Acta Universitatis Sapientiae

Electrical and Mechanical Engineering

Volume 7, 2015

Sapientia Hungarian University of Transylvania
Scientia Publishing House

Contents

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 7 (2015) 5-22

Robust Model-Based Detection of Faults in the Air Path of Diesel Engines

Zsolt HORVÁTH[1], András EDELMAYER[2,3]

[1] School of Postgraduate Studies of Multidisciplinary Technical Sciences, Faculty of Technical Sciences, Széchenyi István University, Győr, e-mail: zsolt2.horvath@audi.hu
[2] Department of Informatics Engineering, Faculty of Technical Sciences, Széchenyi István University, Győr, e-mail: edelmayer@sze.hu
[3] Systems and Control Laboratory, Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, e-mail: edelmayer@sztaki.mta.hu

Manuscript received February 22, 2016; revised April 28, 2016

Abstract: This paper discusses the robust model-based fault detection filter design problem for faults in the air path of diesel engines. Two failure modes in the air path, namely, the Exhaust Gas Recirculation (EGR-Actuator) and Variable Geometry Turbocharger (VGT-Actuator) bias faults were considered. The objective of the design is the detection of failures, which is robust against model uncertainties and external disturbances. By using aH-infinity optimization approach the filter robustness is ensured by the application of a design trade-off that is made between the worst-case disturbance and the L_2 norm of the filter error. Beyond the classical state estimation, the method requires the solution of a linear-quadratic optimization problem that leads to the solution of the Modified Filter Algebraic Riccati Equation.

Keywords: Diesel engine diagnostics, fault detection, H-infinity optimization, robust estimation, actuator fault model.

1. Introduction

With the increasing complexity of combustion engines in current automotive vehicles, the early detection of failures for engine diagnostics plays an increasingly important role. From view of operational reliability of combustion engines, the assurance of error-free operation of the air path is an ultimately important design task. Possible faults are due to actuator, sensor and component failures, which can lead to engine malfunctions or even damages in the worst case.

There are many different types of Fault Detection and Isolation (FDI) methods available in the literature. Data-driven approaches are based on the

comparison of healthy and faulty data collected from the system. It can be quite expensive to collect, store and process data in real-time, especially in case of processes with fast dynamics. Model-based fault diagnosis [17, 18, 19] does not need stored data for analysis but, instead, it uses the mathematical model of the system, which is driven by temporary measurements acquired from the system. The response (output) of this model is compared with the output of the real physical process thus creating the residual. The residual characterizes the behaviour of the system. Perfect model matching creates zero residual meaning a fault-free operation, while a residual significantly different from zero (i.e., in engineering terms, when the residual value exceeds a certain threshold) indicates deviances from normal systems operation that may be caused by a fault. Because of the advantages, the model-based detection approaches became popular in the past decades, see e.g. [7, 15, 16].

Combustion engines can typically be represented by highly nonlinear processes that may have very fast dynamics. This property poses additional requirements for the design, modelling and filter implementation. Complexity of modelling and simplicity of implementation are design properties mutually opposed. From the one hand, the filter should be capable of running recursively, real-time, in few milliseconds cycles by taking the constrained computational capability of on-board microcontrollers into account. On the other hand, the computational complexity of the model might need processing power reasonably not available for the specific application.

The creation of an accurate mathematical model of the system is not an easy engineering task. Combustion engines, for instance, are complex, nonlinear physical systems that are subject to noise and other disturbances originated from many sources. These physical processes can best be approximated by using nonlinear system models. By making reasonable simplifications and conditionally neglecting nonlinearities the complexity of the models can be effectively reduced on the price that modelling will necessarily contain uncertainties. One of the major concerns of model-based detection filter design is, therefore, to ensure robustness of the detection process, meaning that the filter maintains its detection sensitivity to faults in the presence of disturbances and modelling uncertainties.

Model-based diagnosis of automotive engines has been considered earlier in [4], where a Nonlinear Unknown Input Observer (NUIO) for detection of actuator faults in diesel engines is presented. In [3] an adaptive Lyapunov-based fault estimation for leakage in the air path of diesel engines is used. To comply with the severe requirements on recursion speed of the filter by using nonlinear models a fuzzy filtering approach was proposed for sensor fault detection and isolation in diesel air path in [6]. For similar reasons, the study in [5] proposes the usage of neural network for filter implementation.

Advanced methods of robust detection filter design are based on geometric approaches and optimal filtering. Earlier results, see e.g. [7] have shown that the H_∞ fault detection filtering is more flexible and more robust than other approximate detection filter design techniques for example the unknown input observers. In this sense the H_∞ fault detection filter is to be implemented in our paper, whether it would be a suitable method for fault detection of actuator faults in the air path of diesel engines at the chosen operating point. For further investigation, according to our concept, for extending a filtering procedure for the entire engine operation range, a set of local Linear Time Invariant (LTI) models is to be created for different operating points and from these a corresponding filter gains can be calculated (Gain scheduling). In spite of the disturbance and model nonlinearities the H_∞ fault detection filtering approach may contribute to finding a useful solution.

2. Modelling the air path of diesel engines

2.1 Linear Time Invariant (LTI) model investigation, plant properties

The robust fault detection filter design methodologies used in this paper require using of Linear Time Invariant (LTI) model. For this end, we use a third order nonlinear mean-value model parametrized for low and medium speed load points, which covers the New European Drive Cycle (NEDC), it was proposed by Jung [1] for robust control purposes.

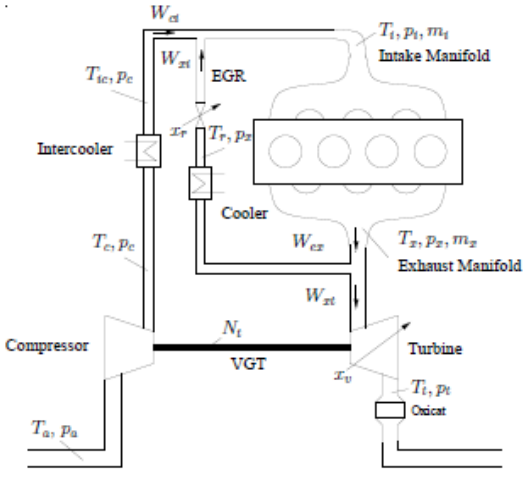


Figure 1: The scheme of the modelled turbocharged diesel engine [1]

The basic structure of the system is shown in Fig. 1 and presented in the following. The turbocharger dominates the air path, it consists of the turbine and compressor. The turbine driven by the exhaust gas flow has a Variable Geometry Turbocharger (VGT), the position of which corresponds to closing or opening the guide vanes, that increases or decreases the turbine speed respectively. When the turbine speed is increased, the air in the inlet charge is more compressed, i.e. increased Intake Manifold Pressure.

The second path from the exhaust gas to the Intake Manifold is the exhaust-gas recirculation, which is needed for NO_x reduction. The burned gas fraction is re-circulated into the Intake Manifold and it displaces fresh air in it. This is lowering the flame temperature and hence decreasing of NO_x in the exhaust gas. The basic structure of the system is shown in *Fig. 1*.

All the model parameters proposed by Jung [1] are summarised in *Table 1*. According to study in [1] the differential equations of the nonlinear model are

$$\dot{p}_i = \frac{RT_i}{V_i}\left(\frac{\eta_c}{c_p T_a}\frac{P_c}{\left(\frac{p_i}{p_a}\right)^{\mu}-1} + \frac{A_{egr}\left(x_{egr}\right)p_x}{\sqrt{RT_x}}\sqrt{\frac{2p_i}{p_x}\left(1-\frac{p_i}{p_x}\right)} - \eta_v\frac{p_i NV_d}{120T_i R}\right)$$

$$\dot{p}_x = \frac{RT_x}{V_x}\left(\begin{array}{l}\eta_v\frac{p_i NV_d}{120T_i R} + \frac{A_{egr}\left(x_{egr}\right)p_x}{\sqrt{RT_x}}\sqrt{\frac{2p_i}{p_x}\left(1-\frac{p_i}{p_x}\right)} - \\ -\left(ax_{vgt}+b\right)\left(c\left(\frac{p_x}{p_a}-1\right)+d\right)\frac{p_x}{p_{ref}}\sqrt{\frac{T_{ref}}{T_x}}\sqrt{\frac{2p_a}{p_x}\left(1-\frac{p_a}{p_x}\right)} + \mathrm{W}_f\end{array}\right) \tag{1}$$

$$\dot{P}_c = \frac{1}{\tau}\left(-\mathrm{P}_c + \eta_m\left(ax_{vgt}+b\right)\left(c\left(\frac{p_x}{p_a}-1\right)+d\right)\frac{p_x}{p_{ref}}\sqrt{\frac{T_{ref}}{T_x}}\sqrt{\frac{2p_a}{p_x}\left(1-\frac{p_a}{p_x}\right)}c_p T_x\eta_t\left(1-\left(\frac{p_a}{p_x}\right)^{\mu}\right)\right).$$

Table 1: Nonlinear model parameters

Parameter	*Name*	*Value*	*Units*
τ	turbocharger time constant	0,11	s
η_m	turbocharger mechanical efficiency	0,98	-
V_i	volume of the Intake Manifold	0,006	m^3
V_x	volume of the Exhaust Manifold	0,001	m^3
η_c	compressor efficiency	0,61	-
T_a	ambient temperature	298	K
c_p	specific heat at constant pressure	1014,4	*J/kgK*
c_v	specific heat at constant volume	727,4	*J/kgK*

$\mu = \frac{c_p - c_v}{c_p}$	specific heat ratio	0,286	-
p_a	ambient pressure	101325	Pa
T_x	exhaust gas temperature	509	K
R	gas constant	287	*J/kgK*
η_v	engine volumetric efficiency	0,87	-
T_i	gas temperature in the Intake Manifold	313	K
V_d	total engine displacement volume	0,002	m^3
p_{ref}	reference pressure	101325	Pa
T_{ref}	reference temperature	298	K
η_t	turbine efficiency	0,76	-
n	number of cylinders	4	-

We linearize this model around the specified equilibrium point (Herceg, 2006), with set values and model variables summarized in *Table 2*.

The operating point has been chosen in the medium speed region at 1900 rpm according to the requirements of NEDC see in study [1].

Table 2: States and Input/Output variables of the system

Variable	*Notation*	*Function*	*Valuein the equilibrium point*
EGR-Actuator effective area	ΔA_{egr}	Input	$4 \cdot 10^{-5}$ m^2
VGT-Actuator position	Δx_{vgt}	Input	70 %
Engine speed	ΔN	Disturbance	1900 1/min
Fuelling	ΔW_f	Input	0.032 kg/s
Intake manifold pressure (p_i)	Δx_1 Δy_1	State variable 1. Output 1.	124200 Pa
Exhaust manifold pressure (p_x)	Δx_2 Δy_2	State variable 2. Output 2.	131000 Pa
Turbine power (P_c)	Δx_3	State variable 3.	930 W
Air mass	ΔW_a Δy_3	Output 3.	-

According to study in [1] the control inputs of the LTI system are the position of the Exhaust Gas Recirculation Valve (EGR-Actuator) and Variable Geometry Turbocharger (VGT-Actuator), the outputs are the Sensor for Air Mass Flow into the Intake Manifold (MAF), the Sensor for Exhaust Gas Pressure (EMAP) and the Sensor for Intake Manifold Pressure (MAP).
Noting that in the original model [1] two outputs have been used, namely MAF and MAP, but for the reason of better observability of the system we have extended this to one additional output namely EMAP, which may be nowadays realized in the real application without any problem. Using tree outputs additionally means that the Exhaust Manifold pressure is measured too. This may result in a better performance of the fault detection filtering.

For the sake of simplification, we considered fuelling as a constant input of the air path and not as disturbance. The disturbance was modelled as the fluctuating change of the engine speed.
By linearization we get the LTI model. It can be written in the state space as

$$\begin{aligned}
\begin{bmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \\ \Delta\dot{x}_3 \end{bmatrix} &= \begin{bmatrix} -5.2643 & 4.7316 & 28.5021 \\ 50.7697 & -156.9827 & 0 \\ 0 & 0.4287 & -9.0909 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \end{bmatrix} + \\
&+ \begin{bmatrix} 1.6111 \cdot 10^{9} & 0 & 0 \\ -1.5720 \cdot 10^{10} & 8.3514 \cdot 10^{4} & 1.46083 \cdot 10^{8} \\ 0 & -141.6484 & 0 \end{bmatrix} \begin{bmatrix} \Delta A_{egr} \\ \Delta x_{vgt} \\ \Delta W_f \end{bmatrix} + \\
&+ \begin{bmatrix} -47.7946 \\ 466.3408 \\ 0 \end{bmatrix} \Delta N, \\
\Delta y = \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \Delta y_3 \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3.924 * 10^{-5} \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \end{bmatrix}.
\end{aligned} \tag{2}$$

The matrices

$$A = \begin{bmatrix} -5.2643 & 4.7316 & 28.5021 \\ 50.7697 & -156.9827 & 0 \\ 0 & 0.4287 & -9.0909 \end{bmatrix},$$

$$B = \begin{bmatrix} 1.6111 \cdot 10^{9} & 0 & 0 \\ -1.5720 \cdot 10^{10} & 8.3514 \cdot 10^{4} & 1.46083 \cdot 10^{8} \\ 0 & -141.6484 & 0 \end{bmatrix}, \tag{3}$$

$$B_{\omega} = \begin{bmatrix} -47.7946 \\ 466.3408 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3.924 \cdot 10^{-5} \end{bmatrix}.$$

are appropriate constant matrices, B_{ω} is the matrix for disturbance acting on the system.

2.2 Modelling of faults in the air path of diesel engines

The model for the faults considered in this paper is described below. First we introduce the concept of fault modelling.

Considering the LTI system model according to study in [7] subjected to disturbance and unknown faults, which can be represented in state space form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_{\omega}\omega(t) + \sum_{i=1}^{k} L_i \nu_i(t), \\ y(t) &= Cx(t). \end{aligned} \tag{4}$$

with $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, $u \in \mathbb{R}^m$. $\omega \in \mathbb{R}^p$ are the process disturbances in $L_2[0,T]$.
A, B, C and B_{ω} are appropriate constant matrices. Assume, that (A,C) is an observable pair.

The cumulative effect of k number of faults appearing in known directions L_i of the state space is modelled by additive linear term $\sum L_i \nu_i(t)$. $L_i \in \mathbb{R}^{nxs}$ and $\nu_i(t)$ are the fault signatures and failure modes respectively. $\nu_i(t)$ are arbitrary unknown time functions for$t \geq t_{ji}$, $0 \leq t \leq T$, where t_{ji} is the time instant when the i-th faults appears and $\nu_i = 0$, if $t < t_{ji}$. If $\nu_i(t) = 0$, for all i, then the plant is assumed to be fault free. Assume, however, that only one fault appears in the system at a time.

The LTI model including the engine speed disturbance is extended to include two actuator faults: an EGR-Actuator fault and a VGT-Actuator fault, denoted by $f_{egr}(t)$ and $f_{vgt}(t)$ respectively. The disturbance was modelled as a changing engine speed caused by variable load (wind, ramp, break, etc.) that corresponds the real driver situation.

According to equation (4) the EGR-Actuator fault and a VGT-Actuator fault can be modelled as additive terms in the state equations as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_{\omega}\omega(t) + Bf_a(t), \\ y(t) &= Cx(t). \end{aligned} \tag{5}$$

As the actuator faults enter the system in the same direction as the input does, the fault directions matrices can be signed as the input matrices.

The faults can be represented as a vector with two arbitrary unknown time functions as

$$f_a(t) = \begin{bmatrix} f_{egr}(t) \\ f_{vgt}(t) \\ 0 \end{bmatrix}. \tag{6}$$

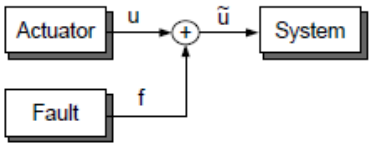


Figure 2: Additive fault model

As mentioned above, the state space realisation of the LTI model of the faulty system is described as

$$\begin{bmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \\ \Delta\dot{x}_3 \end{bmatrix} = \begin{bmatrix} -5.2643 & 4.7316 & 28.5021 \\ 50.7697 & -156.9827 & 0 \\ 0 & 0.4287 & -9.0909 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \end{bmatrix} +$$
$$+ \begin{bmatrix} 1.6111\cdot 10^{9} & 0 & 0 \\ -1.5720\cdot 10^{10} & 8.3514\cdot 10^{4} & 1.46083\cdot 10^{8} \\ 0 & -141.6484 & 0 \end{bmatrix} \begin{bmatrix} \Delta A_{egr} \\ \Delta x_{vgt} \\ \Delta W_f \end{bmatrix} + \begin{bmatrix} -47.7946 \\ 466.3408 \\ 0 \end{bmatrix} \Delta N +$$
$$+ \begin{bmatrix} 1.6111\cdot 10^{9} & 0 & 0 \\ -1.5720\cdot 10^{10} & 8.3514\cdot 10^{4} & 1.46083\cdot 10^{8} \\ 0 & -141.6484 & 0 \end{bmatrix} \begin{bmatrix} \Delta f_{egr} \\ \Delta f_{vgt} \\ 0 \end{bmatrix}, \tag{7}$$

$$\Delta y = \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \Delta y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3.924 \cdot 10^{-5} \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \end{bmatrix}.$$

3. Robust H_∞ detection filter

3.1 The optimal H_∞ detection filtering problem

Our objective is the detection of failure modes of two actuator faults in the presence of the modelling uncertainties and disturbances.

The goal of H_∞ filtering is minimization of the magnitude of perturbations' effects on the filter output through the appropriate choice of filter gain, maximizing the magnitude of the transfer function from failure modes to the filter error.

We can represent this estimation problem as a mixed H_2 / H_∞ filtering problem (Edelmayer, 2012).

The filter gain matrix has to take under consideration that the magnitude of transfer function computed from unknown input to the output error of the filter must be always less than a pre-specified level $\gamma > 0$ (Grimble, 1987).

To propose the design of H_∞ filter we must first consider the LTI system model (see equation (4)) extended to an observer. The state estimate can be obtained as

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + K(C(x(t) - \hat{x}(t))) + Bu(t), \\ \hat{y}(t) &= C\hat{x}(t), \\ \hat{z}(t) &= C_z\, \hat{x}(t). \end{aligned} \tag{8}$$

with the observer state $\hat{x} \in \mathbb{R}^n$, output estimation $\hat{y} \in \mathbb{R}^p$ and weighted output estimation $\hat{z} \in \mathbb{R}^p$. K is the observer gain matrix and C_z is the constant estimation weighting.

The filter error system can be derived as

$$\begin{aligned} \dot{\tilde{x}}(t) &= (A - KC)\,\tilde{x}(t) + B_w w(t) + \sum_{i=1}^{k} L_i \nu_i(t), \\ \varepsilon(t) &= C_z \tilde{x}(t). \end{aligned} \tag{9}$$

where $\tilde{x}(t)$ and $\varepsilon(t)$ are the state error and weighted output error, respectively. They are defined as

$$\begin{aligned} \tilde{x}(t) &= x(t) - \hat{x}(t), \\ \varepsilon(t) &= z(t) - \hat{z}(t). \end{aligned} \tag{10}$$

In the presence of faults, the estimation error does not converge asymptotically to zero, but converges asymptotically to the subspace which is different from zero.

In the following step of the design procedure we have to choose the filter gain, by minimizing the magnitude of the effects of perturbations on the filter output subject to maximize the magnitude of the transfer function from failure modes to the filter error. According to the representation in [7] we consider the auxiliary system written in the form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_{\kappa}\kappa(t) + \sum_{i=1}^{k} L_i \nu_i(t), \\ y(t) &= Cx(t). \end{aligned} \tag{11}$$

This does not includes parametric uncertainty.

$B_{\kappa} = [B_w, L_{\Delta}]$ is the worst-case input direction and $\kappa(t) \in L_2[0,T]$ is the input function for all $t \in \mathbb{R}_+$ representing the worst–case effects of modelling uncertainties and external disturbances.

The performance can be formulated as min-max problem by worst-case disturbance κ , which is minimizing the H-infinity norm of the transfer function of the worst-case disturbance denoted by $H_{\varepsilon\kappa}$ to the filter output. The worst-case performance measure is given by

$$J(\mathrm{K},\kappa) = \sup \frac{\left\| z - \hat{z} \right\|_2}{\left\| \kappa \right\|_2} = \left\| H_{\varepsilon\kappa}(s) \right\|_{\infty} . \tag{12}$$

The filter gain K can be obtained by solving a linear-quadratic optimization problem, which's procedure is interpreted in the following.

