{array} \\ -\gamma_1 (X_{f_i})^T \xi s_l(x) - & \\ -\dfrac{\kappa^{outer}\left(X_{f_i} W_{f_i}^l (X_{f_i})^T\right)}{tr\left\{(X_{f_i})^T X_{f_i}\right\}} & otherwise \end{cases} \tag{37}$$

therefore, we have the following lemma.

Lemma 3 If the initial weights are chosen such that $X_{f_i} \cdot W_{f_i}^l(0) \in P_2$ and $X_{f_i} \cdot W_{f_i}^{*l} \in P_2$ then we have $X_{f_i} \cdot W_{f_i}^l \in P_2$ for all $t \geq 0$.

Proof The above lemma can be readily established by noting that whenever $|X_{f_i} \cdot (W_{f_i}^l)^-| \geq \varepsilon_i$ then

$$\frac{d}{dt}\left(|X_{f_i} \cdot (W_{f_i}^l)^+|^2\right) \leq 0 \tag{38}$$

which implies that after hopping occurs, the weights $(W_{f_i}^l)^+$, are directed towards the interior of P_2. For simplicity, since we will be working from now on with the time $(\cdot)^+$, we omit the + sign from the exponent. It is true that

$$\frac{d}{dt}\left(|X_{f_i} \cdot W_{f_i}^l|^2\right) = {W_{f_i}^l}^T \dot{W}_{f_i}^l X_{f_i} {X_{f_i}}^T \tag{39}$$

Since $X_{f_i} {X_{f_i}}^T > 0$, only ${W_{f_i}^l}^T \dot{W}_{f_i}^l$ determines the sign of the above derivative.

Employing the modified adaptive law (37), we obtain

$$\begin{aligned} \left(W_{f_i}^l\right)^T \dot{W}_{f_i}^l = \; & -\gamma_1 \left(W_{f_i}^l\right)^T (X_{f_i})^T \xi s_l(x) \\ & -\kappa^{outer} \varepsilon_i \frac{\left(W_{f_i}^l\right)^T W_{f_i}^l}{\left\|W_{f_i}^l\right\|} \end{aligned} \tag{40}$$

where $\varepsilon_i = \dfrac{\left(X_{f_i} W_{f_i}^l (X_{f_i})^T\right)}{tr\left\{(X_{f_i})^T X_{f_i}\right\}}$. As concerning the second part of the above equation it is obvious that $\varepsilon_i > 0$ and $\dfrac{\left(W_{f_i}^l\right)^T W_{f_i}^l}{\left\|W_{f_i}^l\right\|} > 0$. So,

$$-\kappa^{outer} \varepsilon_i \frac{\left(W_{f_i}^l\right)^T W_{f_i}^l}{\left\|W_{f_i}^l\right\|} < 0.$$

Now, regarding the first part of eq. (40), we can distinguish two cases:

Case 1: $X_{f_i} \cdot W_{f_i}^l = \varepsilon_i$ and $X_{f_i} \cdot \dot{W}_{f_i}^l < 0$.

From the above notation we have that

$$X_{f_i} \dot{W}_{f_i}^l = -\gamma_1 X_{f_i} (X_{f_i})^T \xi s_l(x) < 0 \Rightarrow \gamma_1 \xi s_l(x) > 0 \tag{41}$$

also, $X_{f_i} W_{f_i}^l \geq \varepsilon_i$ and so the first part of Eq. (40) becomes

$$-\gamma_1 \left(W_{f_i}^l\right)^T (X_{f_i})^T \xi s_l(x) \leq -\gamma_1 \varepsilon_i \xi s_l(x)$$

according to Eq. (41)

$$-\gamma_1 \left(W_{f_i}^l\right)^T (X_{f_i})^T \xi s_l(x) < 0$$

Case 2: $X_{f_i} \cdot W_{f_i}^l \leq -\varepsilon_i$ and $X_{f_i} \cdot \dot{W}_{f_i}^l > 0$.

From the above notation we have that

$$X_{f_i}\dot{W}_{f_i}^{l} = -\gamma_1 X_{f_i}\left(X_{f_i}\right)^T \xi s_l(x) > 0 \Rightarrow \gamma_1 \xi s_l(x) < 0 \tag{42}$$

also, $X_{f_i} W_{f_i}^{l} \leq -\varepsilon_i$ and so the first part of Eq. (40) becomes

$$-\gamma_1 \left(W_{f_i}^{l}\right)^T \left(X_{f_i}\right)^T \xi s_l(x) \geq -\gamma_1(-\varepsilon_i)\xi s_l(x)$$

according to Eq. (42)

$$-\gamma_1 \left(W_{f_i}^{l}\right)^T \left(X_{f_i}\right)^T \xi s_l(x) < 0$$

therefore, we finally obtain

$$\frac{d}{dt}\left(\left|X_{f_i}\cdot W_{f_i}^{l}(t)\right|^2\right) \leq 0$$

□

In the sequel, employing assumption 5, Eq. (27) becomes

$$\dot{V} \leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 + \ell_1\|\xi\|\|x\| \Rightarrow$$

$$\dot{V} \leq -\lambda_{min}(K)\left(\|\xi\|^2 + \|\zeta\|^2\right) + \ell_1\|\xi\|^2 + \ell_1\|\xi\|\|\zeta\| \Rightarrow$$

$$\dot{V} \leq -\begin{bmatrix}\|\xi\| & \|\zeta\|\end{bmatrix}\begin{bmatrix}\lambda_{min}(K)-\ell_1 & -\ell_1 \\ 0 & \lambda_{min}(K)\end{bmatrix}\begin{bmatrix}\|\xi\| \\ \|\zeta\|\end{bmatrix} \tag{43}$$

Hence, if we chose $\lambda_{min}(K) \geq \ell_1$ then Eq. (43) becomes negative. Thus, we have

$$\dot{V} \leq 0. \tag{44}$$

Regarding the ***negativity*** of $\dot{V}$ we proceed with the following lemma.

Lemma 4 Based on the adaptive laws (32), (37) the additional terms introduced in the expression for $\dot{V}$, can only make $\dot{V}$ more negative.

Proof Let that $W_{g_b}^{*}$ contains the actual unknown values of W_{g_b} such that $\left|X_{g_a}\cdot W_{g_b}^{*}\right| >> \theta_b$ and that $\tilde{W}_{g_b} = W_{g_b} - W_{g_b}^{*}$. Then, the weight hopping can be equivalently written with respect to $\tilde{W}_{g_b}$ as $-2\kappa^{inner}\theta_b\tilde{W}_{g_b}/\|\tilde{W}_{g_b}\|$ when the inner hopping condition is activated or $-\kappa^{outer}\rho_b\tilde{W}_{g_b}/\|\tilde{W}_{g_b}\|$ when the outer hopping condition is activated. Under this consideration the modified updating law is rewritten as $\dot{W}_{g_b} = -\gamma_2\left({}^1X^i\right)^T\xi u_j s_i(x) - 2\sigma_i\kappa^{inner}\theta_b\tilde{W}_{g_b}/\|\tilde{W}_{g_b}\| - (1-\sigma_i)\kappa^{outer}\rho_b\tilde{W}_{g_b}/\|\tilde{W}_{g_b}\|$. With this updating law it can be easily verified that (43) becomes

$$\dot{V} \leq -\begin{bmatrix}\|\xi\| & \|\zeta\| & \Theta_g\end{bmatrix}\begin{bmatrix}\lambda_{min}(K)-\ell_1 & -\ell_1 & \\ 0 & \lambda_{min}(K) & \\ & & -\Theta_g\end{bmatrix}\begin{bmatrix}\|\xi\| \\ \|\zeta\| \\ \Theta_g\end{bmatrix} \tag{45}$$

with Θ_g being a positive constant expressed as

$$\Theta_g = \sigma_i\sum 2\kappa^{inner}\theta_b\left((\tilde{W}_{g_b})^T)\tilde{W}_{g_b})\right)/\|\tilde{W}_{g_b}\| + (1-\sigma_i)\kappa^{outer}\sum\rho_b\left((\tilde{W}_{g_b})^T)\tilde{W}_{g_b})\right)/\|\tilde{W}_{g_b}\| \geq 0$$

for all time, where the summation includes all weight vectors which require hopping.

Therefore, the negativity of $\dot{V}$ is actually strengthened due to the last negative term.

By using the modified updating law for $W_{f_i}^{l}$ the negativity of the Lyapunov function is not compromised. Indeed, the first part of the modified form of $\dot{W}_{f_i}^{l}$ shown in Eq. (37), is exactly the same with (26) and therefore according to the development of (26) the negativity of V is in effect. The first part is used when the weights are inside the constraint area (condition if $\left|X_{f_i}\cdot W_{f_i}^{l}\right| \leq \varepsilon_i$) or at the safe limit (condition $X_{f_i}\cdot W_{f_i}^{l} = \pm\varepsilon_i$) but with the direction of updating moving the weights towards the 'safe' region (condition $X_{f_i}\cdot\dot{W}_{f_i}^{l} >< 0$).

In the second part of $W_{f_i}^{l}$, term $-\frac{\kappa^{outer}\left(X_{f_i}W_{f_i}^{l}(X_{f_i})^T\right)}{tr\{(X_{f_i})^T X_{f_i}\}}$ determines the magnitude of weight *hopping*, which as explained in the vectorial proof of "hopping" [22], has to be two κ^{outer} times the distance of the current weight vector. Regarding the ***negativity*** of $\dot{V}$ we proceed as follows.

Let that $W_{f_i}^{*l}$ contains the actual unknown values of $W_{f_i}^{l}$ such that $\left|X_{f_i}\cdot W_{f_i}^{*l}\right| << \varepsilon_i$ and that $\tilde{W}_{f_i}^{l} = W_{f_i}^{l} - W_{f_i}^{*l}$. Then, the weight hopping can be equivalently written with respect to $\tilde{W}_{f_i}^{l}$ as $-\kappa^{outer}\varepsilon_i\tilde{W}_{f_i}^{l}/\|\tilde{W}_{f_i}^{l}\|$. Under this consideration the modified updating law is rewritten as $\dot{W}_{f_i}^{l} = -\gamma_1\left(X^i\right)^T\xi s_l(x) - \kappa^{outer}\varepsilon_i\tilde{W}_{f_i}^{l}/\|\tilde{W}_{f_i}^{l}\|$. With this updating law it can be easily verified that eq. (45) be-

comes

$$\dot{V} \leq -\left[\|\xi\| \quad \|\zeta\|\right] \begin{bmatrix} \lambda_{min}(K)-\ell_1 & -\ell_1 \\ 0 & \lambda_{min}(K) \end{bmatrix} \begin{bmatrix} \|\xi\| \\ \|\zeta\| \end{bmatrix} - \Theta_f - \Theta_g \quad (46)$$

with Θ_f being a positive constant expressed as $\Theta_f = \sum \kappa^{outer} \varepsilon_i \left((\tilde{W}_{f_i}^l)^T) \tilde{W}_{f_i}^l) \right) / \|\tilde{W}_{f_i}^l\| \geq 0$ for all time, where the summation includes all weight vectors which require hopping. Therefore, the negativity of $\dot{V}$ is actually strengthened due to the last negative terms.

Lemma 3 imply that the hopping modifications (32), (37) guarantees boundedness of the weights, without affecting the rest of the stability properties established in the absence of hopping. □

Hence, we can prove the following theorem.

Theorem 5 The control law (21) and (22) together with the updating laws (32) and (37) guarantee the following properties

1. $\xi, \|x\|, W_f, W_g, \zeta, \dot{\xi} \in L_\infty, \quad \|\xi\| \in L_2$
2. $\lim_{t\to\infty} \xi(t) = 0, \quad \lim_{t\to\infty} \|x(t)\| = 0$
3. $\lim_{t\to\infty} \dot{W}_f(t) = 0, \quad \lim_{t\to\infty} \dot{W}_g(t) = 0$

provided that $\lambda_{min}(K) \geq \ell_1$.

Proof From Eq. (44) we have that $V \in L_\infty$ which implies $\xi, \tilde{W}_f, \tilde{W}_g \in L_\infty$. Furthermore $W_f = \tilde{W}_f + W_f^* \in L_\infty$ and $W_g = \tilde{W}_g + W_g^* \in L_\infty$. Since, $\xi = \zeta - x$ and $\zeta, \xi \in L_\infty$ this in turn implies that $\|x\| \in L_\infty$. Moreover, since V is a monotone decreasing function of time and bounded from below, $\lim_{t\to\infty} V(t) = V_\infty$ exists so by integrating $\dot{V}$ from 0 to ∞ we have

$$(\lambda_{\min}(K) - l_1) \int_0^\infty \|\xi\|^2 dt + \lambda_{\min}(K) \int_0^\infty \|\zeta\|^2 dt - \ell_1 \int_0^\infty \|\xi\| \|\zeta\| dt = |V(0) - V_\infty| < \infty$$

which implies that $\|\xi\| \in L_2$. We also have that

$$\dot{\xi} = -K\xi + X_f \tilde{W}_f S_f(x) + X_g \tilde{W}_g S_g(x) u - \omega(x).$$

Hence and since $u, \|x\| \in L_\infty$ $\dot{\xi} \in L_\infty$, the sigmoidals are bounded by definition, $\tilde{W}_f, \tilde{W}_g \in L_\infty$ and Assumption 5 hold, so since $\xi \in L_2 \cap L_\infty$ and $\dot{\xi} \in L_\infty$, applying Barbalat's Lemma [9] we conclude that $\lim_{t\to\infty} \xi(t) = 0$. Now, using the boundedness of $u, S_f(x), S_g(x), x$ and the convergence of $\xi(t)$ to zero, we have that $\dot{W}_f, \dot{W}_g$ also converge to zero. Hence and since $\zeta(t)$ also converges to zero, we have that

$$\lim_{t\to\infty} x(t) = \lim_{t\to\infty} \zeta(t) - \lim_{t\to\infty} \xi(t) = 0$$

Thus,

$$\lim_{t\to\infty} \|x(t)\| = 0.$$

□

Remark 1 *Inequality $\lambda_{min}(K) \geq \ell_1$ shows how the design constant K should be selected, in order to guarantee convergence of the state x to zero, even in the presence of modeling error terms which are not uniformly bounded* a priori*, as assumption 5 implies. The value of K becomes large as we allow for large model imperfections but K is implemented as a gain in the construction of $\dot{\zeta}$ and for practical reasons it cannot take arbitrarily large values. This leads to a compromise between the value of K and the maximum allowable modeling error terms.*

4.2.3 The Modeling Error at Zero Case

In the subsection (4.2.2), we have assumed that the modeling error term satisfies the following condition

$$\|\omega(x,u)\| \leq l_1' \|x\| + l_1'' \|u\|$$

which implies that the modeling error becomes zero when $\|x\| = 0$ and we have proven convergence of the state x to zero, plus boundedness of all signals in the closed-loop. In this subsection however, we examine the more general case which is described by the following assumption.

Assumption 6 The modeling error term satisfies

$$\|\omega(x,u)\| \leq l_0 + l_1' \|x\| + l_1'' \|u\|$$

Having made this assumption, we now allow a not-necessarily-known modeling error $l_0 \neq 0$ at zero. Furthermore, as stated previously, we can find an *a priori* known constant $l_u > 0$, such that

$$\|u\| \leq l_u \|x\|$$

thus making

$$\|\omega(x,u)\| \equiv \|\omega(x)\|$$

and Assumption 6 equivalent to

$$\|\omega(x)\| \leq l_0 + l_1 \|x\| \tag{47}$$

where

$$l_1 = l_1' + l_1'' l_u \tag{48}$$

is a positive constant. Employing (35), Eq. (27) becomes

$$\begin{aligned} \dot{V} &\leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 \\ &\quad + \|\xi\|[l_0 + l_1\|x\|] \\ &\leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 \\ &\quad + l_1\|\xi\|^2 + l_1\|\xi\|\|\zeta\| + l_0\|\xi\|. \end{aligned} \tag{49}$$

To continue, we need to state and prove the following lemma

Lemma 6 The control law

$$u = -[X_g W_g S_g(x)]^+ [X_f W_f S_f(x) + \upsilon] \tag{50}$$

$$\upsilon = (K - A)x \tag{51}$$

where the synaptic weight estimates W_f and W_g, are adjusted according to equations (32), (37) guarantee the following properties

1. $\zeta(t) \leq 0, \quad \forall t \geq 0$

2. $\lim_{t\to\infty} \zeta(t) = 0$ exponentially fast provided that $\zeta(t) < 0$.

Proof Observe that if we use the control laws (50), (51), Eq. (20) becomes

$$\dot{\zeta} = -K\zeta, \quad \forall t \geq 0$$

which is a homogeneous differential equation with solution

$$\zeta(t) = \zeta(0)e^{-Kt}$$

Hence, if $\zeta(0)$ which represents the initial value of $\zeta(t)$, is chosen negative, we obtain

$$\zeta(t) \leq 0 \quad \forall t \geq 0.$$

Moreover, $\zeta(t)$ converges to zero exponentially fast. □

Hence, we can distinguish the following cases:

Case 1: If $x \geq 0$ we have that $\zeta(t) \geq \xi(t)$ but $\zeta(t) \leq 0, \forall t \geq 0$ which implies that $\|\zeta(t)\| \leq \|\xi(t)\|$. So, we have

$$\|x\| \leq \|\zeta\| + \|\xi\| \leq 2\|\xi\|. \tag{52}$$

Therefore, Eq. (49) becomes

$$\begin{aligned} \dot{V} &\leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 \\ &\quad + 2l_1\|\xi\|^2 + l_0\|\xi\| \qquad (53) \\ &\leq -(\lambda_{min}(K)\|\xi\| - 2l_1\|\xi\| - l_0)\|\xi\| \\ &\quad - \lambda_{min}(K)\|\zeta\|^2 \leq 0 \qquad (54) \end{aligned}$$

provided that

$$\|\xi\| > \frac{l_0}{\lambda_{min}(K) - 2l_1} \tag{55}$$

with $\lambda_{min}(K) > 2l_1$.