With substitution of the decision variable $Q \in R^{nxn}$ which is a positive definite matrix, the observer equation can be obtained as

$$\begin{aligned} \dot{\hat{x}}(t) &= (A - QC^T C)\,\hat{x}(t) + Bu(t) + QC^T y(t), \\ \hat{z}(t) &= C_z\, \hat{x}(t). \end{aligned} \tag{13}$$

The goal of the linear-quadratic optimization is to obtain the smallest L_2 – gain of the disturbance input of the system that is guaranteed to be less than a pre-specified positive constant γ_{min} , and, at the same time to increase filtering sensitivity as much as possible (Edelmayer, 2012).The algorithm, which is used iteratively reduces γuntil Q no longer has a positive definite solution. Theγ_{min} , which is reached, is within a given arbitrary small tolerance$\varepsilon > 0$.

The procedure is based on the solution of Modified Filter Algebraic Riccati Equation (MFARE). From the bounded-real lemma, we have $\|H_{\varepsilon\kappa}\|_\infty < \gamma$ if and only if there exists $Q \geq 0$ such as

$$AQ + QA^T - Q\left(C^T C - \gamma^{-2} C_z{}^T C_z\right)Q + B_\kappa B_\kappa{}^T = 0 . \qquad (14)$$

After solving (equation (14)) and getting *Q*, the filter gain matrix can be obtained as

$$K = QC^T . \qquad (15)$$

With the use of γ_{min} the detection threshold of the filter can be given as

$$\tau(C_z) = \gamma_{\min} \left\|\kappa\right\|_2 . \qquad (16)$$

It is important to note, that the failure modes, which's magnitude is smaller than that of the detection threshold, cannot be detected by the filter.

3.2 H_∞ detection filter for detection of actuator faults in the air path of diesel engines

A H_∞ detection filter has been implemented to the fault simulations performed in Matlab software. The filter was designed for a fixed operating point, but it must work properly in a certain region around the operating point even in spite oft he disturbance.

The disturbance was modelled as a fluctuating change of the engine speed caused by variable load (wind, ramp, break, etc.) that corresponds to the real driver situation. Both actuator faults were bias fault, in which the amplitude of control input signal as fault signature was increased up to 30% in the presents of the disturbance.

It is important to note that in our examination it was assumed, that only one fault appears in the system at a time.

The proposed robust fault detection filter scheme (Edelmayer, 1994), that we have implemented for the air path of diesel engines, is shown in *Fig.3* .

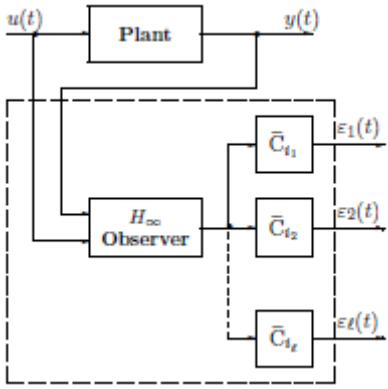


Figure 3: H_∞ detection filter scheme with different weightings C_i

The simulation of the filter responses was performed by solving the filter error system (equation (9)). The filter acts by estimating the states of the LTI model around the specified equilibrium point.

According to equation (9), the filter error system of residual generator for the two actuator faults can be written as

$$\begin{aligned} \dot{\tilde{x}}(t) &= (A-KC)\,\tilde{x}(t) + B_w w(t) + Bf_a(t), \\ \varepsilon(t) &= C_z \tilde{x}(t). \end{aligned} \tag{17}$$

The term $Bf_a(t)$ in the equation (17) can be expressed as

$$Bf_a(t) = B_{fegr}\, f_{egr}(t) + B_{fvgt}\, f_{vgt}(t), \tag{18}$$

with the matrices and fault vector

$$B_{fegr} = \begin{bmatrix} 1.6111\cdot 10^{9} \\ -1.5720\cdot 10^{10} \\ 0 \end{bmatrix}, \quad B_{fvgt} = \begin{bmatrix} 0 \\ 8.3514\cdot 10^{4} \\ -141.6484 \end{bmatrix}, \quad f_a(t) = \begin{bmatrix} f_{egr}(t) \\ f_{vgt}(t) \\ 0 \end{bmatrix}. \tag{19}$$

B_{fegr} is a fault direction matrix for the fault f_{egr}(t), which is composed from the first column of the *B* matrix corresponding to the first row in the fault vector $f_a(t)$. B_{fvgt} is a fault direction matrix for the fault f_{vgt}(t), which is composed

from the second column of the *B* matrix corresponding to the second row in the fault vector $f_a(t)$.

As previously mentioned both actuator faults were bias fault modelled, in which the amplitude of control input signal as fault signature was increased up to 30%. Thus the according failure modes are written

$$\begin{aligned} f_{egr}(t) &= 0.3 \cdot u_{egr}(t - t_{jegr}) , \\ f_{vgt}(t) &= 0.3 \cdot u_{vgt}(t - t_{jvgt}). \end{aligned} \tag{20}$$

where $u_{egr}(t)$ and $u_{vgt}(t)$ are the input signal of EGR-Actuator and VGT-Actuator respectively. t_{jegr} and t_{jvgt} are the time instants when the fault $f_{egr}(t)$ and $f_{vgt}(t)$ appear respectively. $f_{egr}(t) = 0$, if $t < t_{jegr}$ and $f_{vgt}(\mathrm{t}) = 0$, if $t < t_{jvgt}$. If$f_{egr}(\mathrm{t}) = 0$ and $f_{vgt}(\mathrm{t}) = 0$, the plant is assumed to be fault free. Assume, however, that only one fault appears in the system at a time.

In order to show the effect of the weighting matrices on the H_∞ bound, we start with $\gamma_{min} = 3.6 \cdot 10^5$(that corresponds to a very high value), of the corresponding setting of estimation weighting matrices obtained as

$$C_Z = \begin{bmatrix} 10^4 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{21}$$

To get the optimal value of the filter gain we must perform a γ-iterations on the Riccati equation (equation (14)). The filter gainis given as

$$K = \begin{bmatrix} 0.0030 & -0.0005 & 0 \\ -0.0005 & 0.0111 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{22}$$

The plots corresponding to solutions of the filter error system of residual generator (equation (17)) and a comparison between the states of faulty - and faultless system are shown in *Fig. 6* and *Fig. 7*.

In order to prove the filter performance for disturbance attenuation, the transfer function of the disturbance using the filter gain *K* is described as

$$\mathrm{H}_{\varepsilon\omega}(\mathrm{s}) = C_Z(\mathrm{sI}-\mathrm{A}+\mathrm{KC})^{-1}\mathrm{B}_{\omega}, \tag{23}$$

the transfer functions calculated from the fault of the EGR-Actuator and VGT-Actuator to the weighted error residual $\varepsilon(t)$of the filter, respectively

$$\mathrm{H}_{\varepsilon egr}(\mathrm{s}) = C_Z(\mathrm{sI}-\mathrm{A}+\mathrm{KC})^{-1}\mathrm{B}_{fegr}, \tag{24}$$

$$\mathrm{H}_{\varepsilon vgt}(\mathrm{s}) = C_Z(\mathrm{sI}-\mathrm{A}+\mathrm{KC})^{-1}\mathrm{B}_{fvgt}, \tag{25}$$

The magnitude of the above mentioned transfer functions can be seen in *Fig. 4.*

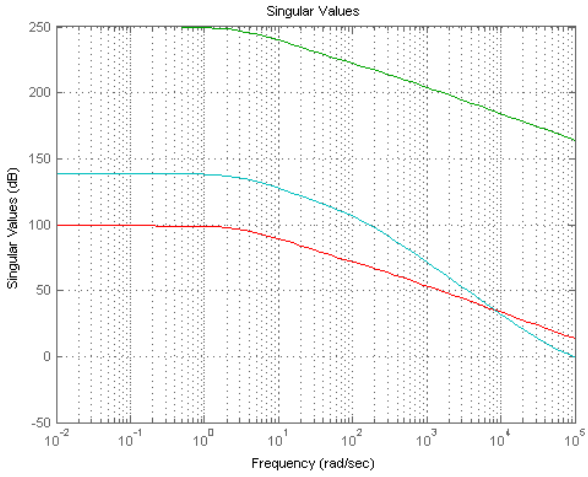


Figure 4: The magnitude (maximal singular values) of transfer functions: $T_{\varepsilon\omega}$ (red line), $T_{\varepsilon fegr}$ (green line), $T_{\varepsilon fvgt}$ (cyan line) for estimation weighting C_z

Following the same procedure, we choose the estimation weight in special form as

$$C_Z^{*} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 25 \end{bmatrix}. \tag{26}$$

By performing γ-iterations again in the case of C_z^*, the optimal value of $\gamma_{min} = 4.824$ is obtained. The corresponding solution MFARE for the decision matrix and throughout the filter gain is given as

$$K^* = \begin{bmatrix} 158.273 & -46.478 & -0.0001 \\ -46.478 & 692.143 & 0 \\ -2.973 & -0.215 & 0 \end{bmatrix}. \quad (27)$$

The magnitude of the transfer functions calculated using the filter gain K^* is shown in *Fig. 5.*

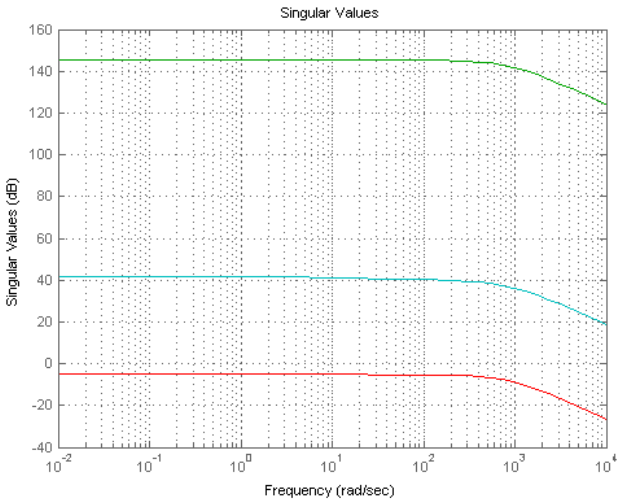


Figure 5: The magnitude (maximal singular values) of transfer functions: $T_{\varepsilon\omega}$ (red line), $T_{\varepsilon fegr}$ (green line), $T_{\varepsilon fvgt}$ (cyan line) for estimation weighting C_z

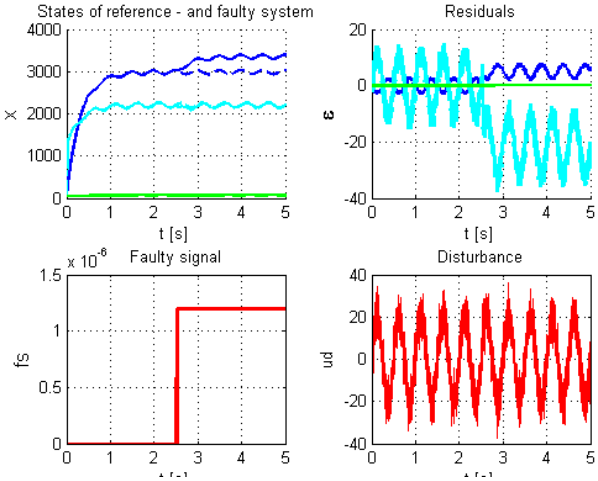


Figure 6: EGR-Actuator bias fault occurring at t= 2s, in the presence of engine speed disturbance
States and residuals: X_1 and ε_1 (blue line), X_2 and ε_2 (cyan line), X_3 and ε_3 (green line) estimation weighting C_z^*. Dashed lines: States of the faultless LTI system.

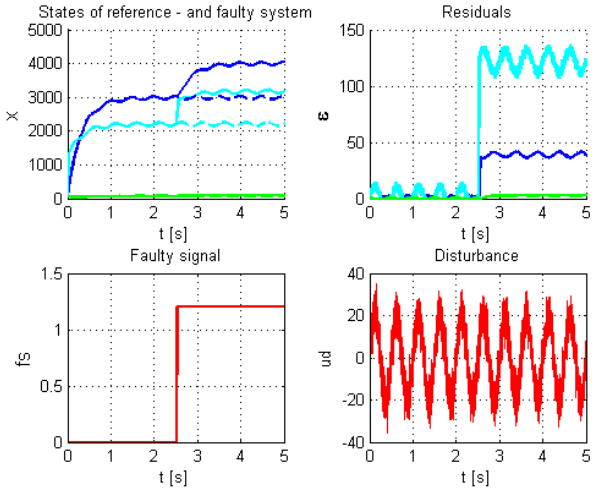


Figure 7: VGT-Actuator bias fault occurring at t= 2s,in the presence of engine speed disturbance
States and residuals: X_1 and ε_1 (blue line), X_2 and ε_2 (cyan line), X_3 and ε_3 (green line) estimation weighting C_z^*. Dashed lines: States of the faultless LTI system.

4. Conclusion

From the simulation results of the H_∞-Filter residuals applied for the air path model of diesel engine, it can be concluded that the filter gives the estimates of two actuator faults, namely bias fault of the EGR-Actuator and VGT-Actuator in the presence of disturbance. The disturbance was modelled as a fluctuating change of the engine speed caused by variable load.

It was shown that the estimation weight has an impact to magnitude of transfer function. By appropriate choice of estimation weight the filter sensitivity can be achieved. By selecting estimation weight C_z^*, as it can be seen in *Fig.5*, the sensitivity is nearby constant in a wide frequency band $\left(10^5 \frac{rad}{s}\right)$. The results indicate that the Filter gain K^*ensures at least min. 50 dB of separation (signal-to-noise-ratio, SNR) between the VGT-Actuator fault and the disturbance effect. From this it can be concluded that this sensitivity is normally enough to detect both the EGR - and VGT-Actuator faults in the presents of the disturbance.

In opposition to them, in the case of the estimation weighting of C_z, the filter does not provide enough sensitivity, because the transfer function of the VGT-Actuator fault decreases faster at the frequency of $10^2 \frac{rad}{s}$ and it even reaches the magnitude of the disturbance at the a frequency of $10^4 \frac{rad}{s}$ as it is shown in *Fig.4.*

If a fault occurs in the EGR-Actuator, the weighted residual ε_1 is strongly and ε_2 is slightly changed as it is shown in *Fig. 6*. The residuals correspond to the state x_1 and x_2, the manifold pressure and exhaust pressure, respectively. It is similar in the case of the VGT-Actuator as it can be seen in *Fig. 7*.

From these it follows that both faults can be detected in spite of worst–case disturbance. As both fault signals do not represent the faults in the residual signal separately, fault isolation with using a single filter could not be guaranteed.

The results confirmed that the H_∞ fault detection filtering is a suitable method for robust fault detection of actuator faults in the air path of diesel engine at the chosen operating point. For further investigation, according to our concept, for extending a filtering procedure for the entire engine operation range, a set of local LTI models is to be created for different operating points and from these a corresponding filter gains can be calculated (Gain scheduling).

References

[1] Jung, M., "Mean-value modelling and robust control of the air path of a turbocharged diesel engine", Ph.D Dissertation, University of Cambridge, 2003.

[2] Herceg, M., "Nonlinear Model Predictive Control of a Diesel Engine with Exhaust Gas Recirculation and Variable Geometry Turbocharger", Diploma Thesis, Slovak University of Technology in Bratislava, Bratislava, 2006.

[3] Ceccarelli, R., "Model-based fault detection in Diesel engines air-path", Ph.D Dissertation, Institut National Polytechnique de Grenoble, Grenoble, 2012.

[4] Boulkroune, B., Djemili, I., Aitouche, A., Cocquempot, V., "Robust nonlinear observer design for actuator fault detection in diesel engines", *International Journal of Applied Mathematics and Computer Science.*, Vol. 23, No. 3, pp. 557–569, 2013.

[5] Sangha, M. S, Yu, D. L., Gomm, J. B., "Sensor fault detection, isolation, accommodation andunknown fault detection in automotive engine using AI", *International Journal of Engineering, Science and Technology*, Vol. 4, No. 3, pp. 53–65, 2012.

[6] Guermouche, M., Benkaci, M., Hoblos, G. and Langlois, N., "Sensor fault detection and isolation in diesel air path using fuzzy-ARTMAP neural network", *IEEE 10th International Conference on Networking, Sensing and Control (ICNSC)*, pp.728-733, Evry, France, 2013

[7] Edelmayer, A., "Fault detection in dynamic systems: From state estimation to direct input reconstruction", Universitas-Győr Nonprofit Kft., Győr, 2012.

[8] Edelmayer, A., Bokor, J., Keviczky, L., "An H_∞ Filtering Approach to Robust Detection of Failures in Dynamical Systems", in Proc. 33th Annual Decision and Control, Conf.,pp. 3037–3039., Buena Vista, USA, 1994.

[9] Edelmayer, A., Bokor, J., Keviczky, L., "An H_∞ Filter Design for Linear Systems: Comparison of two Approaches" IFAC 13th Triennial World Congress, San Francisco, USA, 1996.

[10] Douglas, R. K., Speyer, J. L., "Process and Sensor Noise Robustness in Detection Filter Design", Proc. Amer. Contr. Conf. , pp. 2274-2279, Boston, MA, 1991.

[11] Yaesh, I., Shaked, U., "Game Theory Approach to Optimal Linear State Estimation and Its Relation to the Minimum H_∞ -norm Estimation", *IEEE Trans. Aut. Control*, AC-37(6), pp. 828–831, 1992.

[12] Kulcsár, B., "Design of Robust Detection Filter and Fault Correction Controller", Ph.D Dissertation, Budapest University of Technology and Economics, Budapest, 2005.

[13] Boyd, S., Ghaoui, L. E., Feron, E., Balakrishnan, V., "Linear Matrix Inequalities in System and Control Theory", SIAM, Philadelphia, 1994.

[14] Gahinet, P., Nemirovski, A., Laub, A. J., Chilali, M., "LMI Control Toolbox for Use with Matlab", The Math Works, Natick, Massachusetts, 1995.

[15] Ding, S. X., "Model-Based Fault Diagnosis Techniques", Second Edition, Springer-Verlag, London, 2013.

[16] Isermann, R., "Fault-Diagnosis Applications", First Edition, Springer-Verlag, Berlin, Heidelberg, 2011.

[17] Isermann, R., "Fault-Diagnosis Systems", First Edition, Springer-Verlag, Berlin, Heidelberg, 2006.

[18] Chen, J., Patton, R.J., "Robust Model-Based Fault Diagnosis for Dynamic Systems", First Edition, Springer Science & Business Media, New York, 1999.

[19] Patton, R., Frank, P., and Clark, R., "Fault Diagnosis in Dynamical Systems: Theory and Application", First Edition, Prentice Hall, New Jersey, 1989.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 7 (2015) 23-35

The Design of an Automated Plasma Diagnostic System and its Applications

Nimród KUTASI, Lajos KENÉZ, Emőd FILEP, István SZÖLLÖSI, László JAKAB FARKAS

Department of Electrical Engineering, Faculty of Technical and Human Sciences,
Sapientia-Hungarian University of Transylvania, Tg. Mureş,
e-mail: {kutasi; l_kenez; efilep; iszollosi; jflaci}@ms.sapientia.ro

Manuscript received September, 2015; revised June 10, 2016

Abstract: The paper deals with the design of a new plasma diagnostic system consisting of a microcontroller based power supply for automatic control of the measurement using Langmuir probe, the communication interface of the power supply and the signal processing program. The utility of the designed plasma diagnostic system is presented through measured U-I curves and a method to measure the relaxation time of the plasma.

The main motivation for the design of the automated plasma diagnostic system was given by our research in different methods of plasma nitriding of steels. During the nitriding process, it is important to get information about the plasma, which is the medium where the heat-treatment process is going on. The new diagnostic system is able to perform measurements during the nitriding process, thus we can analyze the plasma at different temperatures and gas mixtures. The obtained voltage-current curves are recorded and transmitted to the computer, where further signal processing is performed. The paper presents the design of the power supply, the measurement results and the developed signal processing software. Further on, the estimation of the relaxation time of the plasma is presented, based on local measurements using Langmuir probe.

Keywords: plasma diagnostic, Langmuir probe, power supply, microcontroller, data acquisition, IIR filtering, plasma relaxation time.

1. Introduction

In the last three years we have succeeded to rebuild and operate a linear plasma reactor for Direct Current and Active Screen plasma nitriding. The design of the system and the first results were reported in [1],[2], where we

The results of this study were partially presented at the 5th International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics 2015.

focused mainly on the plasma nitriding technology. Beyond the nitriding technologies, it is very important to study the behavior of the plasma, thus the vacuum chamber contains a feedthrough for a Langmuir probe and we designed a measurement device for accurate local plasma diagnostic. Embedding the vacuum diagnostic software in the heat treatment process control software, enables the local plasma diagnostic during the heat treatment process, opening new possibilities in the high temperature plasma diagnostic. To be able to work at high temperatures, we had to rebuild the Langmuir probe, the thermal isolation of the feeding cable was improved and the probe was introduced in a ceramic isolator, thus it can be used in the500-600ºC temperature range.

In the following sections we present the design of the local plasma diagnostic system, starting with a short description of the plasma diagnostic techniques, followed by the design of the power supply, the measurement results and the post processing of the measurements.