Case 2: If $x < 0$ we have that $\zeta(t) < \xi(t)$ but $\zeta(t) \leq 0, \forall t \geq 0$ which implies that $\|\zeta(t)\| > \|\xi(t)\|$. So, we have

$$\|x\| \leq \|\zeta\| + \|\xi\| \leq 2\|\zeta\|. \tag{56}$$

Therefore, Eq. (49) becomes

$$\begin{aligned} \dot{V} &\leq -\lambda_{min}(K)\|\xi\|^2 - \lambda_{min}(K)\|\zeta\|^2 \\ &\quad + 2l_1\|\xi\|\|\zeta\| + l_0\|\xi\| \qquad (57) \\ &\leq -(\lambda_{min}(K)\|\xi\| - l_0)\|\xi\| \\ &\quad - (\lambda_{min}(K) - 2l_1)\|\zeta\|^2 \leq 0 \qquad (58) \end{aligned}$$

provided that

$$\|\xi\| > \frac{l_0}{\lambda_{min}(K)} \tag{59}$$

and $\lambda_{min}(K) > 2l_1$.

Conclusively, $\forall x \in R^n$ the Lyapunov candidate function becomes negative when $\|\xi\| > \frac{l_0}{\lambda_{min}(K) - 2l_1}$ and $\lambda_{min}(K) > 2l_1$.

In the sequel, inequality (55) together with (52), (56) demonstrate that the trajectories of $\xi(t)$ and $x(t)$ are uniformly bounded with respect to the arbitrarily small, (since K can be chosen sufficiently large), sets Ξ and X shown below

$$\Xi = \left\{\xi(t) : \|\xi(t)\| \leq \frac{2l_0}{\lambda_{min}(K)-2l_1},\ \lambda_{min}(K) > 2l_1 > 0\right\}$$

and

$$X = \left\{x(t) : \|x(t)\| \leq \frac{2l_0}{\lambda_{min}(K)},\ \lambda_{min}(K) > 2l_1 > 0\right\}.$$

Thus, we have proven the following theorem:

Theorem 7 Consider the system (14) with the modeling error term satisfying (35). Then the control law (21), (22) together with the update laws (32) and (37) guarantees the uniform ultimate boundedness with respect to the sets

1.

$$\Xi = \left\{\xi(t) : \|\xi(t)\| \leq \frac{2l_0}{\lambda_{min}(K)-2l_1},\ \lambda_{min}(K) > 2l_1 > 0\right\}$$

2.

$$X = \left\{x(t) : \|x(t)\| \leq \frac{2l_0}{\lambda_{min}(K)},\ \lambda_{min}(K) > 2l_1 > 0\right\}$$

Furthermore,

$$\dot{\xi} = -K\xi + X_f\tilde{W}_f S_f(x) + X_g\tilde{W}_g S_g(x)u - \omega(x).$$

Hence, since the boundedness of $\tilde{W}_f$ and $\tilde{W}_g$ is assured by the use of the hopping algorithm and $\omega(x)$ owing to (35) and Theorem 7, we conclude that $\dot{\xi} \in L_\infty$.

Remark 2 *The previous analysis reveals that in the case where we have a modeling error different from zero at $\|x\| = 0$, our adaptive regulator can guarantee at least uniform ultimate boundedness of all signals in the closed loop. In particular, Theorem 7 shows that if l_0 is sufficiently small, or if the design constant K is chosen such that $\lambda_{min}(K) > 2l_1$, then $\|x(t)\|$ can be arbitrarily close to zero and in the limit as $K \to \infty$, actually becomes zero but as we stated in Remark 1, implementation issues constrain the maximum allowable value of K.*

5 Simulation Results

To demonstrate the potency of the proposed scheme we present simulation results which assume modeling errors. So, we tested the ability of the proposed direct control scheme to regulate a Dc Motor, under the presence of modeling error distinguishing two cases. The first case defined as "Complete Model Matching at Zero Case" where the modeling error depends on the states and the control inputs, while in the second case we have "Modeling Error at Zero Case" where the modeling errors depends on the states, the control inputs and a not-necessarily-known modeling error which is of the known constant value different than zero. All the simulation results present a comparison between the proposed method and a simple RHONN direct controller [21], which shows off the performance superiority of the proposed method.

5.1 Direct Control of DC Motor When We Have "Complete Model Matching at Zero Case"

We apply the proposed approach to control the speed of a 1 KW DC motor with a normalized model described by the following dynamical equations [21]

$$\begin{aligned} T_a\frac{dI_a}{dt} &= -I_a - \Phi\Omega + V_a \\ T_m\frac{d\Omega}{dt} &= \Phi I_a - K_0\Omega - m_L \\ T_f\frac{d\Phi}{dt} &= -I_f + V_f \\ \Phi &= \frac{aI_f}{1+bI_f} \end{aligned} \tag{60}$$

The states are chosen to be the armature current, the angular speed and the stator flux, $x = \left[I_a\Omega\Phi\right]$. As control inputs the armature and the field voltages, $u = \left[V_aV_f\right]$ are used. With this choice, we have

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}-\frac{1}{T_a}x_1 - \frac{1}{T_a}x_2x_3\\ \frac{1}{T_m}x_1x_3 - \frac{K_0}{T_m}x_2 - \frac{m_L}{T_m}\\ -\frac{1}{T_f}\frac{x_3}{a-\beta x_3}\end{bmatrix} + \begin{bmatrix}\frac{1}{T_a} & 0\\ 0 & 0\\ 0 & \frac{1}{T_f}\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} \tag{61}$$

which is of a nonlinear, affine in the control form.

Assuming the existence of modeling errors, we add disturbance terms in the two states x_1 and x_2 as follows

$$\omega(x_1,u_1) = 2x_1 + 2\sin(10x_1) + \sin(3u_1)$$
$$\omega(x_2,u_2) = 3x_2 + \sin(5x_2) + \sin(2u_2)$$

In many control schemes of the literature V_f is assumed constant. This may naturally occur when the

field is produced by a permanent magnet or when it may be separately excited but is intentionally kept constant. This assumption may facilitate things because if V_f is constant then Φ is constant and the above nonlinear 3^{rd} order system can be linearized and reduced to a second order form having 2 states ($x_1 = I_a$ and $x_2 = \Omega$), with the value Φ being included as a constant parameter.

$$T_a \frac{dI_a}{dt} = -I_a - \Phi\Omega + V_a$$

$$T_m \frac{d\Omega}{dt} = \Phi I_a - K_0\Omega - m_L$$

In a more general case, however, V_f is not considered constant and this scheme can also be used for armature and field weakening control of the separately excited Dc motor. Moreover, if the motor characteristics are not exactly known we may consider that the nonlinear model is unknown and therefore its control can be accomplished using the proposed neuro-fuzzy approach. In this case, the regulation problem of a DC motor is translated as follows: Find a state feedback to force the angular velocity Ω and the armature current I_a to go to zero, while the magnetic flux varies.

Motivated by this simplification (2^{nd} instead of 3^{rd}), we first assume that the system is described, within a degree of accuracy, by the 2^{nd} order neuro-fuzzy system of the form (12), where $x_1 = I_a$ and $x_2 = \Omega$. So, the number of states is $n = 2$, the number of fuzzy output partitions of each f_i is $m = 5$ with the ranges of f_1 [-182.5667 , 0], f_2 [-19.3627 , 30.0566] and the depth of high order sigmoid terms $k = 5$. In this case $s_i(x)$ assume high order connection up to the second order. The number of fuzzy partitions of each g_{ii} is selected to be $m = 3$ with the ranges of g_{11} [148 , 150] and g_{22} [42 , 44], using only the first order sigmoid term.

However, in the simulations carried out, the actual system is simulated by using the complete set of equations (61). The produced control law described in (21) and (22) is applied to this system, which in turn produces states x_1, x_2, which are in the sequel used in the updating laws of the controller's weights.

We simulated a 1KW DC motor with parameter values that can be seen in Table 1 and sampling time 10^{-3} sec. In order our model to be equivalent with RHONN's regarding to other parameters we have chosen the initial values of all variables as $[I_a \quad \Omega \quad \Phi] = [1 \quad 1 \quad 0.98]$, the initial weights $W_{f_i} = [0]$, $W_{g_{ij}} = [1]$ and the updating learning rates $\gamma_1 = 0.01$ and $\gamma_2 = 25$. Also, the parameters of the sigmoidal terms were chosen to be $a_1 = 0.4$, $a_2 = 5$, $b_1 = b_2 = 1$, $c_1 = c_2 = 0$, while the diagonal elements of matrix K were $k_1 = 5$, $k_2 = 10$.

As concerning comparison abilities Figure (5) gives the evolution of the states x_1 and x_2, which are the armature current and angular velocity of the RHONN [21] (red line) and the proposed Fuzzy-RHONN model (blue line), with time respectively where we can observe that the RHONN Model has oscillations, while the Fuzzy-RHONN has smooth development going close to zero as expected. Also, figures (6) and (7) show the evolution of control inputs and disturbances for RHONN (red line) and F-RHONN (blue line) with time, respectively.

5.2 Direct Control of DC Motor When We Have "Modeling Error at Zero Case"

Assuming the existence of modeling errors again to the same Dc Motor, we add disturbance terms in the two states x_1 and x_2 as follows

$$\omega(x_1, u_1) = 3x_1 + \sin(0.1x_1) + \sin(0.1u_1) + 1$$
$$\omega(x_2, u_2) = 2x_2 + \sin(1000x_2) + \sin(1000u_2) + 1$$

We performed three different simulations with varying values for the elements of matrix K, $\kappa_i = 120, 140, 160$ with $i = 1, 2$ and $\kappa_1 = \kappa_2$.

As concerning comparison abilities figures (8), (9) and (10) give the evolution of state x_2, which is the angular velocity of the RHONN [21] (red line) and the proposed Fuzzy-RHONN (blue line) models and the disturbance, with time, respectively. It can be observed that the RHONN Model converges to zero slower when compared to the proposed adaptive control algorithm. Also, while K changes and more precisely when its values are increasing then our model converges to zero faster any time and keeps peak values constant against RHONN's which have slower convergence and bigger peak values.

Table 1. Parameter values for the DC motor.

Parameter	Value
$1/T_a$	148.88 sec^{-1}
$1/T_m$	42.91 sec^{-1}
K_0/T_m	0.0129 $N \cdot m/rad$
T_f	31.88 sec
m_L	0.0
a	2.6
β	1.6

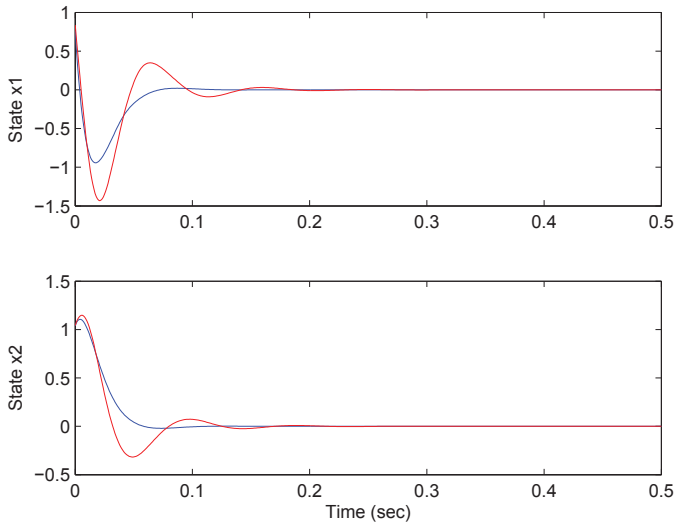


Figure 5. Evolution of armature current and angular velocity x_1 and x_2 respectively, for RHONN's (red line) and F-RHONN approach (blue line).

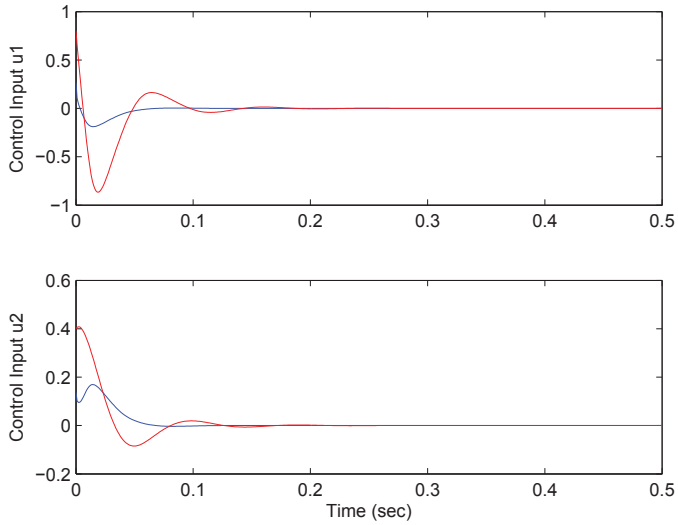


Figure 6. Evolution of control inputs u_1 and u_2 respectively, for RHONN's (red line) and F-RHONN approach (blue line).

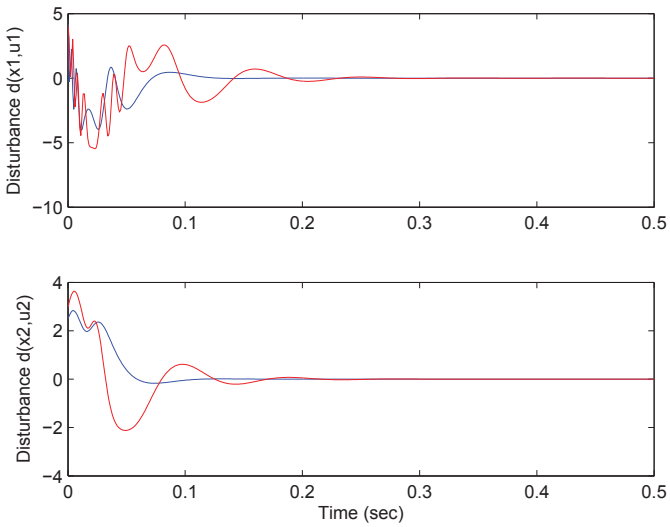


Figure 7. Evolution of disturbances $d(x_1,u_1)$ and $d(x_2,u_2)$ respectively, for RHONN's (red line) and F-RHONN approach (blue line).

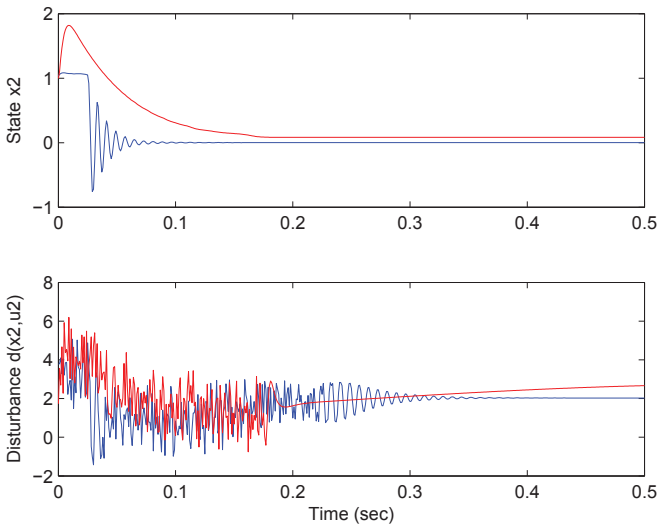


Figure 8. Evolution of angular velocity x_2 for Fuzzy-RHONN and RHONN Models when $\kappa_i = 120$.

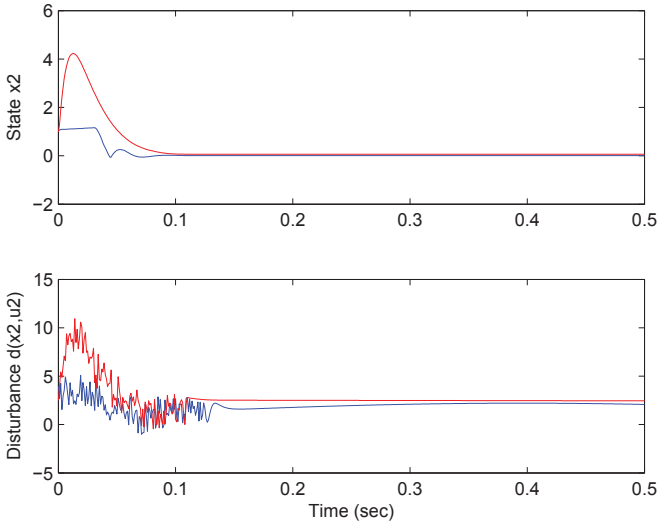


Figure 9. Evolution of angular velocity x_2 for Fuzzy-RHONN and RHONN Models when $\kappa_i = 140$.