2. Plasma diagnostic using Langmuir probe

There are two plasma diagnostics techniques which can be used to explore plasmas [3]. Global techniques are based on the study of the electromagnetic radiation leaving the plasma and provide information on the whole plasma. Local techniques are based on the study of the voltage-current curves of different electrostatic probes inserted in the plasma, which provide local information on the plasma. Both techniques have difficulties, advantages and disadvantages. We designed a local plasma diagnostics setup because we proposed to study the distribution and evolution of local plasma parameters during the nitriding process of steel samples. We use a cylindrical shaped 0,4 mm diameter, 3 mm long probe made of tungsten. The probe is movable, the feedthrough is provided with a computer controlled stepper motor to enable precise control of the probe position. Information on the local plasma parameters can be obtained from the voltage-current curve of the probe. Therefore it is important to ensure proper conditions for these measurements. We designed a complex, computer controlled DC power supply to bias the probe and measure the currents. The power supply is suitable to bias traditional single probes, double probes and emissive probe. All parameters can be adjusted on-demand in the user interface of the program designed for probe measurements. Probe voltage-current characteristics bear information on local plasma parameters. There are different data processing methods which yield information on local electron density, electron temperature (energy). During nitriding these parameters are influenced by the pressure, discharge voltage, temperature, gas mixture and anode-cathode distance.

3. The design of the power supply

The main characteristics of the DC power supply for plasma diagnostics has been defined as follows:

- Output voltage range: -125VDC…+125VDC
- Output current: 0…70mA
- Programmable voltage step resolution 0.01V
- Current measurement accuracy: 1μA
- Output voltage step time: min. 50ms
- Communication interface: RS232 or USB.

The schematic diagram of the designed measurement device is presented in *Fig.1*. A precision linear power supply feeds the low voltage circuits and the measurement modules, while a digital signal controller implements the control and measurement tasks.

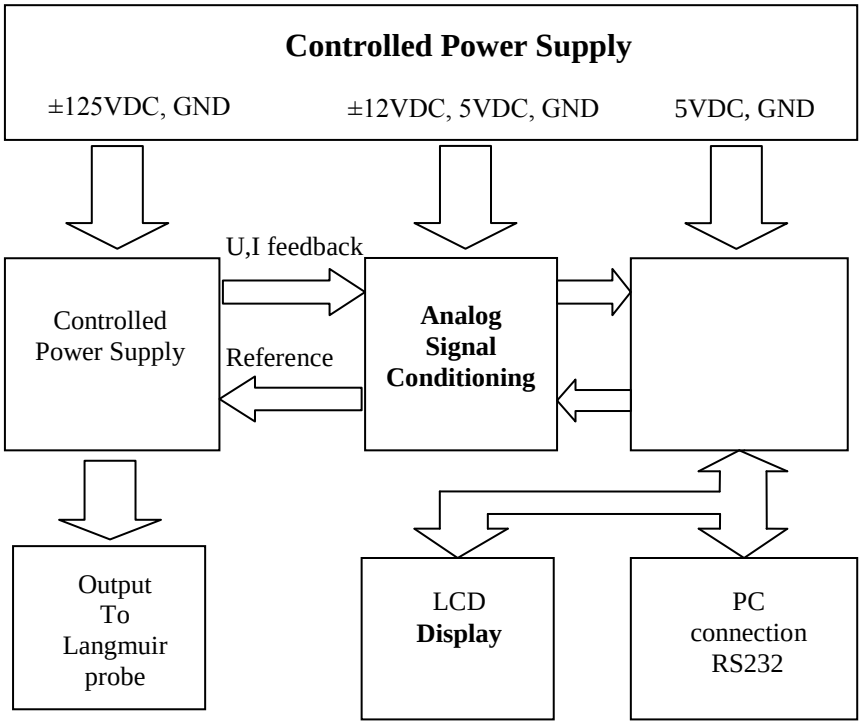


Figure 1: Block diagram of the measurement device

According to *Fig. 2*, the Langmuir probe is supplied with a voltage defined by the measurement software. The output voltage is controlled, thus the probe

voltage is kept constant during the measurement. The actual voltages and actual currents are measured with the digital signal controller.

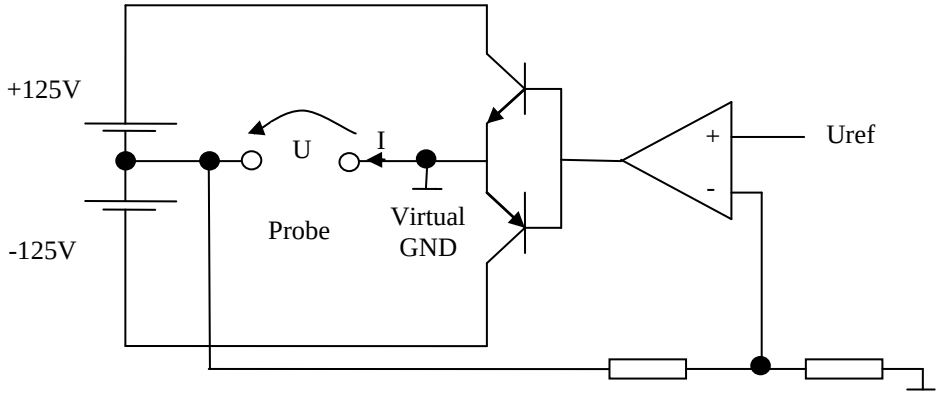


Figure 2: The measurement principle.

During the development several simulations were performed. In *Fig. 3* the pSpice model of the measurement circuit is presented

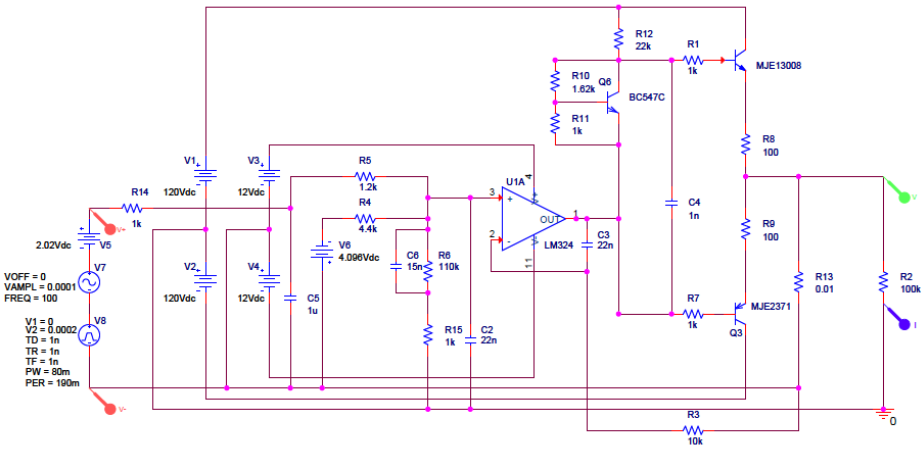


Figure 3: pSpice model of the output voltage control circuit

The most important simulation results are presented in *Fig. 4* and *Fig. 5*. In the first figure the linearity of the output voltage versus the reference voltage is

presented. It can be observed that the output voltage is linear in the whole range, thus the control voltage has a unique and linear mapping on the output voltage.

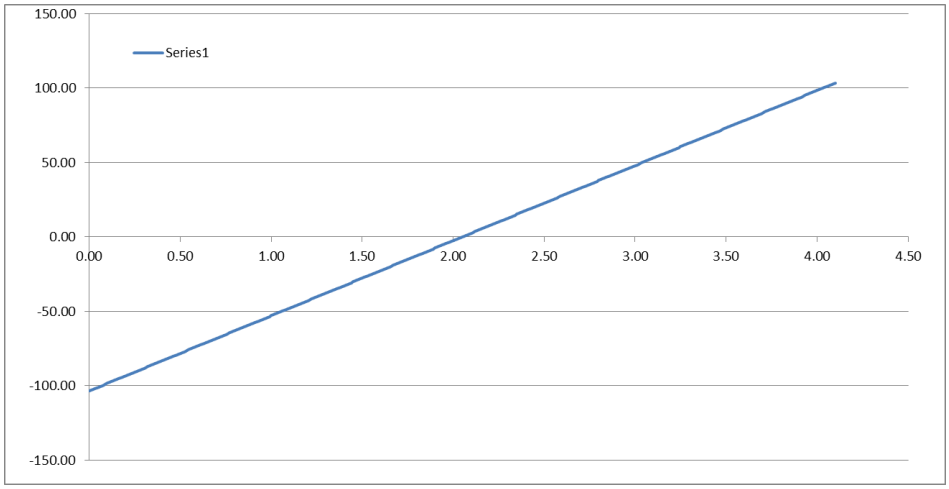


Figure 4: Output voltage [V] versus the control voltage [V]

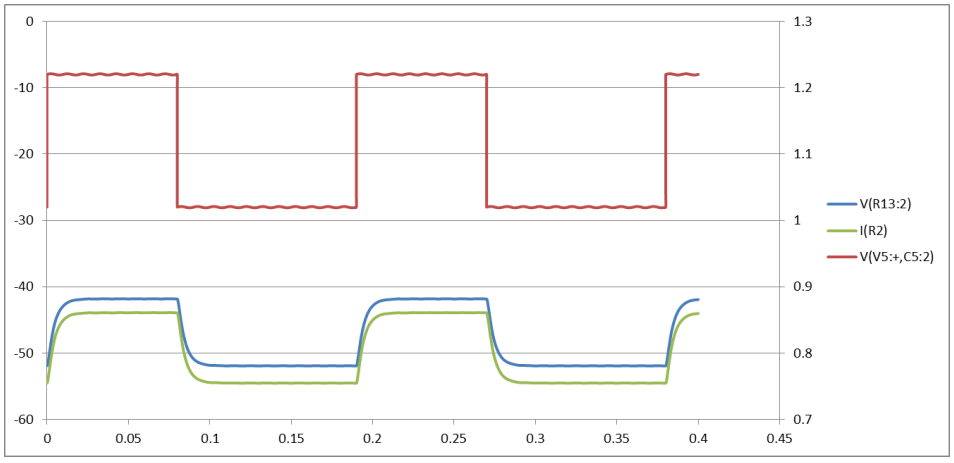


Figure 5: Control and output voltage [V] versus time [s] of the power supply in case square-wave control voltage

In *Fig. 5* the step response of the voltage regulator is presented. The time constant of the closed loop system is about 10 ms, thus the measurement of the output voltage and output current has to start at least with 10 ms delay after the step change on the output voltage. This time constant is acceptable compared to the minimum 50 ms time step defined for the measurement.

4. Practical implementation. Measurement results

The realized power supply is presented in *Fig. 6*. In the front view the operator interface is presented, with an LCD display and the pushbuttons, enabling standalone working of the power supply. In the internal construction one can see the separate modules presented in the block diagram (*Fig. 1*), namely the linear power supply, the analog interface circuit and the control board with the DSP processor.

Figure 6a: Front view of the power supply

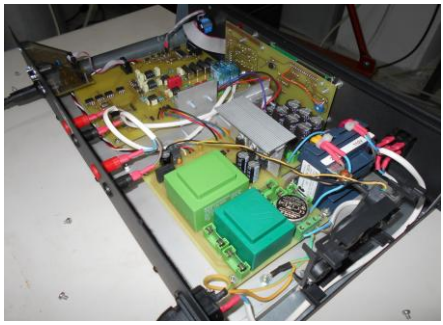
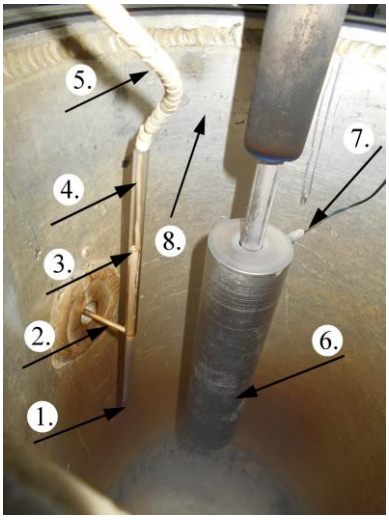


Figure 6b: Internal construction of the power supply

For the local plasma diagnostic the Langmuir probe has been improved, as it can be seen in *Fig. 7*.



1. Cylindrical probe made from wolfram(3mm long, ∅=4mm)
2. Probe holder, allowing radial movement
3. Ceramic isolator introduced in a glass tube
4. Heat-resistant silicon isolated wire
5. Ceramic rings for thermal isolation of the wire
6. The cathode of the discharge
7. Thermocouple for measurement of the cathode's temperature
8. Grounded anode tube, this is basically the water cooled wall of the reactor

Figure 7:The Langmuir probe installed in the linear plasma reactor

The first measurements were made in nitrogen-hydrogen plasma, where the cylindrical probe was heated up to 556ºC and the probe was placed at a distance of 6 centimeters from the cathode. The voltage step applied to the probe was 0.1V and the sampling period was 100ms. The result is depicted in *Fig. 8.*

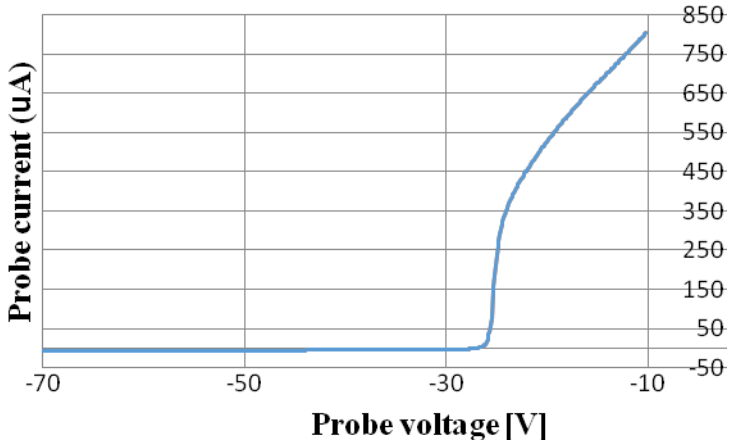


Figure 8: Voltage-current characteristic of the probe measured with the new power supply

For the automated measurements a LabWindows CVI based software has been developed, which has an easy to use graphical user interface (GUI), see *Fig. 9*, and a communication module with the power supply. On the user interface one can set the parameters of the measurement, the measurement results are displayed on a strip chart. In *Fig.10* a set of U-I characteristics is shown, the distance of the probe from the cathode varying from 30 mm to 98 mm.

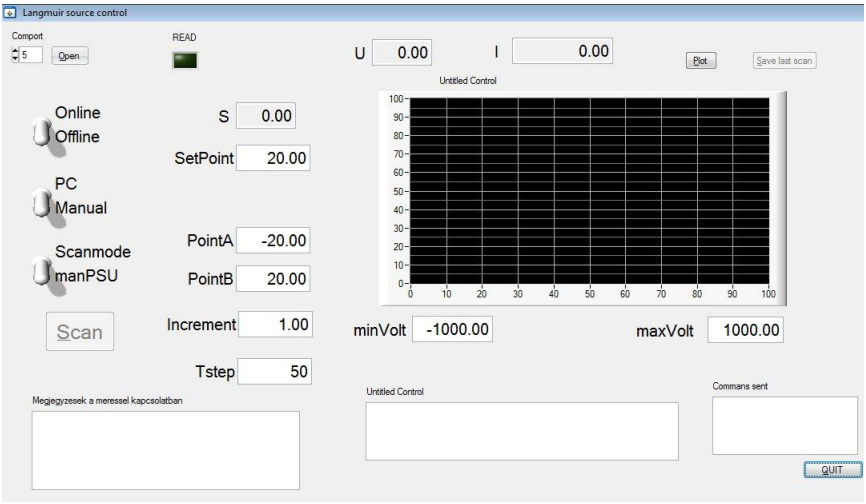


Figure 9: TheGUI of the measurement software

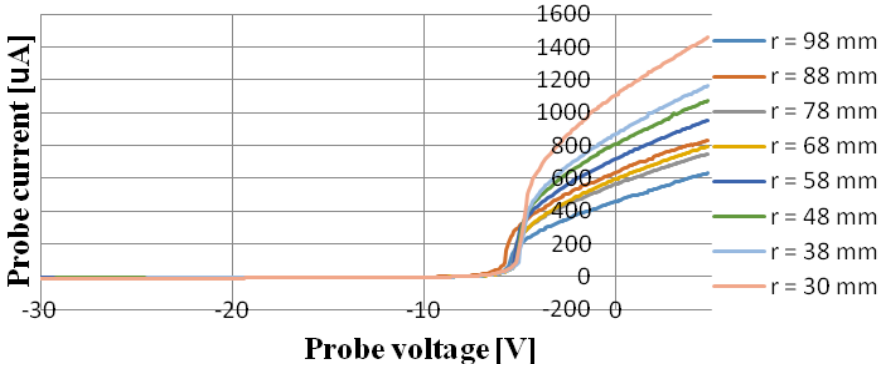


Figure 10: Set of U-I characteristics at 175ºC cathode temperature and different probe distances from the cathode

5. Post processing of the measured data, signal filtering

Although the recorded curves (*Fig. 8, Fig. 10*) look smooth, calculating the slope of the curves in different points, the result was unsatisfactory and very noisy. There are two main reasons for the noise of the derivative. The first is the small measurement noise on the curves, and the other comes from the scale changing during the measurement. When the power supply changes the measurement scale, a small step change on the curve can be observed. To be able to use the measurement result, a digital filter was designed to smooth the measured signals [4], [5]. A satisfactory result was obtained with a fourth order Elliptical IIR filter, designed with the sampling frequency of 10 Hz, and the cutoff frequency of 0.4 Hz, see *Fig. 11*.

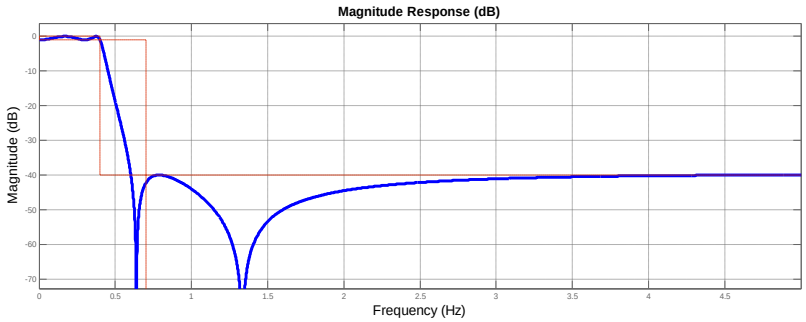


Figure 11: The magnitude response of the fourth order elliptical IIR filter

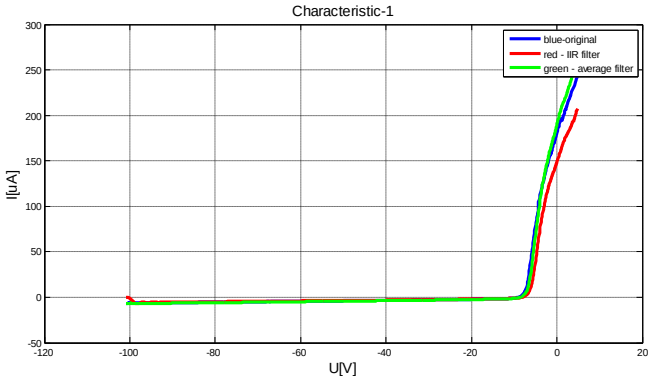


Figure 12: The result of the digital filtering on recorded U-I characteristics

The result of the first filtering of the U-I curve is depicted in *Fig. 12*. The red curve results form the IIR filtering.A phase shift and a magnitude attenuation of the filtered signal can be observed. To assure an accurate interpretation of the derivative of the recorded U-I curves, the phase shift introduced by the digital filter has to be compensated. The phase-shift introduced by the filter can be obtained comparing the derivative of the unfiltered curve with the derivative of the filtered curve. Using a simple shift operation on the filtered derivative, the phase-shift can be eliminated.

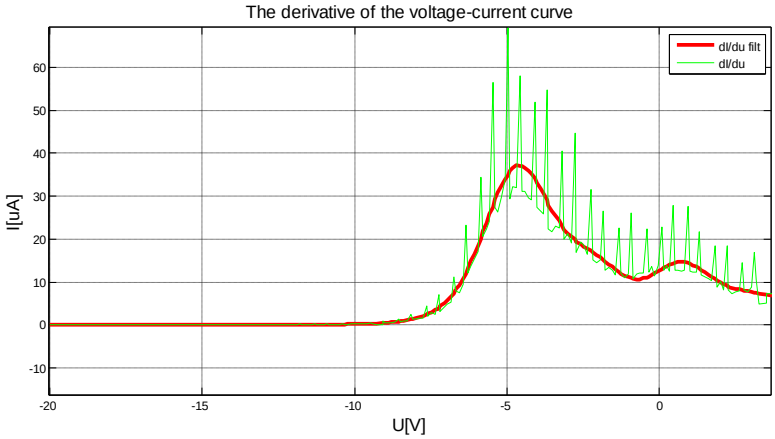


Figure 13: The derivative of the current versus the voltage and the filtered derivative

In *Fig. 13* the derivative of the recorded characteristics is presented. It is imoprtant to notice, that the derivative of the filtered curve is noisy, as it can be seen on the green curve. Appliyng again the same digital filter to the derivative and adjusting the delay caused by the phase shift of the filter, the obtained characteristic is usable for the plasma diagnostic purposes.