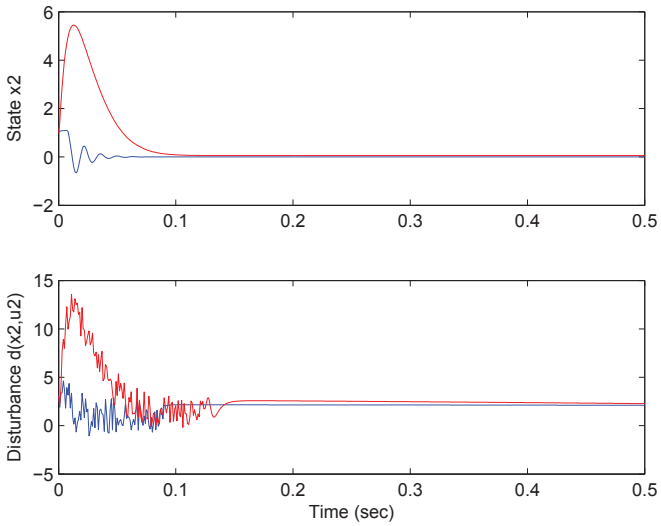


Figure 10. Evolution of angular velocity x_2 for Fuzzy-RHONN and RHONN Models when $\kappa_i = 160$.

6 Conclusion

The robustifying analysis of a direct adaptive control scheme was considered in this paper, aiming at the regulation of nonlinear unknown plants. The approach is based on a new Neuro-Fuzzy Dynamical Systems definition, which uses the concept of Fuzzy Dynamical Systems (FDS) operating in conjunction with High Order Neural Network (F-HONN's). Since the plant is considered unknown, we propose its approximation by a special form of an affine in the control fuzzy system (FDS) and in the sequel the fuzzy rules are approximated by appropriate HONN's. The fuzzy-recurrent high order neural networks are used as models of the unknown plant, practically transforming the original unknown system into a F-RHONN model which is of the known structure, but contains a number of unknown constant value parameters known as synaptic weights. The proposed scheme does not require a-priori experts' information on the number and type of input variable membership functions making it less vulnerable to initial design assumptions, is computationally very fast and thus can be used in several real-time engineering applications. Weight updating laws for the involved HONN's are provided, which guarantee that the system states reach zero exponentially fast, while keeping all signals in the closed loop bounded. A novel method of parameter hopping developed for the first time by the authors, assures the existence of the control signal and is incorporated in the weight updating law. Simulations illustrate the potency of the method in controlling an unknown nonlinear multivariable plant. Compared to simple RHONN direct control, the proposed method proves to be superior.

References

[1] Y. S. Boutalis, D. C. Theodoridis, and M. A. Christodoulou. A new neuro fds definition for indirect adaptive control of unknown nonlinear systems using a method of parameter hopping. *IEEE Transactions on Neural Networks*, 20(4):609–625, 2009.

[2] M. Chemachema and K. Belarbi. Robust direct adaptive controller for a class of nonlinear systems based on neural networks and fuzzy logic system. *International Journal on Artificial Intelligence Tools*, 16:553–560, 2007.

[3] K.B. Cho and B.H.Wang. Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction. *Fuzzy Sets and Systems*, 83:325–339, 1996.

[4] M. A. Christodoulou, D. C. Theodoridis, and Y. S. Boutalis. Building optimal fuzzy dynamical systems description based on recurrent neural network approximation. In *Conference of Networked Distributed Systems for Intelligent Sensing and Control*, pages 82–93, Kalamata, Greece, June 2007.

[5] Y. Diao and K. M. Passino. Adaptive neural/fuzzy control for interpolated nonlinear systems. *IEEE Transactions on Fuzzy Systems*, 10:583–595, 2002.

[6] S. S. Ge and W. Jing. Robust adaptive neural control for a class of perturbed strict feedback nonlinear systems. *IEEE Transactions on Neural Networks*, 13(6):1409 – 1419, 2002.

[7] W. M. Haddada, T. Hayakawaa, and V. Chellaboina. Robust adaptive control for nonlinear uncertain systems. *Automatica*, 39:551 556, 2003.

[8] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[9] P. Ioannou and B. Fidan. *Adaptive control tutorial*. SIAM, 2006.

[10] C.F. Juang and C. T. Lin. An on-line self-constructing neural fuzzy inference network and its applications. *IEEE Transactions on Fuzzy Systems*, 6:12–32, 1998.

[11] Y. T. Kim and Z. Z. Bien. Robust adaptive fuzzy control in the presence of external disturbance and approximation error. *Fuzzy Sets and Systems*, 148:377393, 2004.

[12] E. B. Kosmatopoulos and M. A. Christodoulou. Recurrent neural networks for approximation of fuzzy dynamical systems. *Int. Journal of Intelligent Control and Systems*, 1:223–233, 1996.

[13] C.T. Lin. A neural fuzzy control system with structure and parameter learning. *Fuzzy Sets and Systems*, 70:183–212, 1995.

[14] Y. H. Lin and G. A. Cunningham. A new approach to fuzzy-neural system modelling. *IEEE Transactions on Fuzzy Systems*, 3:190–197, 1995.

[15] E. Mamdani. Advances in the linguistic synthesis of fuzzy controllers. *Int. Journal of Man-Machine Studies*, 8(6):669678, 1976.

[16] S. Mitra and Y. Hayashi. Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE Transactions on Neural Networks*, 11:748–768, 2000.

[17] H. N. Nounou and K. M. Passino. Stable autotuning of hybrid adaptive fuzzy/neural controllers for nonlinear systems. *Engineering Applications of Artificial Intelligence*, 18:317–334, September 2005.

[18] R. Ordonez and K. M. Passino. Adaptive control for a class of nonlinear systems with time-varying structure. *IEEE Transactins on Automatic Control*, 46(1):152–155, 2001.

[19] K. Passino and S. Yurkovich. *Fuzzy Control*. Addison, 1998.

[20] G. A. Rovithakis and M. A. Christodoulou. Adaptive control of unknown plants using dynamical neural networks. *IEEE Transactions on Systems, Man and Cybernetics*, 24:400–412, 1994.

[21] G.A. Rovithakis and M.A.Christodoulou. *Adaptive Control with Recurrent High Order Neural Networks (Theory and Industrial Applications), in Advances in Industrial Control*. Springer Verlag London Limited, 2000.

[22] D. C. Theodoridis, Y. S. Boutalis, and M. A. Christodoulou. Direct adaptive control of unknown nonlinear systems using a new neuro-fuzzy method together with a novel approach of parameter hopping. *Kybernetica*, 45(3):349–386, 2009.

[23] D. C. Theodoridis, Y. S. Boutalis, and M. A. Christodoulou. A new neuro-fuzzy dynamical system definition based on high order neural network function approximators. In *European Control Conference ECC-09*, Budapest, Hungary, August 2009.

[24] D. C. Theodoridis, M. A. Christodoulou, and Y. S. Boutalis. Indirect adaptive neuro - fuzzy control based on high order neural network function approximators. In *In Proccedings 16th Mediterranean Conference on Control and Automation - MED08*, pages 386–393, Ajaccio, Corsica, France, 2008.

[25] S. Tong and T. Chai. Direct adaptive control and robust analysis for unknown multivariable nonlinear systems with fuzzy logic systems. *Fuzzy Sets and Systems*, 106:309–319, 1999.

[26] L. Wang. *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Prentice Hall, 1994.

[27] Y. Yang. Direct robust adaptive fuzzy control (drafc) for uncertain nonlinear systems using small gain theorem. *Fuzzy Sets and Systems*, 151:79–97, May 2005.

JAISCR, 2011, Vol.1, No.1, pp. 81–89

DISCOVERING DIAGNOSTIC GENE TARGETS FOR EARLY DIAGNOSIS OF ACUTE GVHD USING METHODS OF COMPUTATIONAL INTELLIGENCE ON GENE EXPRESSION DATA

Maurizio Fiasché[1], Francesco C. Morabito[1]
Anju Verma[2], Nikola Kasabov[2]
Maria Cuzzola[3], Pasquale Iacopino[3]

[1]*DIMET, University Mediterranea of Reggio Calabria*
Via Graziella, Feo di Vito, 89100 Reggio Calabria, Italy
{*maurizio.fiasche, morabito*}*@unirc.it*

[2]*Knowledge Engineering and Discovery Research Institute,*
AUT350 Queen Street, Auckland, New Zealand
{*averma, nkasabov*}*@aut.ac.nz*

[3]*Transplant Regional Center of Stem Cells and Cellular Therapy, "A. Neri"*
Via Petrara 11, 89100 Reggio Calabria, Italy

Abstract

This is an application paper of applying standard methods of computational intelligence to identify diagnostic gene targets and to use them for a successful diagnosis of a medical problem - acute graft-versus-host disease (aGVHD). This is the major complication after allogeneic haematopoietic stem cell transplantation (HSCT) in which functional immune cells of donor, recognize the recipient as "foreign" and mount an immunologic attack. In this paper we analyzed gene-expression profiles of 47 genes associated with allo-reactivity in 59 patients submitted to HSCT. We have applied different dimensionality reduction techniques of the variable space, combined with different classifiers to detect the aGVHD at onset of clinical signs. This is a preliminary study which utilises both computational and biological evidence for the involvement of a limited number of genes for the diagnosis of aGVHD. Directions for further studies are also outlined in this paper.

1 Introduction

With the completion of the first draft of the human genome the task is now to be able to process this vast amount of ever growing dynamic information and to create intelligent systems for detection, prediction and knowledge discoveries about human pathology and disease. When genes are in action, the dynamics of the processes, in which a single gene is involved, are very complex, as this gene interacts with many other genes and mediators, and is influenced by many environmental factors. The genes in an individual may mutate, change slightly their code, and may therefore express differently at a next time. Modeling these events, learning about them and extracting knowledge are major goals for *bioinformatics* [1, 2].

The potential applications of microarray technology are numerous and include identifying markers for classification, diagnosis, disease outcome prediction, target identification and therapeutic responsiveness [1, 2]. Microarray analysis might help to identify unique markers (e.g. a set of gene) of

clinical importance. Diagnosis and prediction of a biological state/disease is likely to be more accurate by identifying clusters of gene expression profiles (GEPs) performed by macroarray analysis. Based on a genetic profile, it is possible to set a diagnostic test, so a sample can be taken from a patient, the data related to the sample processed, and a profile related to the sample obtained [2]. This profile can be matched against existing gene profiles and based on similarity, it can be confirmed with a certain probability the presence or the risk for a disease. We apply this approach here to detect acute graft-versus-host disease (aGVHD) in allogeneic hematopoietic stem cell transplantation (HSCT), a curative therapy for several malignant and non malignant disorders [3].

Acute GVHD remains the major complication and the principal cause of mortality and morbility following HSCT [4, 5]. At present, the diagnosis of aGVHD is merely based on clinical criteria and may be confirmed by biopsy of one of the 3 target organs (skin, gastrointestinal tract, or liver) [6]. The severity of aGVHD is graded clinically from I to IV using a standardized system, with increased mortality rates associated with significant aGVHD (grades II-IV) [7]. There is no definitive diagnostic blood test for aGVHD, although a lot of blood proteins have been described as potential biomarkers in small studies [8, 9]. A recent report indicates a preliminary molecular signature of aGVHD in allogeneic HSCT patients [10].

In the current project, our primary objective was to validate a novel and not invasive method to confirm the diagnosis of aGVHD in HSCT patients at onset of clinical symptoms. For this purpose, a database has been built using pre-processed experimental measurements from patients, and features were selected to enable a good class separation without using the large amount of variables recorded features thus facing the "curse of dimensionality" problem (i.e., an excessive number of training inputs that increases the system complexity without remarkable advantages in terms of prediction performances). This problem can be considered as a typical inverse problem of pattern classification, starting from experimental database.

The proposed approach uses different dimensionality reduction techniques, such as Principal Component Analysis (PCA), Correlation-based Feature Selection (CFS) algorithm combined with an Artificial Neural Network (ANN) classifier, and also a wrapper method combined with the Naïve Bayesian classifier and with a Support Vector Machine (SVM) classifier to select the most important features (genes) for the diagnosis.

This is the first paper which discusses both computational and biological evidence to confirm the early statement of aGVHD based on selected genetic diagnostic markers. The organization of the rest of the paper is as follows: section 2 explains the data analyzed; a dimensionality reduction technique is applied in order to reduce the number of variables; section 3 describes the results obtained with our approach; section 4 discusses the results of the diagnostic method and the last section of the paper gives conclusions inferred with some possible future applications.

2 Methodology

Feature selection is the process of choosing the most appropriate features (variables) when creating a computational model [11]. Feature evaluation is the process of establishing how relevant to the problem in hand are the features used in the model. Features can be:

- Original variables: used in the first instance to specify the problem.
- Transformed variables: obtained through mapping the original variable space into a new one.

There are different groups of methods for feature selection:

- Filtering methods: features are 'filtered', selected and ranked in advance, before a model is created (e.g. a classification model). Traditional filtering methods are: correlation, t-test, and signal-to-noise ratio.
- Wrapping methods: features are selected on the basis of how well the created model performs using these features.

In this paper we consider three general approaches to feature subset selection, more specifically, wrapper and filter approaches, for gene selection and a

feature extraction technique based on the variance of data (PCA) for obtaining a new problem space of lower order.

Wrappers and filters differ in the way the feature subsets are evaluated. Filter approaches remove irrelevant features according to general characteristics of the data. Wrapper approaches, by contrast, apply machine learning algorithms to feature subsets and use cross-validation to evaluate the score of feature subsets. In theory, wrappers should provide more accurate classification results than filters (Langley, 1994) [11]. Wrappers use classifiers to estimate the usefulness of feature subsets.

The use of "tailor-made" feature subsets should provide better classification accuracy for the corresponding classifiers, since the features are selected according to their contribution to the classification accuracy of the classifiers. The disadvantage of the wrapper approach is its computational requirement when combined with sophisticated algorithms such as support vector machines.

As a filter approach, CFS was proposed by Hall [12]. The rationale behind this algorithm is "a good feature subset is one that contains features highly correlated with the class, yet uncorrelated with each other." It has been shown in Hall [12] that CFS gave comparable results to the wrapper and executes many times faster. It will be shown later in this paper that combining CFS with a suitable classifier, provides a good classification accuracy for diagnosis of aGVHD.

Another point of view is to consider PCA, for finding a representation of the problem space into another orthogonal space, having a smaller number of dimensions defined by another set of variables (eigenvectors). The new set of variables will be considered as an input for a suitable ANN.

2.1 Experimental Data

Fifty-nine HSCT patients were enrolled in our study between March 2006 and July metricconverterProductID2008 in2008 in Transplants Regional Center of Stem Cells and Cellular Therapy "A. Neri" Reggio Calabria, Italy, during a Governative Research Program: *"Project of Integrated Program: Allogeneic Hemopoietic Stem Cells Transplantation in Malignant Hemopathy and Solid Neoplasia Therapy - Predictive and prognostic value for graft vs. host disease of chimerism and gene expression"*.

Because experimental design plays a crucial role in a successful biomarker search, the first step in our design was to choose the most informative specimens and achieve adequate matching between positive cases aGVHD (YES) and negative controls aGVHD (NO) to avoid bias. This goal is best achieved through a database containing high-quality samples linked to quality controlled clinical information. Patients with clinical signs of aGVHD (YES) were selected, and in more than 95% of them aGVHD was confirmed by biopsy including those with grade I.

We used 26 samples from aGVHD (YES) patients that were taken at the time of diagnosis and we selected 33 samples from patients that did not experienced aGVHD (NO). All together YES/NO patient groups comprised a validation set. Total RNA was extracted from whole peripheral blood samples using a RNA easy Mini Kit (Qiagen) according to the manufacturer's instructions. Reverse transcription of the purified RNA was performed using Superscript III Reverse Transcriptase (Invitrogen). A multigene expression assay to test occurrence of aGVHD were carried out with TaqMan® Low Density Array Fluidic (LDA-macroarray card) based on Applied Biosystems 7900HT comparative dd CT method, according to manufacturer's instructions. Expression of each gene was measured in triplicate and then normalized to the reference gene 18S mRNA, who was included in macroarray card. About the project of macroarray card, we selected 47 candidate genes from the published literature, genomic databases, pathway analysis. The 47 candidate genes were involved in immune network and inflammation pathogenesis.

2.2 Dimensionality Reduction Approach

2.2.1 Feature Extraction with PCA

In statistics, PCA [13] is a technique that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components (PCs). PCA can be used for dimensionality reduction in a data set while retaining those characteristics of the data set that contribute most to its variance, by keeping lower-order PCs and ignoring higher-order ones. Such low-

order components often contain the "most important" aspects of the data, considering, moreover, that data set has been preliminarily dewhitened. PCA has the distinction of being the optimal linear transformation for keeping the subspace that has the largest variance.

In this paper we have applied PCA after a suitable data normalization. It is shown that the first 9 PCs account for more than 83% of the variation among the data samples (Fig. 1). A cut-off of the PCs selected has been established in order to balance the retained information and the "curse of dimensionality" problem. In section 3 it will be analyzed an evaluation about the usefulness of this new problem space representation, using a classifier.

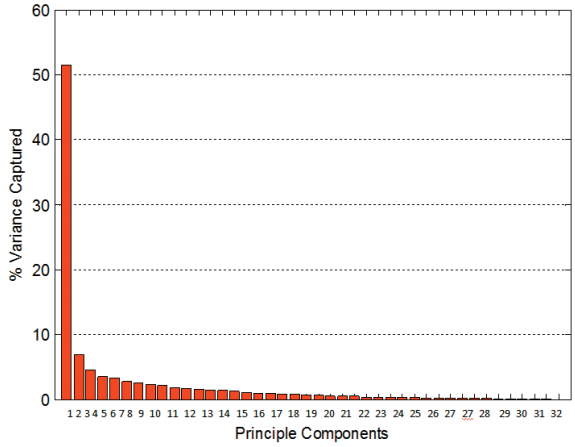


Figure 1. PCA transformation of the GvHD data, and the principal components neat for variance.