6. Estimation of plasma relaxation time

In the linear non-isotherm plasma reactor the plasma is created using direct-current gas discharge and it is necessary to study the physical processes on the surface of the probe (having anodic, cathodic or floating potential), which has direct contact with the laminar streaming gas mixture. Such complex processes are: the well-known direct-current plasma nitriding (DCPN), the most recently developed active-screen plasma nitriding (ASPN) or the anodic nitriding, where

the active screen is substituted with a hollow cathode. The control of these complex processes assumes deep understanding of the local plasma parameters, like the electron density, electron temperature, floating potential. All these parameters are measured using Langmuir probes (*Fig. 7*, *Fig. 8*, *Fig. 12*). As it is shown in *Fig. 12*, the usual measurement range is 120-150 Vdc, and the resolution is 0.5-0.1 V. The number of the measurements is high, but an automated diagnostic system can handle such a fast measurement when the measurement period is correlated with the properties of the plasma for all characteristic regions (ion-current, electron-current and saturated electron-current). Nearby the probe inserted in the plasma, there is an equilibrium distribution of the particle. This distribution depends mainly on the potential of the probe. When the potential is changed the distribution of the particles change and the spatial arrangement of the particles change. The time necessary for a new equilibrium state formation is called plasma relaxation time. Furthermore, the distance from the probe, where the perturbing effect of the probe cannot be sensed is called relaxation distance [6]. To be able to perform an accurate Langmuir probe measurement, it is necessary to know the relaxation time of the plasma, to be able to set the minimum step time of the voltage applied to the probe. In the followings the measurement of the relaxation time is presented using the plasma diagnostic system and an oscilloscope (*Fig. 14*).

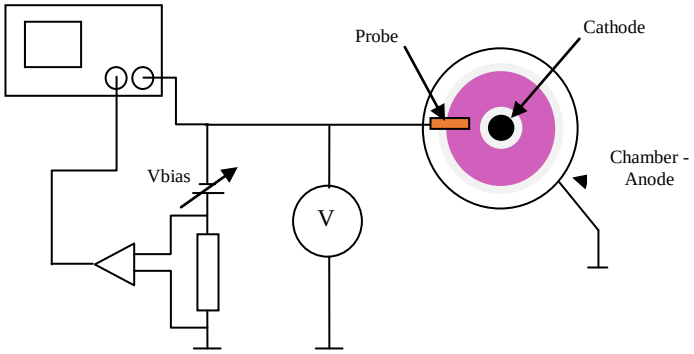


Figure 14: Schematic principle of the relaxation time measurement

Using the automated plasma diagnostic system, during the measurements the bias voltage has been changed in steps and the probe current was recorded using a shunt resistor. The voltage and current waveforms were measured with a digital oscilloscope after the first RC low-pass filter. The effect of the low-pass filter is not significant due to the high cutting frequency (10 kHz). The plasma

relaxation time is estimated similarly to the time constant of a first order linear system. From the step response of the system, one can determine the time constant using the graphical method. In *Fig.15*t he measurement results are presented.

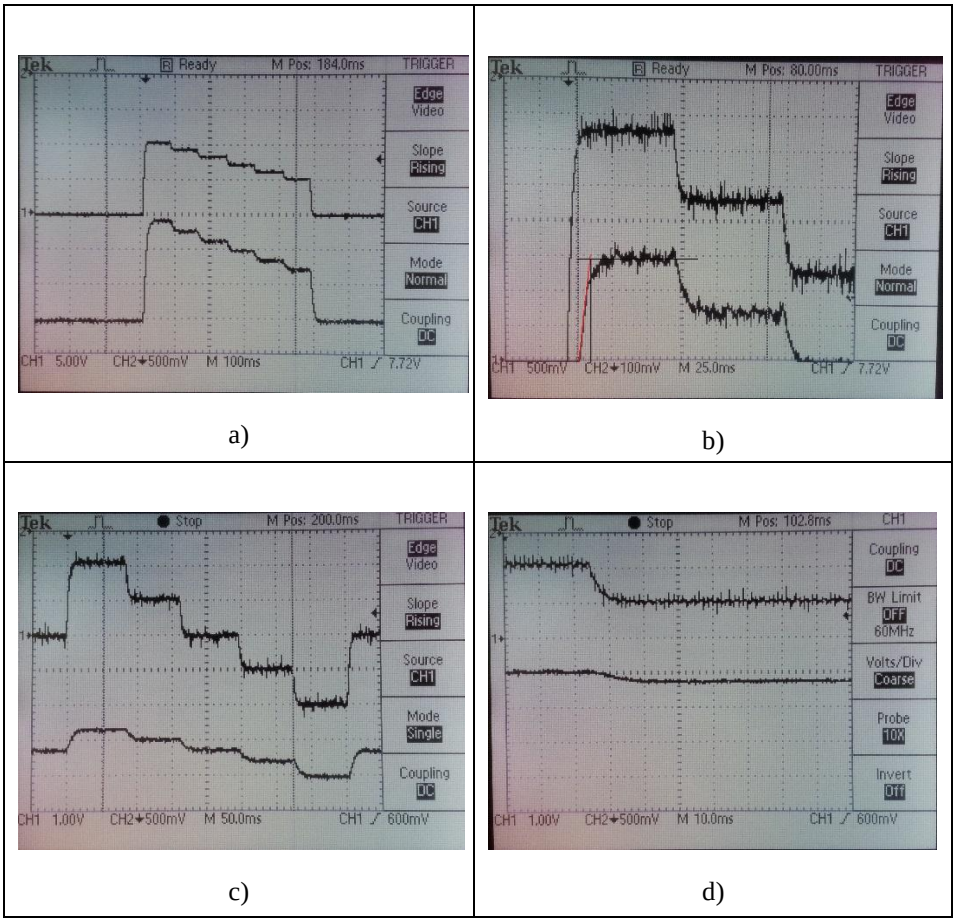


Figure 15: Voltage and current waveforms recorded with the oscilloscope
a) a measurement sequence in the electron-current region. b) the same measurement with higher resolution, the estimation of the time constant is shown. c) Measurement sequence spanning over both electron and ion-current regions. d) The measurement in a 10ms time resolution. The measurements were performed at 200°C of cathode temperature and at 6cm from the cathode in radial direction

According to *Fig. 15.b*,the time constant of the current waveform is about10 ms, and the settling time is about three times the time constant (30 ms), thus an accurate measurement of the current signal has to start at least with 30 ms after the voltage change has been applied. Taking into account the time constant of the voltage controller, which is also about 10 ms (*Fig.5*), the initially defined minimum step time of 50 ms for the applied voltage is correct. In conclusion, the plasma relaxation time is in the range of 10-20 ms in the actual discharge conditions. To have a clear view about the relaxation time in the whole radial region of the plasma (temperature dependence), further measurements will be performed.

7. Conclusion

In the paper we described the design of an automated local plasma diagnostic system. Starting from the design of the power supply we described in detail the steps we made to obtain the desired results. The recorded curves need post processing, thus we designed a software for filter the measured values and the derivative of the curves. The obtained results will be used to collect information about the plasma during different plasma nitriding processes. Using the automated measurement we succeeded to estimate the plasma relaxation time.

Acknowledgements

This work has been supported by the Institute for Research Programs of Sapientia University.

References

[1] Kutasi, N., Kenéz, L., Filep, E., Kelemen, A., Mátyási, Sz.,"Pulsed power supply design for DC and Active Screen Plasma Nitriding", *MACRO2013 International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics Sapientia University*, pp. 115–122, Tg. Mureş 2013.

[2] Kenéz, L., Kutasi, N., Filep, E., Jakab, L. F., Szőcs, I. A.,"Heat Treatment of 16MnCr steel in a linear non-isoterm plasma reactor", *Acta Universitatis Sapientiae, Electrical and Mechanical Engineering*, vol.5, pp.61–72, 2013.

[3] Kenéz, L., Zsakó, Z., Filep, E.,"Automation of plasma diagnostics measurements performed in a non-isotherm plasma reactor", *Studia Universitatis Physica*, LIII/1 2008.

[4] Rorabaugh, C.B., "Digital Filter Designer's Handbook".TAB Books Division of McQraw-Hill, Inc. Blue Ridge Summit, PA 17294–0850.

[5] Ingle, V. K., Proakis, J. G.,"Digital Signal Processing Using MATLAB" (Bookware Companion), Thomson-Engineering; 2 edition August 10, 2006.

[6] Kaptzov, N. A., "Electronica" Ed.Tehn. Bucureşti, p.322, 1956.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 7 (2015) 36-61

Assessing the Efficiency of a P2P Video Client

Miklós MÁTÉ, Rolland VIDA

Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics, Budapest,
e-mail: {mate; vida}@tmit.bme.hu

Manuscript received February 11, 2016; revised May 5, 2016

Abstract: Video on Demand systems are increasingly popular sources of entertainment, rapidly replacing conventional television. To meet the high bandwidth and fault tolerance requirements, most VoD content providers are utilizing distributed data delivery technologies, like Peer-to-Peer schemes. The client programs of such systems usually restrict the P2P scheme into a sliding window to ensure timely arrival of the video segments. In this paper we analyze the performance of a generic P2P VoD client. We develop analytical connections between the state descriptor variables, and provide conditions for the P2P scheme to be self-sufficient. We also examine the effect of limited downlink capacity.

Keywords: Peer-to-Peer, Video-on-Demand, Steady-state, Self-sustainability.

1. Introduction

The traffic transferred through the Internet is growing exponentially, mostly due to the increasing popularity of video content, and the demand for higher video quality increases the bandwidth-demand of the individual streams as well. Content providers have always used distributed server architectures to decrease network load and combat the latency issues by moving the content close to the consumers, but the increasing complexity of creating and maintaining such systems motivates the search for entirely new solutions. A scheme that offers great performance, error-resilience, and requires low maintenance is to utilize the clients as content sources for other clients. The downside of such Peer-to-

Peer (P2P) schemes is that the access networks are typically not optimized for uplink traffic.

The performance of Video-on-Demand (VoD) systems that use P2P technology is heavily influenced by the efficiency of the clients. Therefore, it is important to analyze the performance impact of the details of the P2P technology used in the clients. P2P VoD clients have to provide timely arrival of the video segments, or the playback cannot proceed smoothly; practically, this means that the general P2P schemes must be modified for use in a VoD system. Even though the performance of the general P2P scheme is known, these modifications can alter its properties to great extent.

This model of the P2P clients is based on the BitTorrent protocol, but is not specific to it. The clients use a sliding window of W segments, called P2P window, which is placed somewhere ahead of the playback position, and the P2P segment retrieval scheme is restricted to operate only within the P2P window. We consider two window placement schemes, and two segment selection schemes within the P2P window.

The most important contributions of this paper are the analytical results for the connections between the state descriptor parameters, and the criteria for self-sustained operation of the P2P scheme.

The rest of the paper is organized as follows. In the next section we give an overview of the related work in this field. In Section 3 we describe the model of the P2P client in detail. In Section 4 we construct and analyze the state-space description of the P2P download window. The sections after that each deal with a specific property of the system: the number of Full segments in Section 5, the number of new segment downloads in Section 6, the number of Empty segments in Section 7, and the advance speed of the window in Section 8. Section 9 gives formulas for the probability of missing a segment, and criteria for avoiding that entirely, if possible. Section 10 presents our findings about the performance of the client when the downlink capacity is limited. Finally, Section 11 draws the conclusions.

2. Related work

Peer-to-peer networks gained high popularity in the last decade. They can provide decentralized content indexing, data distribution, or both of them. Perhaps the best known P2P protocol is BitTorrent [4]. Its popularity can be attributed to its relatively simple design, efficient and flexible operation, and its open specification. It is also extremely resilient against network and node failures, which certainly helped its adoption as a distribution method of large files. It divides the content into small pieces, called chunks, which are exchanged among the clients interested in the content: each participant is a

downloader and an uploader at the same time. Those who have all pieces, and therefore are not downloading anymore, are called *seeders*, while the others are the *leechers*.

The incentive to upload is raised through the peer selection scheme of BitTorrent. It uses the tit-for-tat scheme, which is the best known solution to the repeated prisoner's dilemma [1]. In this scheme the clients prefer uploading to other clients who were willing to upload recently, punish denial with denial (this is called the choking mechanism), but are quick to forgive (optimistic unchoke). It is possible to free-ride in BitTorrent, and to build false reputation [6], but there are also some proposed optimizations to avoid that [14]. To maximize the throughput by eliminating bottlenecks, BitTorrent clients try to download the rarest chunks first. This scheme efficiently equalizes the availability of the chunks within the content.

BitTorrent has been thoroughly analyzed in the research community, several measurement studies have been conducted [9], analytical models have been constructed to explain its scalability [10], and these models were then validated by additional measurements [5].

However, despite the numerous advantages of BitTorrent, it is not directly suitable for video delivery, because the rarest-first chunk selection policy prevents the playback until a significant portion of the video is downloaded. To alleviate this issue the chunk selection scheme of BitTorrent must be modified to provide more-or-less in-order download policy. The usual approach is to introduce a download window, and only allow BitTorrent downloads within the window [11]. Some proposals let the client select chunks from outside the window with a given probability [12], and others use an exponentially weighted priority instead of a window [11], which may increase the efficiency of data sharing, but they also increase the chance of a buffer underrun. For this simulation study a closed download window seemed optimal, because it provides the strongest guarantee for timely arrival, and it is easier to examine analytically.

Numerous papers have analyzed the efficiency of P2P video distribution systems analytically. There are two research tracks that are important for us. In [8] the mean of the achievable throughput and the starting latency for in-order and rarest-first segment selection schemes are analyzed with a fluid model, based on the one presented in [10] for the unmodified BitTorrent protocol. In [3] there is a detailed analysis on the remaining server load and the self-sustainability of the P2P system; however, that analysis assumes strictly linear segment retrieval.

There is analysis on live video streaming systems as well, where determining the optimal buffer size [13] and the priorities for downloading each segment in the buffer [15] are the important questions. Live streaming is fundamentally

different from stored video streaming (e.g., the playback position is the same for all clients, none of them has any segments outside the playback buffer, and the missed segments are simply skipped), but their methodology is applicable.

3. Model of the P2P client

3.1. The P2P window

In this paper the only part of the client application that will be in focus is the P2P download window. The size of the window is always W segments, and the segments in the window are numbered $1 \ldots W$, where segment 1 is the closest to the playback position. In this model the P2P scheme only works inside the P2P window.

In this paper we only consider the case, where the download window is somewhere in the middle of the video. In the beginning the startup process is hard to model analytically, and it's quite different from the rest of the download process. Similarly, at the end of the video the P2P window stops, and the remaining segments are downloaded with an endgame protocol, which can be different from the main downloading mechanism.

When a client attempts to start downloading a segment of the video, it succeeds with probability p, which depends on the global state of the P2P swarm. Failing to start a download can happen for various reasons. It can happen that all the clients holding that segment are occupied, or maybe none of the other clients hold that segment yet. In the latter case, the clients revert to downloading the segment from the server, once it leaves the P2P window, and enters the fallback window, but that mechanism is outside of the scope of this study.

We assume that p is constant, which means that the P2P swarm is in equilibrium. This happens for example, when the new clients enter the system with constant intensity λ arrivals per second, and the seeders leave the system with constant intensity μ departures per second. We start with this known p, and derive formulas for the state variables of the system.

In this model the P2P window is thought of as a stochastic process, representing the statistical ensemble of several individual video downloads. This means that the variables that describe the system are random variables, and their distribution evolves with time throughout the download process. In this study the state of the individual clients will be irrelevant, and only their average behavior will be analyzed.

The downlink capacity of the VoD clients may not necessarily be infinite. In this model the downlink capacity of the clients is at most D ongoing downloads

at the same time. Throughout most of this paper $D=\infty$ will be assumed, except where explicitly stated otherwise.

3.2. State variables

Within the P2P window the segments can be in two different states. The *Full* segments are the ones, which are being downloaded, or already downloaded. The rest of the segments are *Empty*, and the goal of the download process is to convert as many Empty segments into Full as possible, before they leave the P2P window. The number of Full segments in the window is *F*, the number of Empty segments is *E*.

Since the P2P window must remain ahead of the playback position, it might happen that an Empty segment has to be shifted out. That event is a *Miss*, and the probability of missing a segment is *M*. The main goal of this study is to determine the *p* required for Miss-free operation, if it can be achieved. *Fig. 1* shows an example of the P2P window with a missed segment.

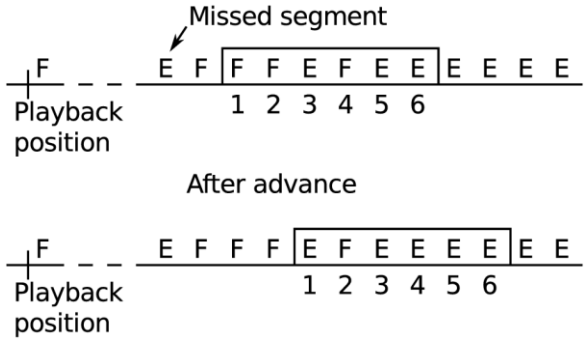


Figure 1: Illustration of the P2P window with *W*=6, and an example of Progressive window advancement

The download process works in rounds. In each round the algorithm scans the P2P window, and tries to start as many new downloads as it can. The number of newly started downloads in a round is *S*.

The number of Full segments at the beginning of the P2P window is *a*, and the advance speed of the P2P window is *A*. The connection between these two will be explained shortly.

These state variables are all non-negative integer valued random numbers. In most cases we will only consider their expected values, because their distributions are difficult to determine. The notation $\overline{X}$ will be used for the expected value of *X*, calculated as the ensemble average. Obviously, these expected values are not necessarily integer.

Time is measured in *timeframes*, one timeframe is the time it takes to play one segment of the video. Downloading a segment from a peer takes T timeframes, which is a constant in this paper.

The client initiates new downloads in rounds: it periodically runs the segment selection, and the window positioning algorithms for each timeframe.

If $T > 1$, there are usually some ongoing segment downloads when the round starts. The number of ongoing downloads is $O \leq D$.

```
function LINEAR SELECTION
    for i = 1...W do
        segment = segments[window_begin + i]
        if EMPTY(segment) then
            if CAN START DOWNLOADING(segment) then
                segment.state = "Full"
            else
                return
            end if
        end if
    end for
end function
function RANDOM SELECTION
    for i =SHUFFLE(1...W) do
        segment = segments[window_begin + i]
        if EMPTY(segment) then
            if CAN START DOWNLOADING(segment) then
                segment.state = "Full"
            else
                no action
            end if
        end if
    end for
end function
```

Figure 2: Pseudocode of the segment selection schemes

The pseudocode of the periodic window management of the P2P client is shown in *Fig. 3*. There are two points in the code marked as **A**, and **B**. The state variables that are absolute quantities will be indexed with the point, where the quantity is measured; thus, E_A is the number of Empty segments after the window was advanced, and E_B is the number of Empty segments after new downloads have been started. These indexed variables are *E*, *F*, and *O*, and *a*, while *S*, *M*, and *A* describe changes that happen during transitions between point A and point B.

```
function MANAGE P2P WINDOW
    if Progressive then
        while segments[window_begin].state is "Full" do
            window_begin++                                  ▷ Natural advance
        end while
    end if
    if window_begin < playback_position + guard_distance then
        window_begin = playback_position + guard_distance   ▷ Coerced advance
    end if
    Point A: after window advance
    Start new downloads
    Point B: before window advance
end function
```

Figure 3: Pseudocode of the P2P window management, which the client calls periodically for each timeframe

3.3. Segment selection schemes

In this paper we examine two segment selection schemes within the P2P window.

Definition 1 Linear selection: the segments are selected strictly in-order. The round ends, if all of the Empty segments in the window were started, or one segment download failed to start.

It follows from the above definition that $F = a$ with Linear selection.

Definition 2 Random selection: the Empty segments of the P2P window are tried in random order. The round ends, if starting each of them was attempted once.

The pseudocodes for the two segment selection schemes are shown in *Fig. 2*.

It is obvious that Random selection is more efficient than Linear, because it scans all Empty segments in each round, but the latter is important for performance comparison, and it has a few interesting properties as well.

3.4. Window positioning schemes

Naturally, the P2P window has to be ahead of the playback position at all times. It's also obvious that there should be a gap of at least *T* segments between them, because downloading a segment takes *T* timeframes. These constraints don't fully determine the positioning of the P2P window, however. We will examine two different schemes for positioning the P2P window: the Streaminglike scheme, and the Progressive scheme.

Definition 3 Streaminglike: maintain a constant distance; thus, $A = 1$.