2.2.2 Gene Feature Selection

Feature Selection is a technique used in machine learning of selecting a subset of relevant features to build robust learning models. The assumption here is that not all genes measured by a macroarray method are related to aGVHD classification. Some genes are irrelevant and some are redundant from the machine learning point of view [12, 21]. It is well-known that the inclusion of irrelevant and redundant information may harm performance of some machine learning algorithms. Feature subset selection can be seen as a search through the space of feature subsets. CFS evaluates a subset of features by considering the individual detector ability of each feature along with the degree of redundancy between them

$$CFSs = \frac{k\bar{r}_{cf}}{\sqrt{k+k(k-1)\bar{r}_{ff}}} \quad (1)$$

Where

- $CFSs$ is the score of a feature subset S containing k features,
- $\bar{r}_{cf}$ is the average feature to class correlation ($f \in S$),
- $\bar{r}_{ff}$ is the average feature to feature correlation.

The distinction between normal filter algorithms and CFS is that while normal filters provide scores for each feature independently, CFS presents a heuristic "merit" of a feature subset and reports the best subset it finds. To select the genes with CFS, we have to:

a) Choose a search algorithm,

b) Perform the search, keeping track of the best subset encountered according to $CFSs$,

c) Output the best subset encountered.

The search algorithm we used was the best-first with forward selection, which starts with the empty set of genes. The search for the best subset is based on the training data only. Once the best subset has been determined, and a classifier has been built from the training data (reduced to the best features found), the performance of that classifier is evaluated on the test data. The 13 genes selected by CFS are reported in Table 1.

A leave-one-out cross validation procedure was performed to investigate the robustness of the feature selection procedures. In 29 runs, the subset of 13 genes was selected 28 times (96%) by CFS. Now it is possible to use a classifier to estimate the usefulness of feature subsets.

2.2.3 Wrapper Method

While CFS assigns a score to subset of features, Wrapper approaches take biases of machine learning algorithms into account when selecting features. The wrapper method applies a machine learning algorithm for a feature subset selection and uses cross-validation to compute a score for them.

Table 1. The 13 genes selected from CFS, with their names and meaning.

Gene Name	Official full name	Immune function
BCL2A1	BCL2-related protein A1	Anti- and pro-apoptotic regulator.
CASP1	Caspase 1, apoptosis-related cysteine peptidase	Central role in the execution-phase of cell apoptosis.
CCL7	chemokine (C-C motif) ligand 7	Substrate of matrix metalloproteinase 2
CD83	CD83 molecule	Dendritic cells regulation.
CXCL10	chemokine (C-X-C motif) ligand 10	Pleiotropic effects, including stimulation of monocytes, natural killer and T-cell migration, and modulation of adhesion molecule expression.
EGR2	Early growth response 2	transcription factor with three tandem C2H2-type zinc fingers.
FAS	TNF receptor superfamily, member 6)	Central role in the physiological regulation of programmed cell death.
ICOS	Inducible T-cell co-stimulator	Plays an important role in cell-cell signaling, immune responses, and regulation of cell proliferation.
IL4	Interleukin 4	Immune regulation.
IL10	Interleukin 10	Immune regulation.
SELP	selectin P	Correlation with endothelial cells.
SLPI	Stomatin (EPB72)-like 1	Elemental activities such as catalysis.
STAT6	transducer and activator of transcription 6, interleukin-4 induced	Regulation of IL4- mediated biological responses.

Table 2. The 7 genes selected (marked with °) through the wrapper- naïve Bayes method with their names and meaning. The 5 genes selected with SVM are marked with *.

Gene Name	Official full name	Immune function
CASP1°*	Caspase 1, apoptosis-related cysteine peptidase	Central role in the execution-phase of cell apoptosis.
EGR2°	Early growth response 2	transcription factor with three tandem C2H2-type zinc fingers.
CD52°*	CD52 antigen	B-cell activation.
SLPI°	Stomatin (EPB72)-like 1	Elemental activities such as catalysis.
ICOS°*	Inducible T-cell co-stimulator	Plays an important role in cell-cell signaling, immune responses, and regulation of cell proliferation.
IL10°*	Interleukin 10	Immune regulation. Foxp-3 * forkhead box P3 Regulatory T cells play important roles in the maintenance control of transplantation tolerance.
CXCL10°	chemokine (C-X-C motif) ligand 10	Pleiotropic effects, including stimulation of monocytes, natural killer and T-cell migration, and modulation of adhesion molecule expression.

In general, filters are much faster than wrappers. However, as far as the final classification accuracy is concerned, *wrappers* normally provide better results. The general argument is that the classifier that will be built from the feature subset should provide a better estimate of accuracy than other methods.

The main disadvantage of *wrapper* approaches is that during the feature selection process, the classifier must be repeatedly called to evaluate a subset. The main disadvantage of wrapper approaches is that during the feature selection process, the classifier must be repeatedly called to evaluate a subset. To select the genes using a wrapper method, we have to:

a) Choose a machine learning algorithm to evaluate the score of a feature subset.

b) Choose a search algorithm.

c) Perform the search, keeping track of the best subset encountered.

d) Output the best subset encountered.

As a machine learning algorithm, here we used s simple Bayesian classifier naïve Bayes and a SVM. The naïve Bayes classifier assumes that features are independent given the class. Its performance on data sets with redundant features can be improved by removing such features. A forward search strategy is normally used with naïve Bayes as it should immediately detect dependencies when harmful redundant features are added.

SVMs use a kernel function to implicitly map data to a high dimensional space. Then, they construct the maximum margin hyperplane by solving an optimization problem on the training data. Sequential minimal optimization (SMO) [16] is used in this paper to train a SVM with a Linear Kernel. SVMs have been shown to work well for high dimensional microarray data sets [17]. However, due to the high computational cost it is not very practical to use the wrapper method to select genes for SVMs.

Also here the search algorithm was the best-first with forward selection, starting with the empty set of genes. We report here the accuracy of classifiers built from the best feature subset found during the search. The search for the best subset is based on the training data only. Once the best subset has been determined, and a classifier has been built from the training data (reduced to the best features found), the performance of that classifier is evaluated on the test data. The 5 Genes selected using the wrapper method are shown in table 2.

Most of the genes selected are also similar to those of the 13 genes selected using the CFS method and the only two genes that are different are actually correlated to other genes from the set of 13 genes.

A leave-one-out cross validation procedure was performed to investigate the robustness of the method over the training set: in 29 runs, the subset of 7 genes was selected 26 times (90%) by the nave Bayes wrapper and the group of 5 genes, 29 times (100%) by the SMO. Section 4 has shown the performance of this technique estimated on the testing data.

3 Neural Network Model for Early Diagnosis Using the Selected Gene Diagnostic Markers

Artificial neural networks (ANNs) are commonly known as biologically inspired, highly sophisticated analytical techniques, capable of modeling extremely complex non-linear functions. Some ANN models are inspired by learning processes in cognitive system and by neurological functions of the brain. ANN are capable of predicting new observations (on specific variables) from other observations (on the same or other variables) after executing a process of learning from existing data [14]. Here we aim at a comparison between two models built with the use of two different feature selection methods. We have used a popular ANN architecture called MLP with back-propagation learning algorithm. The MLP is known to be a robust function approximator for prediction/classification problems. A suitable subset of samples for biological peculiarities has been chosen as training data set. The training data set had 29 patient samples (13 aGVHD(Yes) and 16 aGVHD(No)). The test data set consisted of 30 patient samples (13 aGVHD(Yes) and 17 aGVHD(No)). The ANN's outputs were:

– 0, if aGVHD diagnosis was Yes;

– 1, if aGVHD diagnosis was No.

The ANN based system was trained with adaptive rate of learning during a period of 400 epochs. We have developed one MLP model with 13 genes as input variables, and another - with the 9 PCs as input variables. The ANN, according to a consequence of the Kolmogorov's theorem [15], has a hidden layer with 27 neurons (for the 13 gene MLP model) and 19 neurons (for the 9 PCs MLP model). As an activation function for the hidden layer the tan-sigmoid function is used. A pure linear function was used for the output layer (Figure 2). After the training phase, the ANN was tested - final results shown in section 4.

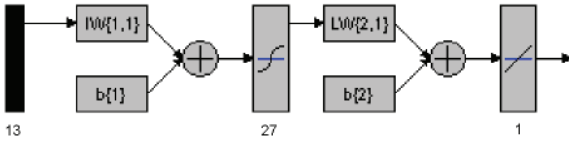


Figure 2. Structure of the implemented ANN.

In this study both the ANN and the SVM classifiers obtain similar results. The results confirm that it is possible to diagnose the aGVHD using a selected number of variables. Only one case escaped all our classification models, which achieved 97% accuracy in a leave one-out cross-validation on the testing data set. Experimental results are shown in Table 3 and Figure 3.