Definition 4 Progressive: advance, until the first segment in the window is Empty; thus,

$$A = \begin{cases} 1 & \text{if } a_B < 1 \\ a_B & \text{if } a_B \geq 1 \end{cases}$$

The reason for the two cases with the Progressive scheme is that the playback time t_m, where the state of the window is examined, is far from the beginning of the video, and mostly constant a_B can be assumed. This suggests that if a_B is usually less than one, the playback position has caught up with the P2P window regardless of the initial distance between them. In this case $A=1$ has to be maintained in order to keep the window ahead of the playback position. On the other hand, if $a_B > 1$, the P2P window is well ahead of the playback position in t_m.

The Streaminglike scheme borrows its name from the live streaming systems, where the playback position and the P2P window (buffering in this case) are tied together. This is the most conservative window advancement scheme. The Progressive scheme pushes the P2P window forward as fast as possible, without jumping over Empty segments. *Fig. 1* shows an example for Progressive window advancement.

It follows from the definitions that these two window positioning schemes are indistinguishable, when $a_B < 1$. The following lemma also shows a trivial consequence of the definitions.

Lemma 1 With Linear segment selection and Progressive window placement $a_B = S$ and $E_A = W$.

Proof The Progressive advance scheme leaps over all the Full segments at the beginning of the window. The Linear segment selection starts the new downloads at the beginning of the window. Therefore, every round begins with $F_A = 0$, and $E_A = W$. From this starting condition $S = a_B$ follows naturally. □

4. System dynamics and the steady-state

4.1. State-space description of the system

The dynamics of the P2P window of the average of the clients is best described with a difference equation of the state variable $\overline{E}$. The number of Empty segments is updated in every round in accordance to the segment selection scheme and the window advancement scheme.

Theorem 1 The change in the number of Empty segments in a round on average is

$$\Delta\overline{E} = \overline{A} - \overline{S} - \overline{M}. \quad (1)$$

Proof In each round the number Empty segments is changed by the following three factors:

- Some of them are converted to Full
- When the window advances, new Empty segments are shifted in
- Miss occurs, if the segment shifted out is Empty

When combining these three we get $\Delta E = A - S - M$ for a single client. Calculating the expected value of this yields equation (1), because the mean of a sum is the sum of the means. □

Equation (1) is a first order, linear, time-invariant difference equation for the variable $\overline{E}$. On the right hand side $\overline{S}$ is a function of $\overline{E}$, and $\overline{A} - \overline{M}$ is the drive, because they are independent of $\overline{E}$ (at least in the Streaminglike scheme).

4.2. Steady state

Definition 5 Steady-state: a constant response of the system for constant input.

The sufficient condition for the steady-state to exist is the asymptotic stability of the system.

Definition 6 Asymptotic stability: a system is asymptotically stable, if its output is $y(t) \to 0$, when its input is $s(t) = 0$ (undriven response).

Note that usually the steady-state of a stochastic system is defined such that the distribution of the system variable converges to a certain distribution at time $t \to \infty$. In this work we could only examine the ensemble average in most cases; thus, we had to relax the requirements here.

In the following sections most of the theorems are going to present results for the steady-state of equation (1), but first the existence of the steady-state has to be proven. The steady-state does not necessarily exist for a dynamic system. Theoretically it is quite possible that the output keeps oscillating in the absence of an input signal, if there is no energy loss in the system.

Theorem 2 The system defined in equation (1) is asymptotically stable for both Linear and Random segment selection.

Proof The undriven response is obtained for $\overline{A} = 0$, in which case $\overline{M} = 0$ follows naturally. For both segment selection schemes $E > \overline{S}(E, p)$ if $E > 0$. Obviously, $S = 0$ if $E = 0$. Thus, from any $E_0 > 0$ starting point $E_0 + \sum \Delta E = E_0 - \sum \overline{S}$ reaches $E = 0$ eventually. □

In the steady-state the following connections between the parameters are true.

Lemma 2 With all segment selection and window placement schemes $\overline{S} = \overline{A} - \overline{M}$ in the steady-state.

Proof In the steady state $\Delta E = 0$; thus, we get the statement of the lemma from equation (1) for the system dynamics. □

Lemma 3 In the steady-state $O_A = ST$ with all segment selection and window placement schemes.

Proof In the steady state in every round S new downloads are started, and they all last for T timeframes. □

Lemma 4 In the steady-state

$$\overline{M} = \begin{cases} 0 & \text{if } \overline{a}_B > 1 \\ 1 - \overline{a}_B & \text{if } \overline{a}_B < 1 \end{cases} \tag{2}$$

with all segment selection and window placement schemes.

Proof If there is at least one Full segment at the beginning of the window, there is no miss, when the window advances. If $\overline{a}_B < 1$, it means that there is a Full segment at the beginning of the window with probability $\overline{a}_B$. In such cases $A = 1$ with both window placement schemes (see their definitions in Section 3.4); thus, a segment is missed with probability $1 - \overline{a}_B$. □

5. The number of full segments at the beginning of the window

Using the lemmas from the previous section we can prove a surprising statement that is a refined, stronger version of the statement of lemma 2 for the Progressive scheme.

Theorem 3 With the Progressive window positioning scheme $\overline{S} = \overline{a}_B$ in the steady-state.

Proof For Linear segment selection this is trivial, and it has already been part of lemma 1.

With Random segment selection we know from lemma 2 that $\overline{S} = \overline{A} - \overline{M}$.

If $\overline{a}_B > 1$, then $A = \overline{a}_B$ from the definition of the Progressive scheme, and $\overline{M} = 0$ from lemma 4; thus, $\overline{S} = \overline{a}_B - 0$.

If $\overline{a}_B < 1$, then we have $A = 1$ from the definition of the Progressive scheme, and $\overline{M} = 1 - \overline{a}_B$ from lemma 4; thus, $\overline{S} = 1 - (1 - \overline{a}_B)$. Note that this also proves $\overline{S} = \overline{a}_B$ for Linear selection; thus, using lemma 1 was not necessary. □

This theorem describes a strange phenomenon. The Random segment selection scheme starts downloads all over the P2P window, yet the number of

newly started downloads equals the number of Full segments at the beginning of the window. Note that only the average value of these two quantities equal: they have entirely different probability distributions, as *Fig. 4* shows. Their domain can also be different, if $D < W$, because $a \in \{0 \ldots W\}$, and $S \in \{0 \ldots \min(D,W)\}$.

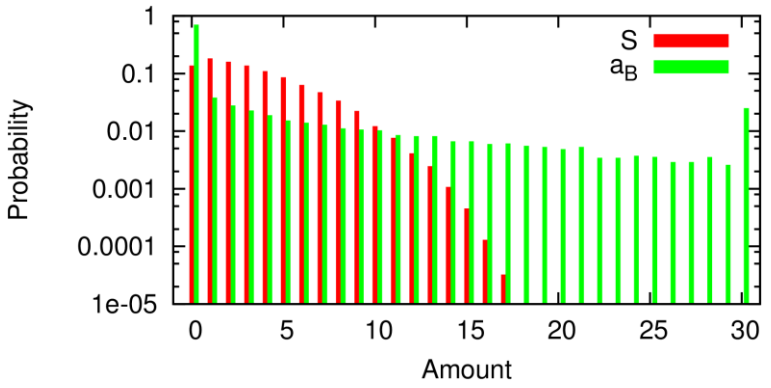


Figure 4: Comparison of the distributions of a_B, and S for the Progressive Random scheme (p=0.3, W=30, D=∞)

6. Number of new segment downloads initiated in a round

The next two theorems show how the number of new downloads started in a round can be calculated, if the number of Empty segments in the P2P window is known.

Theorem 4 With Random segment selection the number of new downloads started in a round is

$$\overline{S} = p\overline{E}_A \tag{3}$$

Proof If the number of Empty segments in the window is E_A in a round at point A, then the number of segments started follows Binomial distribution B(E_A, p), and pE_A downloads are started on average. However, E_A is also a random variable. Calculating average S for the average E_A yields

$$\overline{S} = \sum_{i=1}^{W} pi\mathbf{P}(E_A = i) = p\sum_{i=1}^{W} i\mathbf{P}(E_A = i) = p\overline{E}_A \tag{4}$$

which had to be proven. □

Theorem 5 With Linear segment selection the number of new downloads started in a round is

$$\overline{S} = p\frac{1-p^{\overline{E}_A}}{1-p} \tag{5}$$

Proof With Linear segment selection the round starts with F_A Full segments at the beginning of the window, and the remaining W - F_A segments are Empty. The probability that i segments are started in a round can be calculated from

$$P_i = \mathbf{P}(i\text{th is the first Empty}) = p^{i-1}(1-p) \tag{6}$$

Starting all of the Empty segments happens with probability p^{E_A}. The expected number of newly started downloads is

$$\begin{aligned}\overline{S} =& \mathbf{E}\{P_i\} + (E_A+1)p^{E_A} - 1 = \sum_{i=1}^{E_A} ip^{i-1}(1-p) + (E_A+1)p^{E_A} - 1 = \\ =& \frac{1-p^{E_A}}{1-p} - E_A p^{E_A-1} + (E_A+1)p^{E_A} - 1 = p\frac{1-p^{E_A}}{1-p},\end{aligned} \tag{7}$$

where the derivative of the formula for the sum of Geometric series was used. The generalization for the average $\overline{E}_A$ is done the same way as for the Random scheme. □

These results are applicable for the system at any time, not just the steady-state, but they depend on $\overline{E}_A$, which is unfortunately only known in the steady-state, as the next section shows.

7. Number of empty segments with the streaminglike scheme

The Streaminglike scheme always advances the window with A = 1 segments. The main concern in this case is the number of Empty segments in the window in the steady-state, because that will be the key to determine the necessary conditions for $\overline{M} = 0$.

7.1. Linear segment selection

If the window advancement scheme is less aggressive than the Progressive scheme, there may be some Full segments at the beginning of the window. The remaining E_A = W - F_A segments are all Empty, because the download process terminates at the first unsuccessful attempt.

Theorem 6 With Linear segment selection the expected value of the number of Empty segments in the steady-state is

$$\overline{E}_A = \begin{cases} \frac{\log(2p-1)-\log(p)}{\log(p)} & \text{if } p > 0.5 \\ W & \text{if } p \leq 0.5 \end{cases} \tag{8}$$

Proof We have seen in lemma 2 that $\overline{S} = \overline{A} - \overline{M}$ in the steady-state. With the Streaminglike scheme $A = 1$; thus,

$$\overline{S} = 1 - \overline{M} \tag{9}$$

If $\overline{M} = 0$, then $\overline{S} = 1$. Solving equation (5) for $\overline{S} = 1$ yields the first part of the formula in the statement. That formula yields $E = \infty$ for $p < 0.5$; thus, we also got the threshold p required for $\overline{M} = 0$.

If $\overline{M} > 0$, then we know from lemma 4 that $\overline{M} = 1 - \overline{a}_B$. We also know from the definition of the Linear selection that $F_B = a_B$. Using these we get that

$$\overline{S} = 1 - \overline{M} = \overline{a}_B = \overline{F}_B = W - \overline{E}_B = W - \overline{E}_A + \overline{S} \tag{10}$$

Thus, we got $\overline{E}_A = W$. □

Fig. 5 shows that this scheme results in a two-phase behavior: the segments in the P2P window are either all Empty, or all Full, depending on p. The results in the supercritical phase are indistinguishable from the Random segment selection, but in its subcritical phase the Linear scheme performs much worse.

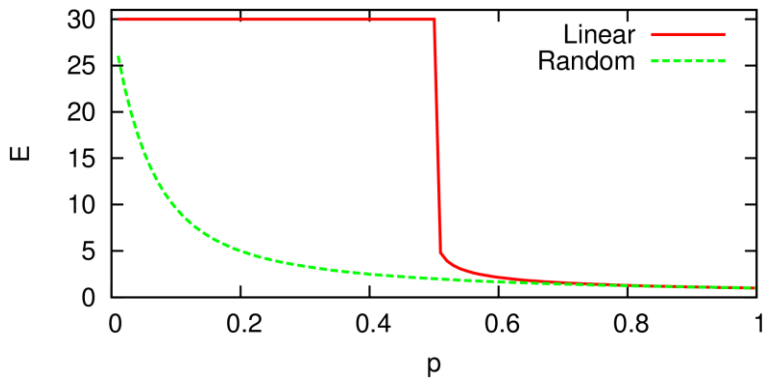


Figure 5: Number of Empty segments in the P2P window in the steady-state of the Streaminglike scheme; W=30

7.2. Random segment selection

The distribution of the number of Empty segments in the window is difficult to calculate, but getting their average number is quite easy.

Theorem 7 With Random segment selection the expected value of the number of Empty segments in the steady-state is

$$\overline{E}_A = \frac{1-(1-p)^W}{p} \tag{11}$$

Proof In the Streaminglike scheme $A = 1$; thus, we know that the ith segment of the P2P window has spent $W - i$ rounds in the window. Therefore, it is Empty, if it wasn't started in any of those rounds. The probability of a segment being empty is thus

$$\mathbf{P}(s_i = E_A) = (1-p)^{W-i} \tag{12}$$

independently of the other segments. The expected value of the number of Empty segments is the sum of the individual Empty probabilities

$$\overline{E}_A = \sum_{i=1}^{W} P(s_i = E) = \sum_{i=1}^{W} (1-p)^{W-i} = \frac{1-(1-p)^W}{p} \tag{13}$$

which had to be proved. □

In this case there is no phase change. As *Fig. 5* shows, the number of Empty segments gradually decreases as p increases. The minimum of both curves is at $\overline{E}_A = 1$, because after the advance there is always one Empty segment – the one at the end of the window.

8. Advance speed in progressive scheme

The P2P scheme is always self-sufficient, if there are no missed segments. In the Progressive scheme $\overline{a}_B > 1$ is a sufficient condition for this, because it means that the distance between the P2P window and the playback position keeps expanding. By the time it reaches the steady state, the probability that starting the download of the first segment fails so many times that a coerced advance becomes necessary is negligible.

In this section we calculate the expected advance speed of the P2P window, and the download initiation probability p_c required for $\overline{a}_B > 1$.

8.1 Linear segment selection

Theorem 8 With Linear segment selection the advance speed of the P2P window is

$$\overline{A} = \begin{cases} \overline{S} & \text{if } \overline{S} > 1 \\ 1 & \text{if } \overline{S} < 1 \end{cases} \tag{14}$$

Proof If we substitute $S = a_B$ from lemma 1 into the definition of the Progressive scheme, we get the equation of the statement. □

To calculate the download initiation probability p required for $S \geq 1$, and thus $M = 0$, equation (5) has to be evaluated for $S = 1$ and $E_A = W$.

$$1 = p\frac{1-p^W}{1-p} \quad \rightarrow \quad 0 = p^{W+1} - 2p + 1 \tag{15}$$

This is a polynomial of degree W+1, for which no general solution formula exists. We solved this equation numerically, and compared the results to the Random scheme in *Fig. 6*. As expected, $p = 0.5$ is the threshold value for reasonable window sizes ($W > 10$), which is significantly higher than the threshold value of the Random scheme.

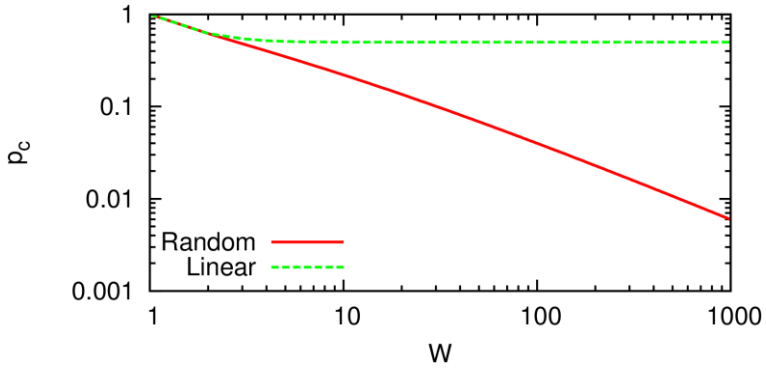


Figure 6: Comparison of the p_c threshold for $a > 1$, for Random and Linear segment selection schemes

8.2. Random segment selection

Determining the advance speed for Random segment selection is significantly harder than for Linear segment selection. The probability of a segment being Empty depends on the state of all the other segments, and the distribution of the advance speed. In turn, the distribution of the advance speed depends on the Empty probabilities of all the segments. We found that determining even the average advance speed requires multivariate numerical

optimization. Instead of finding an exact solution, we decided to construct an approximation of $\overline{a}_B$ that ignores this inter-dependency, yet gives usable results.

If we ignore the inter-dependency between *A* and *E*, and assume near constant $\overline{A}$ in at least the previous $W / \overline{A}$ rounds, the segment in the *k*th position has been in the window for $(W-k+1)/\overline{A}$ rounds, including the current one (in point B). In other words, it had exactly that many opportunities to become Full so far. The first one is special, because it must have been definitely left Empty in the last round, otherwise the window would have advanced at least one segment more. The number of opportunities each segment had so far is thus

$$g(k) = \begin{cases} \frac{W-k+1}{\overline{A}} & \text{if } k > 1 \\ 1 & \text{if } k = 1. \end{cases} \tag{16}$$

Our first idea was to take formula (11) for $\overline{E}_A$ in the Streaminglike case, adapt it to $A \neq 1$ using $g(k)$, and calculate $\overline{S} = p\overline{E}_A$, as shown in Theorem 4, but that was a failure. The results were completely different from the simulation results.

The approach that gave usable results was to construct the probability distribution of the number of Full segments at the beginning of the P2P window. Here we assume that $\overline{A} = \overline{a}_B > 1$*; t*he distribution of the number of Full segments at the beginning of the window is thus

$$\begin{aligned} \mathbf{P}(f(p) = 0) &= (1-p)^{g(1)} \\ \mathbf{P}(f(p) = W) &= \prod_{j=1}^{W} (1-(1-p)^{g(j)}) \\ \mathbf{P}(f(p) = i-1) &= (1-p)^{g(i)} \prod_{j=1}^{i-1} (1-(1-p)^{g(j)}), \end{aligned} \tag{17}$$

where segments [1,*i*-1] are Full and segment *i* is the first Empty. This is a probability measure, as its sum is 1. The expected number is thus

$$\begin{aligned} \overline{A} &= \overline{a}_B = \mathbf{E}\{f(p)\} = \\ &= \sum_{i=2}^{W} (i-1)(1-p)^{g(i)} \prod_{j=1}^{i-1} (1-(1-p)^{g(j)}) + W \prod_{j=1}^{W} (1-(1-p)^{g(j)}). \end{aligned} \tag{18}$$

This cannot be solved analytically, because $g(k)$ in the exponents also contain *A*. We solved equation (18) numerically by fine-tuning the *A* substituted

into $g(k)$ until equation (18) returned the same A, for the given p, and W. When comparing the results with simulation results, we found that the probability distribution in equation (17) is completely different from the real distribution, as demonstrated in *Fig. 7*, but $\bar{a}_B$ from equation (18) always underestimates the real one, and it's usually about 95-98% of it. *Fig. 8* shows an example of this result for $W = 10$.

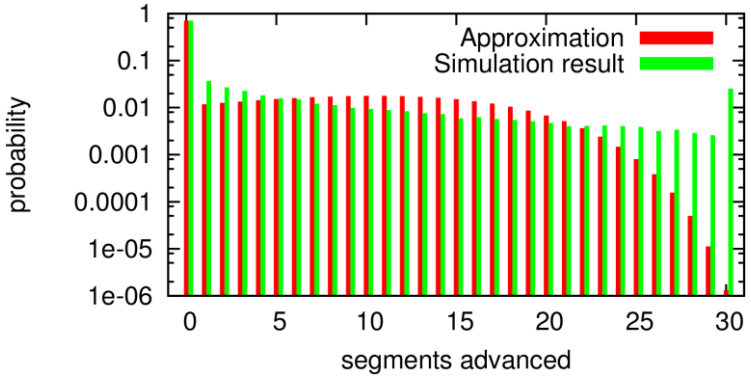


Figure 7: Comparison of the real distribution of a, and the distribution predicted by the approximation (p=0.3, W=30, D=∞, a_{approx}=3.185, $a_{measured}$=3.265)

The above approximation method assumes that $\bar{A} = \bar{a}_B > 1$. As *Fig. 8* shows, for $a_B < 1$ this method indeed looses its precision considerably. We tried to correct this by using the same formula, but substituting $A = 1$ into $g(k)$; however, it made the results worse. Accurate results for $a_B < 1$ would have been useful to calculate the miss probability, but at least this approximation technique is good enough to provide a threshold p for miss-free operation.

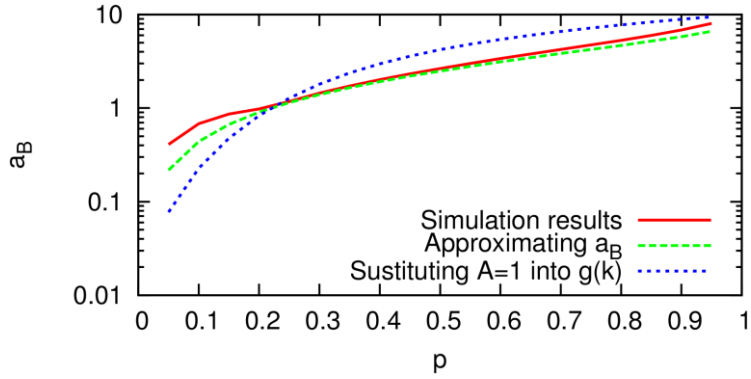


Figure 8: Comparison of the simulation result, the approximations of a_B with the assumptions A=a_B, and A=1 (W=10, D=∞)

Solving equation (18) numerically for $\overline{a}_B = 1$ yields a threshold value of p for the P2P scheme to be self-sufficient. As *Fig. 9* shows, this result overestimates the real threshold value obtained from simulations, because the approximation underestimates $\overline{a}_B$. This threshold is shown in *Fig. 6*, compared to the threshold value for the linear scheme. As expected, the Progressive scheme is much more efficient than the Streaminglike scheme.

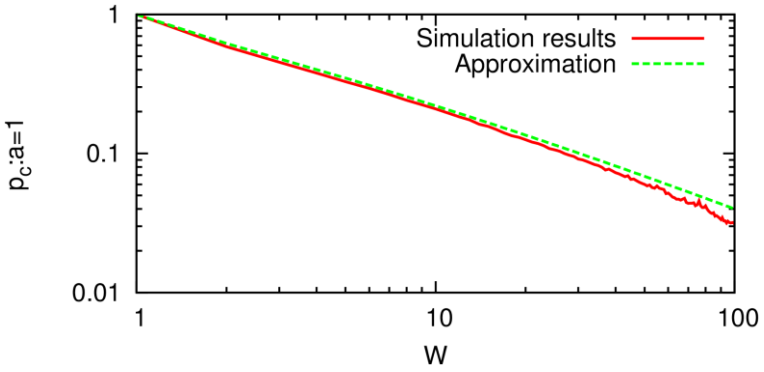


Figure 9: Comparison of the approximation and the simulation results for the p_c threshold for a>1 for the Progressive Random scheme

9. The probability of missing a segment

At the beginning of the download process the transient processes behave quite erratically, which prevents determining analytically the number of segments missed. In the steady-state, however, the miss probability is easy to calculate. The theorems in this section are specializations of lemma 4.

9.1. Linear segment selection

With this segment selection scheme the miss probability is the same for both window placement schemes under normal circumstances.

Theorem 9 With Linear segment selection the miss probability in the steady-state is

$$\overline{M} = \begin{cases} 1 - \overline{S} & \text{if } p < 0.5 \\ 0 & \text{if } p \geq 0.5. \end{cases} \tag{19}$$

With the Progressive scheme the threshold is higher for tiny P2P windows (see *Fig. 6*), but that can be safely ignored.

Proof For the Streaminglike case we have already seen this in the proof of theorem 6. If $p < 0.5$, then $\overline{M} = 1 - \overline{S}$. If $p \geq 0.5$, then $\overline{S} = 1$, and $\overline{M} = 0$.

In the Progressive case theorem 8 showed that the window is well ahead of the playback position, if $\overline{S} \geq 1$, for which the necessary condition is $p \geq 0.5$ for reasonable P2P window sizes (see *Fig. 6*); in this case $\overline{M} = 0$. If $\overline{S} < 1$, the system behaves exactly like the Streaminglike scheme. □

For small *p* one would think that the window is completely Empty, and the first segment is converted to Full with probability *p*; thus, $\overline{M} = 1 - p$. This is only true for a single client; though, because the average behavior of several clients also includes the uncertainty of a_A.

The exact value of $\overline{M}$ in the steady-state can be computed from equation (5), by substituting equation (8).

The advantage of the Linear segment selection scheme is thus the ability to run the system without missed segments regardless of the window positioning scheme, even though the required *p* is rather high.

9.2. Random Segment Selection

With Random selection the window placement scheme can heavily influence the miss probability.

Theorem 10 In the steady-state of the Streaminglike Random case the number of segments missed is

$$\overline{M} = (1 - p)^W \tag{20}$$

Proof The probability of missing a segment equals the probability of the first segment in the window being Empty. That segment has spent *W* rounds in the window; thus, it is still Empty only if all of the starting attempts failed, which happens with probability $(1 - p)^W$.

The same result can be obtained by using theorem 4, lemma 2, theorem 7, and knowing that *A*=1 with the Streaminglike scheme. The miss probability is thus

$$\overline{M} = A - \overline{S} = 1 - \overline{S} = 1 - p\overline{E} = 1 - p\frac{1 - (1 - p)^W}{p} = (1 - p)^W \tag{21}$$

as expected. □

Theorem 11 In the steady-state of the Progressive Random case the number of segments missed is

$$\overline{M} = \begin{cases} 1 - \overline{a}_B & \text{if } \overline{a}_B < 1 \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

Proof This is basically lemma 4 with theorem 3 substituted. □

In the Progressive Random case the numerical result for $\overline{a}_B$ can be used to estimate p_c, and it can also give a crude estimate of $\overline{M}$.

Although the Random segment selection scheme cannot guarantee that the system is self-sufficient in all cases, the probability of a missed segment can be kept minimal by choosing a sufficiently large P2P window. *Fig. 10* shows a comparison of the Miss probability for all four combinations. The Random selection scheme is much more efficient than the Linear selection for low *p* values, but when the download initiation probability is high enough, the performance of the two segment selection schemes is identical.

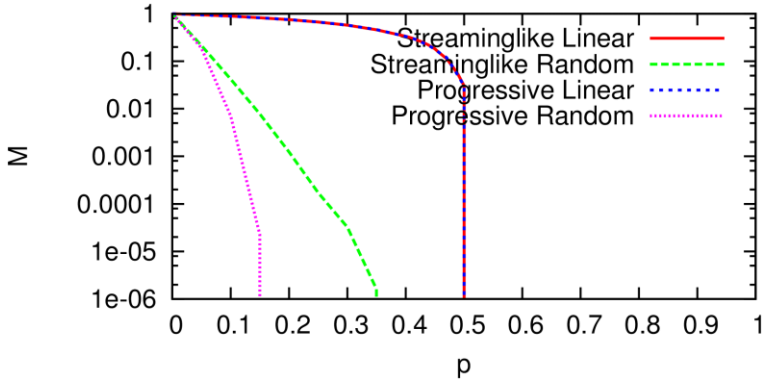


Figure 10: Miss probability with the four combinations of the schemes; *W*=30

10. Limited downlink capacity

In the previous sections $D = \infty$ was always assumed, but in real P2P clients that is not always true. In the literature it is generally assumed that the performance of a P2P system is limited by either the uplink speed of the clients, or their downlink speed [13, 8]. In this section we show that these two kinds of limitation can occur at the same time.

We initially discovered the phenomenon discussed in this section, when we were conducting our simulation study of a combined caching-P2P VoD system [7]. In that simulation study we used our own network simulator, *cdnsim*, which we also used to obtain some of the results presented in this section.

The remainder of the results of this section was obtained with a much simpler simulator, which we called *sim.pl*. We originally created it to validate the theoretical results presented in the previous sections, and later extended it to support limited downlink capacity. It basically starts with an array of "E", runs the algorithm of *Fig. 2*, with a predefined constant download initiation probability, and stops when all elements became "F".

The analysis presented here focuses mainly on the Progressive Random scheme, because, being the most efficient combination, it is the one that is affected the most by the limited downlink. Most of the observations are also applicable to the other scheme combinations, but the numeric values are different.

The downlink capacity D of the clients cannot be arbitrarily small, and there is a finite size, above which it is not limiting. The following lemmas establish the valid interval for the downlink capacity.

Lemma 5 The downlink capacity of the clients must be $D \geq T$.

Proof A timeframe is the time it takes to play a video segment, and it takes *T* timeframes to download a segment. Therefore, the client needs to be able to download at least *T* segments in parallel to keep up with the playback speed.

Lemma 6 If $p = 1$, the downlink limits the efficiency of the P2P downloading, when $D < WT$ in the Progressive scheme.

Proof If $p = 1$, all of the Empty segments in the P2P window are filled in each round; thus, $A = W$, and the next round is started with $E_A = W$ independent of the segment selection scheme. Therefore, $S = W$, and $O = WT$.

Lemma 7 If $p = 1$, the downlink limits the efficiency of the P2P downloading, when $D < W + T$ in the Streaminglike scheme.

Proof If $p = 1$, the window is filled in the first timeframe; thus, $S = W$. In the subsequent rounds there is only one Empty segment in the window, because *A*=1, and the segment is converted into Full as soon as it is shifted into the window; therefore, *S*=1. The number of ongoing downloads is maximal in the *T*th round: this is the last one, where the initial downloads are still ongoing. In this round $O = W + T$.

Note that the Streaminglike scheme only needs at most *D*=*T*, if the startup phase is not considered, because $S \leq 1$ in the steady-state with both segment selection schemes. If $T < D < W + T$, then it takes more time to reach the steady-state, but in the steady-state the system is not limited by the downlink capacity.

As explained in section 3.1, the download initiation probability *p* in the previous sections was a supply/demand ratio that represents the global state of the P2P swarm. In this section

$$q = p * \mathbf{P}(O < D) \tag{23}$$

will be the *real* download initiation probability, which may be smaller than p due to the limited downlink. It is obvious that q changes even within a round, because O is increased with each new successful download attempt.

In summary, to determine whether D limits S, we need to know the distribution of O, and to compute that we need the distribution of S. The distribution of S obviously depends on the distribution of E_A, and O; thus, constructing an exact formula for q seems hopeless. For this reason, there will only be simulation results in the remainder of this section.

Fig. 11 shows the download initiation probability in *cdnsim*. The oscillation in the startup process makes it difficult to see, but q starts at around 0.13, and it decreases to around 0.08 due to the limited downlink capacity. At the end of the download process the P2P window cannot go further; thus, with the decreasing number of remaining Empty segments the downlink capacity of the client is no longer a limiting factor. This causes the $q=p$ spike at the end of the curve.

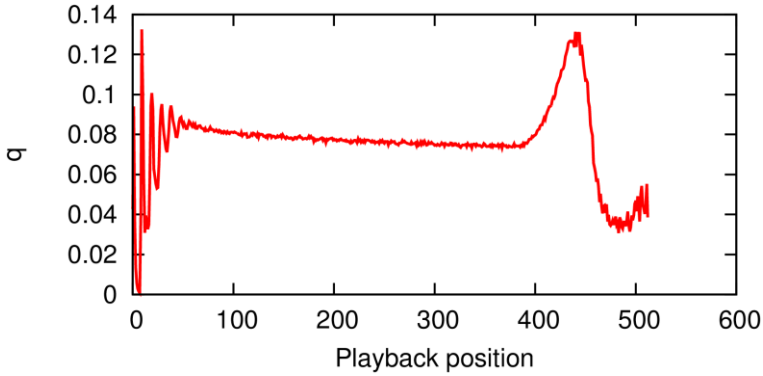


Figure 11: Download initiation probability during the playback process in cdnsim with limited downlink; *T*=8, *W*=30, *D*=16

A similar pattern can be observed in *Fig. 12*, but this time p is known precisely, because for *sim.pl* it is an input parameter. This figure also shows $p\mathrm{P}(O_B < D)$ for comparison, and in this parameter configuration the two almost perfectly match. With other parameter settings the difference between them can be larger, because the downlink can get filled anywhere between point A and point B during the round.

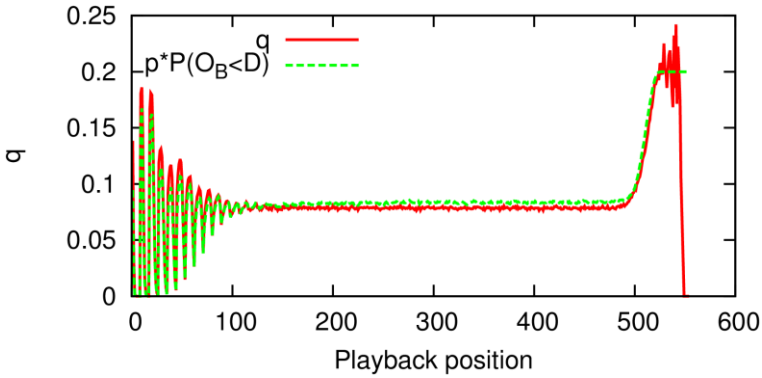


Figure 12: Download initiation probability during the playback process in sim.pl with limited downlink; p=0.2, T=9, W=30, D=10

To further verify that this phenomenon is indeed caused by the limited downlink, and not by the finite P2P window size, *Fig. 13* offers some more insight. The q/p ratio is 1 for small p values, but after that it decreases rapidly. With higher downlink capacity it only starts to decrease later, but the steepness of the slope remains almost the same. The q/p ratio also depends on W, of course: with bigger P2P window the downlink limitation becomes more severe.

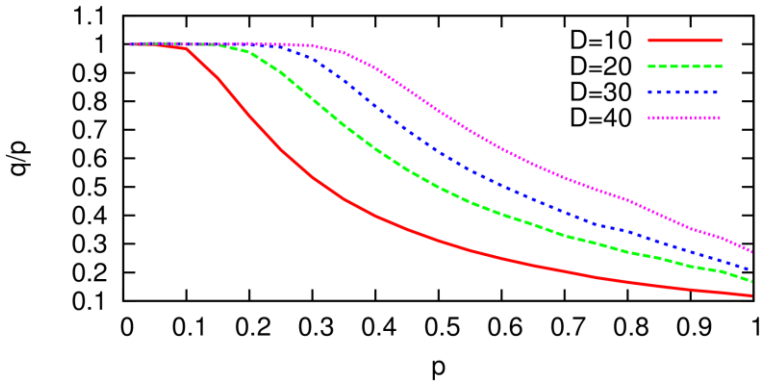


Figure 13: The q/p ratio for various downlink capacities in sim.pl; T=9, W=30

There is yet another curious phenomenon, which arises with limited downlink capacity. Fig. 14 shows the distance between the playback position and the start of the P2P window. With the Progressive scheme it is naturally not constant, but the peak at the end is quite remarkable. When q increases at the end of the video, the speed of the P2P window follows, and thus creates a peak on this diagram as well. According to our simulation results, the window position doesn't always have a peak, when q does.

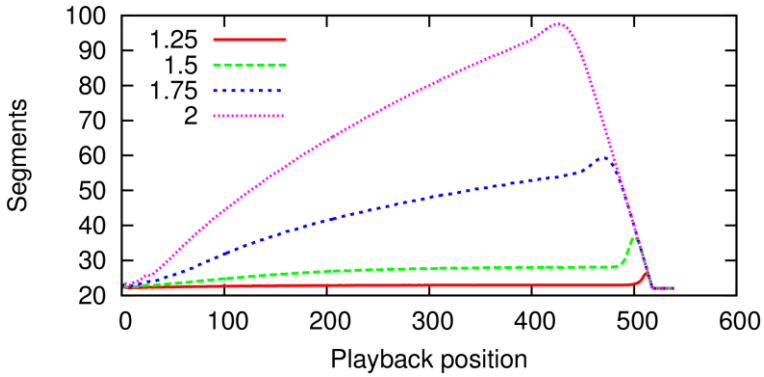


Figure 14: Distance between the playback position and the beginning of the P2P Window with the Progressive Random scheme in cdnsim, for some downlink speeds in range [1.25…2] relative to the video bitrate

This pattern only arises for the average behavior of several clients; though. When observing a single client, the relative position of the P2P window is completely different, as *Fig. 15* demonstrates.

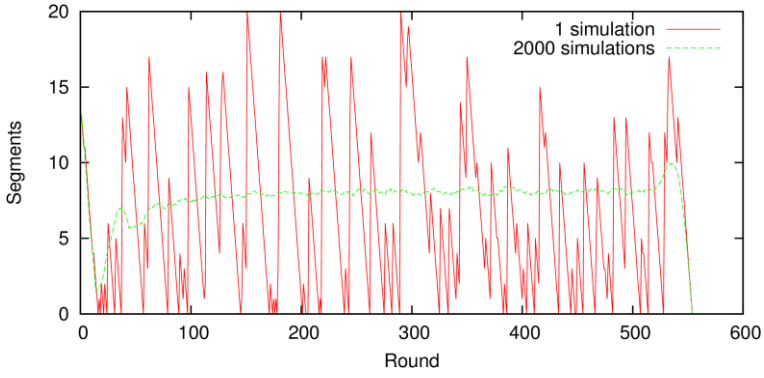


Figure 15: Distance between the playback position and the beginning of the P2P Window; comparing the behavior of a single client and the ensemble average

This description of the limited downlink has the advantage that the effect of $D < \infty$ is contained entirely in the download initiation probability; thus, the results of the previous sections still hold, if q is used in place of p in the formulas.

11. Conclusions

In this paper we examined analytically the performance of a generic P2P VoD client that uses closed P2P windowing. In this model the clients can only initiate the download of the video segments that are within the P2P window; the segments of the window can be selected in linear or random order. We examined two methods of positioning the P2P window: the Streaminglike scheme maintains a fixed distance between the window and the playback position, while the Progressive scheme pushes the window forward as fast as possible.

We described the evolution of the state of the P2P window with a linear difference equation, and showed that it can reach a steady-state. We developed formulas for the state descriptor quantities, depending on the segment selection scheme, the window positioning scheme, and the segment download initiation success probability. Perhaps the most important results of this analysis are the criteria for the P2P system to be self-sufficient, but we also discovered an interesting connection between the advance speed of the P2P window and the number of downloads initiated in a round, when using the Progressive window placement scheme. Finally, we examined the effect of limiting the downlink speed of the clients, and found that, unlike the popular assumption, the uplink and the downlink capacities limit the efficiency of the content sharing at the same time.

References

[1] Axelrod, R., "The evolution of cooperation". *Basic Books* (1984).

[2] Carlsson, N., Eager, D. L., "Peer-assisted on-demand streaming of stored media using BitTorrent-like protocols". In: *Proc. of NETWORKING'07*, pp. 570–581. Springer-Verlag, Atlanta, GA, USA (2007).

[3] Ciullo, D., Martina, V., Garetto, M., Leonardi, E., Torrisi, G. L., "Stochastic analysis of self-sustainability in peer-assisted vod systems". In: *Proc. of INFOCOM'12*. IEEE, Orlando, FL, USA (2012).

[4] Cohen, B., "Incentives build robustness in BitTorrent". In: *Proc. of 1st Workshop on Economics of Peer-to-Peer Systems* (2003).

[5] Erman, D., Saavedra, D., Sánchez González, J., Popescu, A., "Validating bittorrent models". *Telecommunication Systems 39*, 103–116 (2008).

[6] Locher, T., Moor, P., Schmid, S., Wattenhofer, R., "Free riding in bittorrent is cheap". In: *Proc. of HotNets-V*. Irvine, CA, USA (2006).

[7] Máté, M., Vida, R., Mihály, A., Császár, A., "Offloading video servers: P2p and/or caches?" In: *Proc. of NETWORKS2012*, pp. 400–405. Rome, Italy (2012).

[8] Parvez, N., Williamson, C., Mahanti, A., Carlsson, N., "Analysis of BitTorrent-like protocols for on-demand stored media streaming". *SIGMETRICS Perform. Eval. Rev.*, ACM 36(1), 301–312 (2008).

[9] Pouwelse, J. A., Garbacki, P., Epema, D. H. J., Sips, H. J., "The bittorrent p2p file-sharing system: Measurements and analysis". In: *Proc. of IPTPS'05*. Ithaca, NY, USA (2005).

[10] Qiu, D., Srikant, R., "Modeling and performance analysis of bittorrent-like peer-to-peer networks". In: *Proc. of SIGCOMM '04*, pp. 367–378. Portland, Oregon, USA (2004).
[11] Shah, P., Pâris, J. F., "Peer-to-peer multimedia streaming using bittorrent". In: *Proc. of IPCCC'07*. New Orleans, LA, USA (2007).
[12] Vlavianos, A., Iliofotou, M., Faloutsos, M., "BiToS: Enhancing BitTorrent for supporting streaming applications". In: *Proc. of INFOCOM'06*, pp. 1–6. IEEE, Barcelona, Spain (2006).
[13] Ying, L., Srikant, R., Shakkottai, S., "The asymptotic behavior of minimum buffer size requirements in large p2p streaming networks". In: *Proc. of ITA2010*, pp. 1–6. San Diego CA, USA (2010).
[14] Zghaibeh, M., Harmantzis, F., "A lottery-based pricing scheme for peer-to-peer networks". *Telecommunication Systems 37*, 217–230 (2008).
[15] Zhao, B., Lui, J., Chiu, D. M., "Exploring the optimal chunk selection policy for data-driven p2p streaming systems". In: *Proc. of IEEE P2P'09*, pp. 271–280. Seattle, WA, USA (2009).