Table 3. Experimental results of a CFS with ANN classifier and a wrapper method combined with the naïve Bayesian classifier and with SVM. The starting set has been divided in training set and test set, a leave one-out cross-validation has been calculated for the two subsets.

Method	Training set	Test set
CFS-ANN	28(29)	29(30)
PCA-ANN	-	29(30)
Wrapper-Naïve Bayes	26(29)	29(30)
Wrapper-SVM	29(29)	29(30)

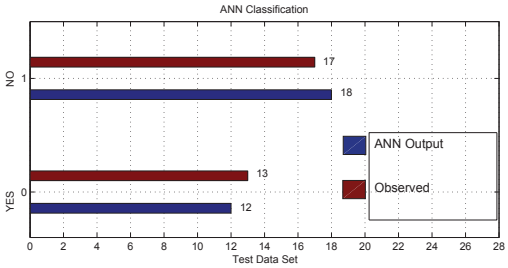


Figure 3. Test results for the ANN model.

Along with the good classification results that suggest a possible use for a clinical diagnostics, the following discoveries about related important genes were made:

a) In patients with aGVHD (YES), the following immune genes were over-expressed when compared with the reference normal values (it is assumed to be = 1): BCL2A1, CASP1, CCL7, CD83. For these genes it is very important to establish the cut-off expression value correlated with the YES event.

b) In contrast, the genes: CXCL10, EGR2, FAS, ICOS, IL-4, IL-10, SELP, SLP1, STAT6 were under-expressed during aGvHD and before of pharmacological treatment.

4 Biomedical Conclusions and Future Work

We examined the immune transcripts to study the applicability of gene expression profiling (macroarray) as a single assay in early diagnosis of aGVHD. Our interest was to select fewer number of molecular biomarkers from an initial gene panel and exploiting this to develop a fast, easy and non-invasive diagnostic test. The proposed method provides a good overall accuracy to confirm aGVHD development in HSCT setting.

From a biological point of view, the results are reliable. Others have reasoned that Th2 cell therapy could rapidly ameliorate severe aGVHD via IL-4 and IL-10 mediated mechanisms [19]. It is noteworthy that in our study a set of genes, indicated by computational analysis, included same mediators of Th2 response such as IL10, and signal transducer and activator of transcription 6, interleukin-4 induced (STAT6). All these were strongly down-regulated in aGVHD (YES) setting, suggesting ab-

sence of control mediated by Th2 cells. Therefore, we highlight the fact that defective expression of ICOS impaired the immune protective effectors during clinical aGVHD. This evidence is supported by a previous report about ICOS as regulatory molecule for T cell responses during aGVHD. It has been showed that ICOS signal inhibits aGVHD development mediated by CD8 positive effector cells in HSCT [20]. According to previous reports, mediators of apoptosis cells and dendritic cell activators were involved.

Altogether our results strongly outlined the importance and utility of non-invasive tool for aGVHD diagnosis based on GEP. We believe that to achieve an advantage from GEP performance, it is very important to know:

a) the transcript levels of immune effector cells in early time post-engraftment in order to better understand polarization of Th2 cells;

b) the CD8 positive cell action.

As a clinical trial, tissue biopsies were performed to confirm the above diagnostic results. In conclusion, our models may prevent the need for an invasive procedure.

This study demonstrated, for the first time, that the proposed here methodology for the selection of gene diagnostic targets and their use for early diagnosis of aGVHD results in a satisfactory 97% accuracy over independent test data set of HSCT population.

We plan to extend the system as a personalized model to capture peculiarity of patients through an optimization method [21-24]. A further approach to feature selection and model creation is the so called integrated approach [25], where features and model parameters are optimised together for a better accuracy of the model, which is an extension of the wrapper approach. As a classifier, a spiking neural network can be explored [25,26]. The authors are engaged in this direction.

References

[1] N. Kasabov, E*volving Connectionist Systems: The Knowledge Engineering Approach*, Springer, London, 2007.

[2] N. Kasabov , I.A. Sidorov, D.S. Dimitrov, *Computational Intelligence, Bioinformatics and Computational Biology: A Brief Overview of Methods*, Problems and Perspectives, J. Comp. and Theor. Nanosc., Vol.2,No.4,2005,473–491.

[3] F. R. Appelbaum, *Haematopoietic cell transplantation as immunotherapy*, Nature, 411, 2001, 385–389.

[4] D. Weisdorf, *Graft vs. Host disease: pathology, prophylaxis and therapy: GVHD overview*, Best Pr. & Res. Cl. Haematology, vol. 21, 2008, No.2, 99-100.

[5] P. Lewalle, R. Rouas, and P. Martiat, *Allogeneic hematopoietic stem cell transplantation for malignant disease: How to prevent graft-versus-host disease without jeopardizing the graft-versus-tumor effect?*, Drug Discovery Today: Therapeutic Strategies |Immunological disorders and autoimmunity,Vol.3,1, 2006.

[6] J.L. Ferrara, *Advances in the clinical management of GVHD*, Best Pr. & Res. Cl. Haematology, Vol.21,4, 2008,677–682.

[7] D. Przepiorka, D. Weisdorf, and P. Martin, *Consensus Conference on acute GVHD grading*, Bone Marrow Transplanation, vol. 15, 1995, 825–828.

[8] S. Paczesny, J.E. Levine, T.M. Braun, and J.L. Ferrara, *Plasma biomarkers in Graft-versus-Host Disease: a new era?*, Biology of Blood and Marrow Transplantation, vol. 15, 2009, 33–38.

[9] S. Paczesny, I. K. Oleg, and M. Thomas, *A biomarker panel for acute graft-versus-host disease*, Blood, vol. 113, 2009, 273–278.

[10] M.P. Buzzeo, J. Yang, G. Casella, and V. Reddy, *A preliminary gene expression profile of acute graft-versus-host disease*. Cell Transplantation, vol.17,5,2008,489–494.

[11] P. Langley, *Selection of relevant features in machine learning*. Proceedings of AAAI Fall Symposium on Relevance, 1994, 140–144.

[12] M.A. Hall, *Correlation-based feature selection for machine learning*. Ph.D. Thesis. Department of Computer Science, University of Waikato, New Zealand.

[13] P.J.A. Shaw, *Multivariate Statistic for the Environmental Sciences*, Hodder-Arnold, 2003.

[14] C. Bishop, *Neural Networks for Pattern Recognition*. Calderon-Press, Oxford, 1995.

[15] V. Kurkova, *Kolmogorov's theorem and multilayer neural networks*. Neural Networks, 5, 1992, 501–506.

[16] J. Platt., *Fast training of support vector machines using sequential minimal optimization*. Advances in Kernel Methods–Support Vector Learning. MIT Press, 1998.

[17] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer and D. Haussler, *Support vector machine classification and validation of cancer tissue samples using microarray expression data*. Bioinformatics 16, 2000, 906–914.

[18] D.B. Fogel, *An information criterion for optimal neural network selection*. IEEE Tran. N.N., 1991, 490–497.

[19] J.E. Foley Jason, J. Mariotti, K. Ryan, M. Eckhaus, and D.H. Fowler, *The cell therapy of established acute graft-versus-host disease requires IL-4 and IL-10 and is abrogated by IL-2 or host-type antigen-presenting cells*, Biology of Blood and Marrow Transplantation, vol. 14, 2008, 959–972.

[20] X.-Z Yu. Y. Liang, R.I. Nurieva, F. Guo, C. Anasetti, and C. Dong, *Opposing effects of ICOS on graft-versus-host disease mediated by CD4 and CD8 T cells1*, The Journal of Immunology. 176, 2006, 7394–7401.

[21] Y. Hu, Q. Song and N. Kasabov, *Personalized Modeling based Gene Selection for Microarray Data Analysis*. In: The 15th Int. Conf. on Neuro-Information Processing, ICONIP, Auckland, New Zealand, Nov. 2008, Springer LNCS vol.5506/5507, 2009

[22] N.Kasabov, *Global, local and personalised modelling and profile discovery in Bioinformatics: An integrated approach*, Pattern Recognition Letters, Vol.28, 6, 2007, 673–685.

[23] Q. Song and N. Kasabov, *TWNFI- a transductive neuro-fuzzy inference system with weighted data normalisation for personalised modelling*, Neural Networks, Vol.19, 10, Dec. 2006, 1591–1596

[24] N. Kasabov, *Data Analysis and Predictive Systems and Related Methodologies*, PCT Patent, NZ reg. 572036, 2008

[25] S. Schliebs, M. Defoin-Platel, S. Worner and N. Kasabov, *Integrated Feature and Parameter Optimization for Evolving Spiking Neural Networks: Exploring Heterogeneous Probabilistic Models*, Neural Networks, 22, 623–632, 2009.

[26] N. Kasabov, *To spike or not to spike: A probabilistic spiking neural model*, Neural Networks, Vol.23,1, Jan. 2010, 16–19