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 7 (2015) 62-79

Seasonal Affective Disorder Speech Detection on the Base of Acoustic-Phonetic Speech Parameters

Gábor KISS, Klára VICSI

Department of Telecommunication and Media Informatics,
Budapest University of Technology and Economics, Hungary
e-mail: {kiss.gabor; vicsi}@tmit.bme.hu

Manuscript received February 11, 2016; revised April 11, 2016.

Abstract: The development of an online monitoring system is shown in order to track the physiological and cognitive condition of crew members of the Concordia Research Station in Antarctica, with specific regard to depression. Follow-up studies were carried out on recorded speech material in such a way that segmental and supra-segmental speech parameters were measured for individual researchers weekly, and the changes in these parameters were detected over time. Typical acoustic-phonetic parameters were selected on the base of statistical analyses. On the base of the selected parameters, a function was developed which indicates the likelihood of depression at the examined person.

Keywords: Acoustic-phonetic speech analysis, seasonal affective depression, cognitive status monitoring, statistical analysis

1. Introduction

Speech reflects the physiological and cognitive condition of humans: therefore, there are changes in the acoustic phonetic parameters of speech when the condition of humans changes. For example, in case of vocal disorders, the acoustical parameters of disordered speech are significantly different from normal speech [1], [2], [3].

Psychology says experience of failure can cause depression, which has emotional, cognitive, physical and motivational symptoms; thus these symptoms may also be observed in speech. Depression is a major public health challenge, its prevalence is high and it has a major impact on sufferers [4]. Effective treatment is available, but depression care is facing barriers at several levels, such as under-recognition, stigmatization, inadequate treatment and mistreatment.

Because speech reflects the physiological and cognitive condition of humans, doctors can diagnose depression not only from patients' speech content but also from their speech quality. They often characterize depressed speech as faded, slow, monotonous, lifeless and metallic. We can link these properties with acoustic characteristics such as fundamental frequency, amplitude modulation, formant structure, energy distribution, etc.

Numerous studies have identified acoustic features that can be linked to depression. In some of the studies the differences between the speech of healthy and depressed people is measured; others perform follow-up monitoring to gather features with high classification performance. Prosodic parameters like rhythm, intonation, accent and timing are sensitive to changes in mood states and emotions [5], [6], [7], [8]. A lot of the research that has been done over the last few years deals with the relationship between depression and different acoustic phonetic parameters [9], [10], [11], [12]. One early study identified fundamental frequency as one of the most important acoustic features of depressed speech [13]. Nowadays many parameters are investigated at different levels of speech production: fundamental frequency, variation of fundamental frequencies, formants, power spectral density [14], cepstrum [15] or MFC coefficients [16], speech rate [17], glottal features [18], amplitude modulation and other different prosodic parameters [19].

The data suggest that depressed patients take more time to express themselves. They speak with greater hesitation, but they do not vocalize more. In this way they are producing more cumulative and variable pauses. Consequently, voice acoustic measures are examined that reflect depression severity such as the percentage of pause time, vocalization or pause ratio, and speaking rates. Pitch variability and first and second formants correlate significantly with overall depression severity.

Our goal is to develop a monitoring system with specific regard to Seasonal Affective Disorder (SAD) type of a depression. The monitoring system's aim is to track the physiological and cognitive condition of crew members of the Concordia Research Station in Antarctica. For the segmentation we used an automatic language-independent program, developed earlier, to segment the records in phoneme level for measurement [20].

For this research we have developed four types of databases: Seasonal Affective Disorder Speech Database and Healthy Reference Speech database are used for the examination of the sensitivity of acoustic-phonetic parameters of speech regarding depression and Concordia Speech Database 2013 and Concordia Speech Database 2014 are used for the physiological and cognitive status monitoring of the crew members in the Concordia research station; and the Healthy Follow-Up Speech Database is used for the physiological and cognitive status monitoring of healthy Hungarian speakers.

The paper is structured as follows. The descriptions of the databases used are presented in Section 2, detailed descriptions of evaluation methods in Sections 3 and 4, followed by results and conclusions in Sections 5 and 6.

2. Databases

In this part we give short descriptions of the databases that we developed and used in our research.

A.Seasonal Affective Disorder Speech Database

The database contains 55 sufferers: 35 female and 20 male. A psychiatrist from the Neurology Department of Semmelweis University, Hungary assisted in the selection of the patients. The database consists of two parts: the first part is a collection of spontaneous speech obtained from the discussion between the patient and the doctor; in the second part patients read a standard phonetically balanced short folk tale (about 6 sentences altogether) called "The North Wind and the Sun", frequently used in phoniatry practice for all European languages. The recordings were recorded with clip-on microphones (Audio-Technica ATR3350), an external USB sound card, at 44,100 Hz at a 16 kHz sampling rate, quantized at 16 bits.

In order to measure depression severity, Beck Depression Inventory (BDI) was used [21]. The BDI indices are within the range of 14 to 43. The mean age of subjects was 31,5 years, with a standard deviation of 12,3 years and a range of 18 to 63 years. For each patient we noted additional information about smoking habits, illnesses and prescribed medication.

B. Healthy Reference Speech Database

For the Healthy Reference Speech Database72 healthy speakers (28 male and 44 female) were asked to read the same tale, "The North Wind and the Sun". The recording conditions were the same as in the Seasonal Affective Disorder Speech Database, using clip-on microphones (Audio-Technical ATR3350) at a sampling rate of 44,100 Hz, quantized at 16 bits; the reference database recordings were also annotated and segmented on phoneme level, using the SAMPA phonetic alphabet. The BDI indices are within the range of 0 to 13. The mean age of subjects was 28,7 years, with a standard deviation of 10,4 years and a range of 18 to 52 years.

C. Concordia Speech Database 2013 and 2014

We are participating in an international ESA project, AO-11-Concordia, entitled Psychological Status Monitoring by Computerized Analysis of Language Phenomena (COALA). The records for this database were collected from crew members using their mother tongue. The records were made in a weekly period during their stay at the Concordia Station. Baseline recordings were made from the same crew members in normal circumstances in Europe before their departure for Antarctica. This way we had the opportunity to monitor the impact of hypoxia on speech and occasional occurrence of SAD symptoms.

The database consists of two parts: the first part is a collection of spontaneous speech obtained from the recorded diaries; in the second part the participants read the same phonetically balanced short folk tale (as described above) in their mother tongue.

The experiment was planned to be carried out in two seasons: 2013 and 2014. The crew members were native French, Italian or Greek a total of 20 people. In both seasons the doctor (always a member of the crew) was a native Greek speaking in English. The recordings were made with clip-on microphones (Audio-Technica ATR3350) at a sampling rate of 44,100 Hz, using 16 bits.

D. Healthy Follow-Up Speech Database

In this database we gathered the recordings of healthy people in everyday conditions. In our everyday lives there are many factors that can affect our speech (mood, fatigue, stress); thus the effect of these factors on speech is an important consideration. The creation of the database was necessary in order to have a good reference point for our studies to be able to compare the results, in this case when monitoring depression.

The database contains recordings of 10 participants: 5 female and 5 male. The male age distribution is between 21 and 29, and the female age distribution is between 26 and 57. The participants read the same phonetically balanced short folk tale as in the previous two databases, in Hungarian. The recordings were collected over three months at weekly intervals, gathering a total of 156 recordings from 9 people. For each person we noted additional information about smoking habits, illnesses and prescribed medication, and before the first recording we checked the participants' BDI score. As expected, all participants had low BDI scores: their distribution was between 0 and 8.

The recordings were made with the same equipment as in the previous two databases, using clip-on microphones (Audio-Technica ATR3350), with

external USB sound card, at 44,100 Hz at a 16 kHz sampling rate, quantized at 16 bits.

E. Segmentation and Labelling

All the databases' recordings need to be segmented into phoneme level in order to measure the acoustic-phonetic parameters. For this reason we used an automatic language-independent segmentation program, developed earlier in our laboratory. For each recording, manual correction was done.

3. Analysis of the acoustic-phonetic parameters

Our final goal is to monitor the speech of the crew members at the Concordia Station, and indicate if any of the crew members show any sign of depression which could lead to a serious level of depression. Speech, as an acoustic product, is very diverse. We can distinguish between inter-individual differences, that is, variety in speech among different people, and intra-individual differences, i.e., variety in the speech of one person. This variety can arise for many reasons, primarily the physical condition of the speaker (for example flu', alcohol condition, emotional state, stress, sleepiness, etc.), but it has some natural variety too, as speech is a non-deterministic process. This is the reason that our work includes the examination of the relationship between the acoustic-phonetic parameters and other biological and psychological data of the crew members measured at the station, such as long term medical survey data (LTMS) and oxygen saturation data.

In spite of the considerable variety in speech parameters, we would like to identify specific changes which indicate depression. This task is not obvious. We can eliminate the variety between speakers by monitoring only the changes of the acoustic parameters for each person over time.

In order to distinguish the changes in acoustic parameters caused by depression from the natural intra-individual variety we investigated the changes in acoustic parameters in speech of nine healthy people in normal circumstances. For this investigation the Healthy Follow-Up Speech Database was used.

A. Selection of acoustic-phonetic parameters

Having previously carried out a study to decide what kind of acoustic parameters can indicate depression [22], the variation in these acoustic parameters was examined over time: variance of intensity (VI), fundamental frequency (F0), variance of fundamental frequency (VF0), first formant (F1), variance of first formant (VF1), second formant (F2), variance of second

formant (VF2), jitter (J), shimmer (S), articulation rate (AR), speech rate (SR), and total length of pauses (TLoP). We added one further acoustic parameter which showed significant difference between healthy speech and depressed speech: the rate of transient (Rot) [23].

B. Pre-processing and segmentation

The recorded speech was automatically segmented into phoneme units with an automatic segmentation program, developed in our laboratory [23].

The acoustic-phonetic parameters were examined in two groups according to their segmental and supra-segmental (prosodic) features.

The segmental features were measured at the middle of the same vowel, in our case the vowel was "E". The following segmental features were measured as previously mentioned: F0 of 'E' vowels, F1 and F2 of 'E' vowels (F1, F2), VF1 and VF2 of 'E' vowels, jitter of the 'E' vowels, and shimmer of the 'E' vowels. For the measurement of formants, fundamental frequency and the spectral values, a Hamming window was used with 25 ms frame size; these features were always evaluated from the middle of each vowel 'E'.

The supra-segmental (prosodic) features were measured by the total length of each recording. The following features were measured: VI as volume dynamics of speech (range of intensity), VF0 as fundamental frequency dynamics of speech (range of fundamental frequency), ratio of total length of pauses and the total length of the recording, articulation and speech rate and rate of transient (Rot).

C. Variation of the selected acoustic-phonetic parameters in case of healthy speech

The selected acoustic parameters were evaluated for each record for each person. The difference from the mean value for each parameter was calculated. An example is presented in *Fig. 1.* Interpolation was applied between the measuring points.

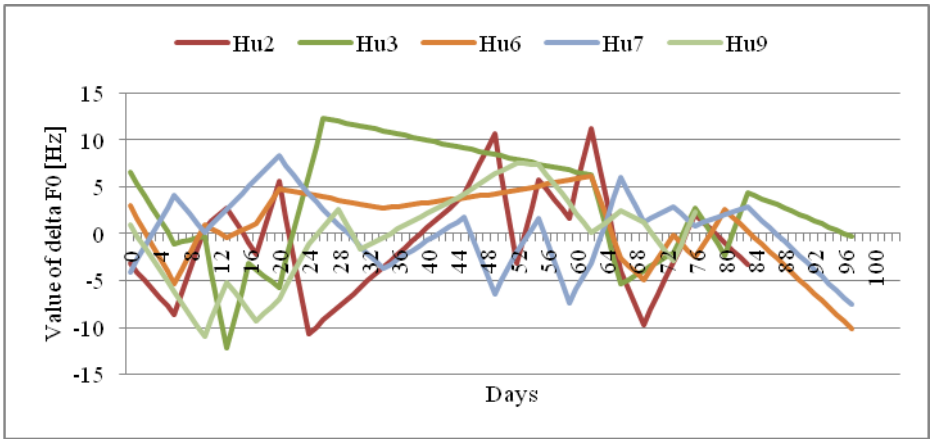


Figure 1: An example for the delta F0 changing in time for nine healthy males. Day 0 is the beginning of the examination

Maximum difference, from the mean value of the selected parameters, was calculated for each person for each parameter, and presented in *Table 1*.

Table 1: Absolute difference of the intra-individual variety in case of healthy speech

	Average		Maximum		Minimum	
	of the Absolute Difference of the Personal Variety					
	Female	Male	Female	Male	Female	Male
VI [dB]	0.58	0.55	1.39	1.27	0.18	0.15
F0 [Hz]	9	5	12	10	6	3
VF0 [Hz]	8	7	14	14	2.5	2
F1 [Hz]	23	24	34	36	11	10
VF1 [Hz]	11	10	21	19	5	5
F2 [Hz]	38	36	57	53	22	23
VF2 [Hz]	28	24	53	49	14	14
J [%]	0.6	0.5	1.1	1.0	0.2	0.2
S [%]	1.5	1.3	2.7	2.6	0.4	0.4
AR [phon. / s]	0.8	0.7	1.5	1.5	0.4	0.3
SR [unit / s]	0.9	0.9	1.4	1.5	0.4	0.4
TLoP [s]	2.6	2.8	3.5	4.1	1.5	1.6
RoT [%]	3.4	3.8	4.7	5.1	2.2	2.3

D. Statistical difference of the acoustical phonetic parameters between healthy speech and depressed speech

For our study we examined depressed speech, and selected a group of speech parameters which differ significantly in depressed speech when compared to normal speech [22]. We examined the weight of these parameters in the discrimination of healthy and depressed speech. Principal component analysis was carried out using Matlab on both the Seasonal Affective Disorder Speech Database and the Healthy Reference Speech Database.

Principal component analysis is an established tool for multivariate data analysis. Its goal is to predict which parameters address the greatest weight in the description of the information content of the data set.

During the individual examination of the components we did not find characteristic patterns. This indicates that we did not use features that are irrelevant in terms of depression; indeed, all the parameters are useful indicators of depression.

We investigated differences in the mean values of the selected parameters between the healthy and the depressed speech, especially in view of the intra-individual variation in the normal case. The results are presented in Table 2.

Table 2: The differences in the mean values of the changes of the acoustic phonetic parameters between healthy and depressed speech, compared to the normal intra-individual variation in case of healthy speech

	Mean difference between healthy and depressed speech		The average variation of the parameter	
	Female	Male	Female	Male
VI [dB]	-0.55	-0.1	+/- 0.58	+/-0.55
F0 [Hz]	-21	-6.6	+/- 9	+/-5
VF0 [Hz]	-4	-2.5	+/- 8	+/-7
F1 [Hz]	-18	-28.2	+/- 23	+/-24
VF1 [Hz]	+2.8	+6.3	+/- 11	+/-10
F2 [Hz]	+35	+42.1	+/- 38	+/-36
VF2 [Hz]	+11	+21.2	+/- 28	+/-24
J [%]	+0.3	+0.3	+/- 0.6	+/-0.5
S [%]	+1.5	+1.5	+/- 1.5	+/-1.3
AR [phon. / s]	-0.3	-1.7	+/- 0.8	+/-0.7
SR [unit / s]	-0.13	-1.6	+/- 0.9	+/-0.9
TLoP [s]	+1.1	+3.7	+/- 2.6	+/-2.8
RoT [%]	-1.5	-5.25	+/- 3.4	+/-3.8

As can be seen from the *Table 2*, the differences in the mean values of the acoustic phonetic parameters between the healthy and the depressed speech are almost as big as the variation of these parameters in normal, healthy speech. This can cause many problems in processing and separation. However, it does not mean that we are not able to distinguish between healthy speech and depressed speech, rather that the task is hard, because only the *likelihood* of depression can be predicted on the basis of the parameters.

4. Depression status monitoring of the crew members at Concordia

We built an online monitoring system that follows the acoustic-phonetic parameters of a subject's speech over time and compares the collected data with the mean value. The system is designed to create an alert if one of the subjects suffers depression. An automatic (semi-automatic) online alerting system was built which records speech at a given time rate and creates alerts to the appearance of depression. The flow chart of the online alerting system is shown in *Fig. 2.*

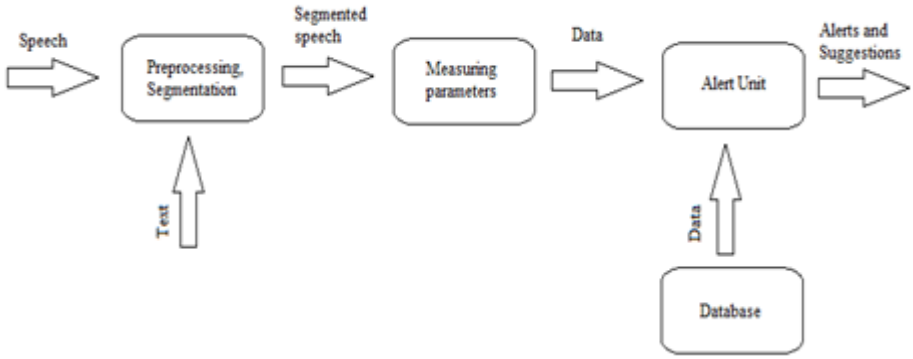


Figure 2: The online alerting system flowchart

The system has three main parts: Preprocessing and Segmentation unit, the unit that measures the acoustic parameters (Measuring Parameters), and the Alert unit. Preprocessing and Segmentation and Measuring the parameters were described in paragraph 3.2. For the alert unit we developed a method which can predict the likelihood of depression based on the acoustic phonetic parameters; this is described in paragraph 4.1.

A. Alert unit

Each acoustic-phonetic parameter is given as a function of time. The main question is what quantity of the difference in the actual measured speech parameters indicates depressed speech. The question is whether this difference indicates the presence of depression, and, if it does, to what extent: how likely is it that this value indicates depression?

We have developed a "Depression Probability Function" (DP) which takes the actual measured differences (compares the actual values with the mean value) of any acoustic-phonetic parameter, and suggests how likely it is that the value indicates depression at the given time. The operation of the function for a given acoustic-phonetic parameter *p* is described below.

The values of the measured *p* parameter are normalized to the mean of measured values from the corresponding healthy database separately for each gender. That is, the average value of parameter *p* (calculated from the healthy database) is subtracted from every other value of *p*. The result of this is that the mean of the parameter *p* is zero in the normalized healthy database and different (but close to zero) in the normalized depressed database. For each *p* four reference distributions are generated from the Healthy Reference Speech Database and Seasonal Affective Disorder Speech Database: healthy male and female distributions (HD_p^{male}, HD_p^{female}), and depressed male and female distributions (DD_p^{male}, DD_p^{female}).

The normalized distributions are expressed as a percentage as follows, where *p* is the measured acoustic parameter,*s* is gender (male, female)and $HD_p^s(x)$ is the frequency of a given value (*x*) of the measured acoustic parameter:

$$\sum_{x=-\infty}^{+\infty} HD_p^s(x) = \sum_{x=-\infty}^{+\infty} DD_p^s(x) = 100\% \tag{1}$$

Let the measured mean value of parameter *p* on the speech sample be *x*. Two variables, "Normal Chance" (NC) and "Depressed Chance" (DC), are calculated, which indicate the likelihood that the measured value *x* is derived from the HD and from the DD respectively. These variables are calculated differently according to the sign of the difference between the mean values in HD and DD. Let us assume that the difference of mean values for the depressed distribution is less than 0 (mean of HD – mean of DD). In this case DC and NC are calculated as follows:

$$NC_p(x) = \sum_{y=x}^{-\infty} HD_p^s(y) \tag{2}$$

$$DC_p(x) = \sum_{y=x}^{+\infty} DD_p^s(y), \tag{3}$$

where $NC_p(x)$ and $DC_p(x)$ is the Normal Chance and Depressed Chance for measured value *x* of acoustic parameter *p*. $HD_p^s(y)$ and $DD_p^s(y)$ are the probability values associated with *y* from the healthy and depressed distributions for gender *s* and parameter *p*. For a given *x* only the corresponding gender is used.

If the difference of mean values for the depressed distribution is greater than 0 (mean of HD – mean of DD), then the -∞ and + ∞ are switched in the sums. The final measure of the likelihood of depression (Depression Probability, DP) for parameter *p* is calculated using NC and DC.

$$DP_p(x) = DC_p(x) - NC_p(x), \text{ if } DP_p(x) > 0, \text{ else } DP_p(x) = 0 \tag{4}$$

where $DP_p(x)$ is the likelihood of depression for parameter p for measured value *x*.

This calculation is done for all acoustic-phonetic parameters.

The Mean Depressed Probability Function (MDP) for one person for a given time is calculated in the following way:

$$MDP(x_i) = \frac{\sum_{i=0}^{n} DP_{p_i}(x_i)}{n} \tag{5}$$

where *n* is the count of the selected parameters.

This mean function has the advantage that it reduces inaccuracies caused by the natural variation of the parameters, assuming that the variables have independent diversity. Of course it may be that the simple mean is not the best, because some acoustic phonetic parameters can be more relevant than others. But, as we described in paragraph 3.3, all of these parameters seem equally important on the basis of principal component analysis. (At a later stage, when more data is available, weighted averages will improve the quality of the alert unit.)

In the case of monitoring, the MDP is measured during monitoring time at a given time rate.

5. Results

For the final examination AR, SR, TLoP parameters were excluded, as only the "tale" parts were used in the examination. However, because the subjects always read the same text, it naturally became faster and faster. We excluded VI and VF0 too, because their variances were too large, and showed no relevant results at all. Thus for the final examinations the following parameters were used: F0, F1, VF1, F2, VF2, Jitter, Shimmer, and RoT.

The MDP was calculated using Seasonal Affective Disorder Speech Database, and Healthy Speech Database, The evaluation was done on the basis of the final selected parameters for each person for the following databases: Healthy Follow-Up Speech Database, Concordia Speech Database 2013, and Concordia Speech Database 2014. While MDP was calculated on Hungarian databases, it is absolutely correct to use the Hungarian Healthy Follow-Up Speech Database for the evaluation. However our results on Hungarian language was compared with results in different European languages [24], [25], [26]. We found that the tendencies were the same for all the selected parameters, for each European languages used in Concordia Speech Databases. This is natural, since the changes of the studied parameters reflects in general human biological process and does not depend on the language. Thus probable we can use the results of the Hungarian databases to indicate depression in the Concordia Speech Database 2013 and Concordia Speech Database 2014.

A. Determination of an alert threshold for the detection of depression by the Mean Depressed Probability Function

For the determination of an alert threshold for the detection of depression the Healthy Follow-Up Speech Database was used. For each person in the Healthy Follow-Up Speech Database the MDP was measured for the finally selected parameters as a function of time.

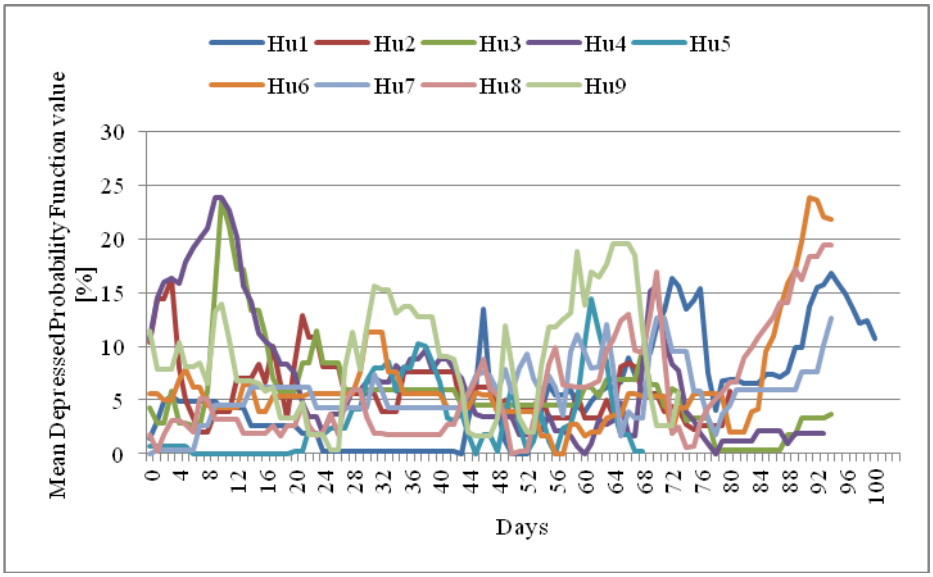


Figure 3: The Mean Depressed Probability Function values on the Healthy Follow-Up Speech Database

As can be seen in *Fig. 3.*, the maximum value of MDP was below 25%. Thus it can be stated that the value of the MDP is equal to or less than 25% in the case of healthy speech. Due to the small number of people, in order to make the method more robust the alert level has been set higher, at 35%.

B. Examination of the Mean Depressed Probability Function on the Concordia Crew Members Speech Database 2013 and 2014

The MDP values were calculated for Concordia Speech Database 2013 and Concordia Speech Database 2014. Day 0 was always 1st January of the year in question. The beginning of the winter was around day 150. Unfortunately, due to technical problems with Concordia Speech Database 2013, we only have data from day 60 (beginning of March). In both cases, the subjects arrived at the station around day 30 (beginning of December). The first values were the values of the reference records before arrival at Concordia Station, under normal circumstances in Europe.

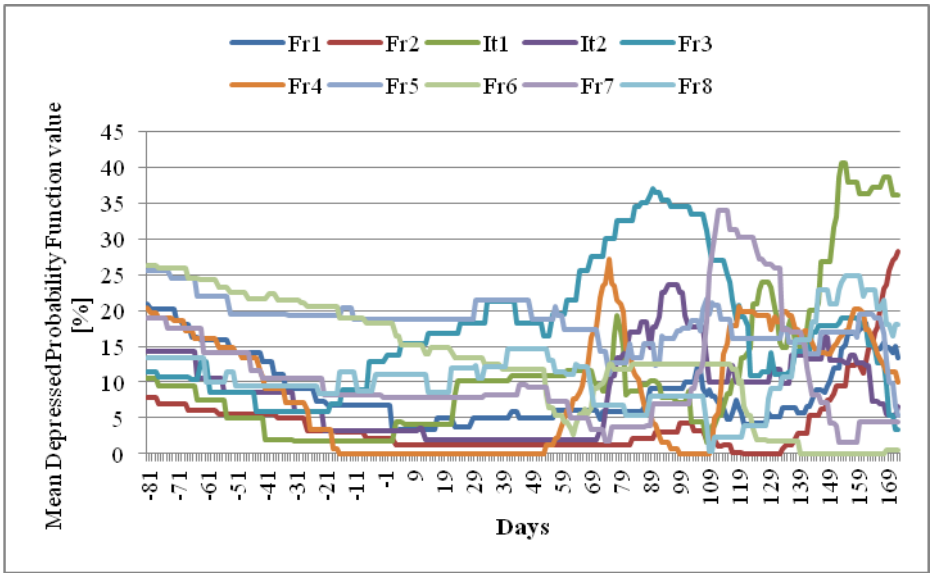


Figure 4: The Mean Depressed Probability Function values on the Concordia Speech Database 2013

As can be seen in *Fig. 4.*, there were two people where MDP value was above 35% (It1 and Fr3). It is noteworthy that several subjects' initial MDP level is around 20-25%, which is below alert level, but the acoustic parameters show large variations. The cause of this may be that the value of MDP not only indicates depression but also other cognitive states, such as stress.

Fr3's maximum MDP value was 36%, reached around April. This could be a sign some kind of depression or bad mood. It is probably not a sign of SAD, because it was at its maximum in April, became normal around May, and then remained there for the rest of the examination (till the middle of the winter): SAD is typical of the winter.

It1's maximum MDP value was 40%, reached around the beginning of June, which coincides with the beginning of winter. The depression probability value stayed around 35-40% during the rest of the examination (till the middle of the winter).

So it might be that there was only one person who suffered from SAD amongst the Concordia Crew members in 2013. Presumably the severity of the depression for this person was not high.

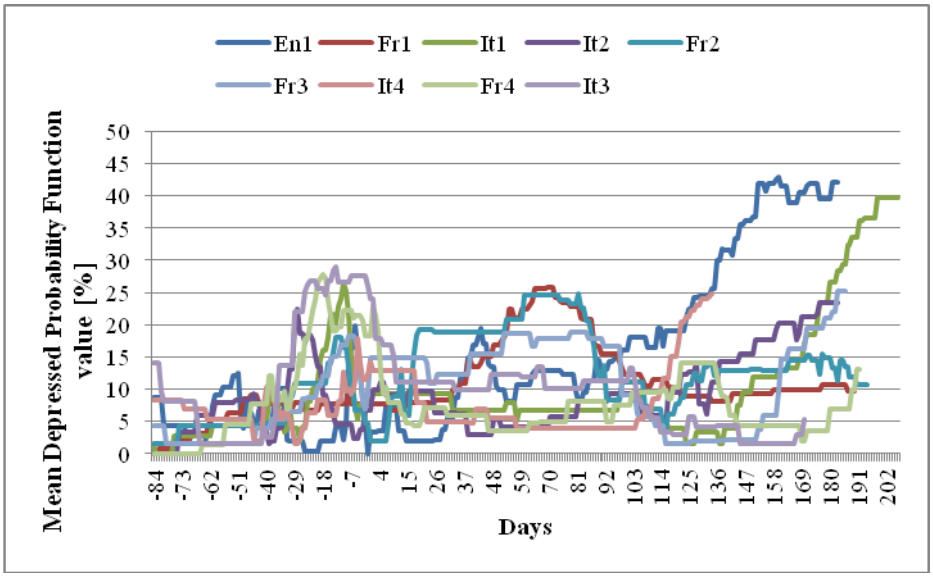


Figure 5:The Mean Depressed Probability Function values on the Concordia Speech Database 2014

As can be seen in *Fig. 5.*, there were two people whose MDP Function values were above 35% (It1 and En1). Here, and in contrast to the 2013 results, the starting MDP values were low for all the subjects, below 15%. It can be observed that on arrival at Concordia station several subjects' MDP value was relatively high: presumably this was caused by the change of environment.

En1's maximum depression probability value was 43%, reached around the end of May. This coincides with the beginning of winter. The depression probability value stayed around 39-44% during the rest of the examination (till the middle of the winter).

It1's maximum depression probability value was 40%, reached around the beginning of August, which coincides with the end of the winter, but the value started to rise from the middle of the winter (darkest day of the year). The depression probability value stayed around 40% during the rest of the examination (till the beginning of the spring).

So in 2014 it might be there were two people who suffered from SAD amongst the Concordia Crew. Presumably the severity of the depression for these subjects was not high, either.

6. Conclusions, future tasks

In this study we reviewed an online monitoring system to monitor the psychological condition of the Concordia Station's crew members and to give alerts if any vocal disorders occur that could be a sign of cognitive dysfunction (especially SAD). We analysed segmental and supra-segmental acoustic-phonetic parameters from continuously read speech in order to show significant differences between the speech of depressed people and that of healthy people. In the speech of depressed people, we found that segmental parameters, such as fundamental frequency, F1, F2 formants frequencies, jitter, and shimmer, and supra-segmental parameters such as speech rate, length of pauses, intensity and fundamental frequency dynamics, show significant changes.

We examined the normal variety of the selected acoustic phonetic parameters and we found relatively large values (as big as the mean differences of the given parameter between healthy and depressed groups).

Our database is under continual expansion because the number of depressed people examined is underrepresented compared to the incidence of depression in the population. We are expanding our database with further people speaking in different languages as their mother tongue, in order to perform a full analysis and select a complete set of acoustic features that will enable more precise conclusions to be drawn. Our aim is to find a clear correlation between the severity of depression and the change of acoustic-phonetic parameters.

For further analysis free speech will also be used, in addition to the read folk tale, from the same people. The free speech recordings are stored in both databases, Seasonal Affective Disorder Speech Database and in Concordia Speech Database 2013 and Concordia Speech Database 2014.

As the result of this study, we suggest a method that can calculate depression probability as a function of time. We have found that, according to our method, a depression probability value below 25% should be considered as normal. We have monitored two groups, and in each group we have found people suffering from SAD. It would be very useful to verify our method results with other results, but sadly we are still waiting for LTMS data.

Acknowledgements

The authors would like to thank the COALA project: Psychological Status Monitoring by Computerised Analysis of Language phenomena (COALA) (AO-11-Concordia).

References

[1] Goberman, A. M. (2005), "Correlation between Acoustic Speech Characteristics and Non-speech Motor Performance in Parkinson's Disease", *Med Sci Monit*; vol. 11(3): pp. 109–116, 2005.

[2] Goberman, A. M., McMillan, J., "Relative Speech Timing in Parkinson's Disease", *Commun Science Disord*, vol.32, pp. 22–29, 2005.

[3] Metter, E. J., Hanson, W. R., "Clinical and Acoustical Variability in Hypokinetic Dysarthria", *Commun Disord*,vol. 19(5), pp. 347–366, 1986.

[4] Coppens, E., Van Audenhove, C., Scheerder, G., Arensman, E., Coffey, C., Costa, S., Koburger, N., Gottlebe, K., Gusmão, R., O'Connor, R., Postuvan, V., Sarchiapone, M., Sisask, M., Székely, A., van der Feltz-Cornelis, C., Hegerl, U., "Public attitudes toward depression and help-seeking in four European countries baseline survey prior to the OSPI-Europe intervention", *Journal of Affective Disorders*, vol 150(2), pp. 320–329, 2013.

[5] Ivry, R. B., Justus, T. C., Middleton, C., "The cerebellum, timing, and language: Implications for the study of dyslexia" In *M. Wolf (ed.), Dyslexia Fluency and the Brain. Timonium, MD: York Press*, pp. 198–211, 2001.

[6] Esposito, A., Bourbakis, N., "The role of Timing in Speech Perception and Speech Production Processes and its Effects on Language Impaired Individuals." *IEEE Computer Society, Sixth Symposium on BioInformatics and BioEngineering (BIBE'06)*, pp. 348–356, 2006.

[7] Vicsi, K., Sztahó, D., "Problems of the Automatic Emotion Recognitions in Spontaneous Speech; An Example for the Recognition in a Dispatcher Centre", In: *Esposito, A., Martone, R., Müller, V., Scarpetta, G., Toward Autonomous, Adaptive, and Context-Aware Multimodal Interface, Theoretical and Practical Issues. (6456) Heidelberg: Springer*, pp. 331–339, 2011.

[8] Tóth, Sz. L, Sztahó, D, Vicsi, K., "Speech Emotion Perception by Human and Machine", In: *Proceeding of COST Action 2102 International Conference: Revised Papers in Verbal and Nonverbal Features of Human-Human and Human-Machine Interaction. Patras, Springer*, pp. 213–224, 2007.

[9] France, D. J, Shiavi, R. G, Silverman, S., Silverman, M., Wilkes, D. M., "Acoustical properties of speech asindicators of depression and suicidal risk", *IEEE Transactions of Biomedical Engineering*, pp. 829–837, 2000.

[10] Cannizzaro, M., Harel, B., Reilly, N, Chappell, P, Snyder, P. J., "Voice acoustical measurement of the severity of major depression", *Brain and Cognition*, vol. 56, pp. 30–35, 2004.

[11] Cannizzaro, M, Reilly, N, Mundt, J. C, Snyder, P. J., "Remote capture of human voice acoustical data by telephone: A methods study", *Clinical Linguistics & Phonetics*, vol. 19, pp. 649–658, 2005.

[12] Garcia-toro, M., Talavera, J. A, Saiz-Ruiz, J., Gonzalez, A., "Prosody impairment in depression measured through acoustic analysis", *The Journal of Nervous and Mental Disease*, vol. 188, pp. 824–829, 2000.

[13] Askenfelt, A., Sjölin, A., "Voice analysis in depressed patients: Rate of change of fundamental frequency related to mental state", *Dept. for Speech, Music and Hearing, Quarterly Progress and Status Report*, vol. 21(2), pp. 71–84, 1980.

[14] Daniel J. et al, "Acoustical Properties of Speech as Indicators of Depression and Suicidal Risk", *IEEE Transactions on Biomedical Engineering*, vol. 47(7), pp. 829–837, 2000.

[15] Thaweesak Y. et al, "Characterizing Sub-Band Spectral Entropy Based Acoustics as Assessment of Vocal Correlate of Depression", *International Conference on Control, Automation and Systems*, pp. 1179–1183, Oct. 2010.

[16] Terapong B. et al, "Assessment of Vocal Correlates of Clinical Depression in Female Subjects with Probabilistic Mixture Modelling of Speech Cepstrum", *2011 11th International Conference on Control, Automation and Systems*, pp. 387–391, Oct. 2011.

[17] James C. et al, "Voice acoustic measures of depression severity and treatment response collected via interactive voice response (IVR) technology", *J Neurolinguistics*, vol. 20(1), pp. 50–64, Jan. 2007.

[18] Elliot M. et al, "Investigating the Role of Glottal Features in Classifying Clinical Depression", *Proceedings of the 25th Annual International Conference of the IEEE*, pp. 2849–2852, 2003.

[19] Michelle Hewlett Sanchez, et al, "Using Prosodic and Spectral Features in Detecting Depression in Elderly Males", in *Proc. INTERSPEECH 2011*, pp. 3001–3004, 2011.

[20] Kiss, G., Sztahó, D., Vicsi, K., "Language independent automatic speech segmentation into phoneme-like units on the base of acoustic distinctive features." In: *4th IEEE International Conference on Cognitive Infocommunications – CogInfoCom*, pp. 579–582, 2013.

[21] Abela, J. R. Z., D'Allesandro, D. U., "Beck's cognitive theory of depression: The diathesis-stress and causal mediation components." *British Journal of Clinical Psychology*, vol. 41, pp. 111–128, 2002.

[22] Kiss, G., Vicsi K., "Physiological and cognitive status monitoring on the base of acoustic-phonetic speech parameters", In: *L Besacier, A-H Dediu, C Martín-Vide (ed.) Lecture Notes in Computer Science: Statistical Language and Speech Processing*, pp.120-131, 2014.

[23] Kiss, G., Sztahó D.,Vicsi, K., Golemis, A., "Connection between body condition and speech parameters - especially in the case of hypoxia", In: *5th IEEE International Conference on Cognitive Infocommunications – CogInfoCom*, pp. 333–336, 2014.

[24] Cummins, N., Sethu, V., Epps, J., Krajewski, J., "Probabilistic Acoustic Volume Analysis for Speech Affected by Depression", In: *15th Annual Conference of the International Speech Communication Association - InterSpeech*, pp. 1238–1242, 2014.

[25] Bozkurt, E., Toledo-Ronen, O., Sorin, A., Hoory, R., "Exploring Modulation Spectrum Features for Speech-Based Depression Level Classification", In *15th Annual Conference of the International Speech Communication Association - InterSpeech*, pp. 1243–1247, 2014.

[26] Hönig, F., Batliner, A., Nöth, E., Schnieder.S., Krajewski, J., "*Automatic Modelling of Depressed Speech: Relevant Features and Relevance of Gender*", In:*15th Annual Conference of the International Speech Communication Association – InterSpeech*, pp. 1248–1252, 2014.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 7 (2015) 80-92

Half and Fully Automatic Character Identification in Movies Based on Face Detection

Gábor SZŰCS[1], Borbála MAROSVÁRI[2]

[1] Department of Telecommunications and Media Informatics,
Faculty of Electrical Engineering and Informatics,
Budapest University of Technology and Economics, Hungary, e-mail: szucs@tmit.bme.hu
[2] Department of Telecommunications and Media Informatics,
Faculty of Electrical Engineering and Informatics,
Budapest University of Technology and Economics, Hungary, e-mail: bmarosvari@yahoo.com

Manuscript received April 18, 2016

Abstract: In this paper we present our solution for character annotation in videos based on the faces of the actors. We have developed a fully automatic annotation system without any end user intervention and a half automatic one with possibility of end user's interaction. For this task we have used face detection, face recognition and we have improved them by face aligning. Another contribution is the developed clustering procedure for the extracted features of actors' faces. The key question regarding the identification of characters is whether visual information of the actors can be gathered beforehand or not. For the former case (half automatic annotation) we used Fisherfaces face recognition, as a supervised machine learning algorithm; for the latter we developed our fully automatic face identification method, and added a quick mapping and checking feature (which belongs to half automatic annotation as well) to increase the efficiency of the system by user inputs.

Keywords: annotation of film; clustering; face detection; face recognition; Fisherfaces

1. Introduction

Nowadays films are becoming increasingly complex with complicated story lines and with many characters. In the case of movies with numerous characters the individual characters are often challenging to follow, and because of this the movies can be harder to understand. Our aim was to develop an automated annotation process for following movie characters based on their faces. This system would store the appearances and disappearances of characters in the film in order to help with following fictional characters. This annotation is useful for

understanding, analyzing, archiving certain details of the film. The main scientific challenge in this problem was the face detection and recognition components to identify the different characters in the film.

An ideal face detection system should find all the faces independent of size, shape, and relative positions of the faces in the images. There are several face detection algorithms searching faces utilizing different methods. Real time face detection and tracking is already solved in normal indoor settings, but in outdoor environments (and under other special circumstances, which often occur in films) the algorithms are inaccurate [11].

Humans are able to detect, recognize, distinguish faces of different persons and recall them for the rest of their lives [14]. In computer vision finding and identifying faces are often met with difficulties. Because of different poses, different relative positions of the camera and the face, the images of the same person will look different. Depending on the perspective, the same face can be perceived as completely differently shaped objects. Some parts of the face may become completely obscured by hair, or other objects. Facial expressions strongly influence the look of the face, as well as glasses, facial hair and clothing, all of which can cause difficulties during detection. The circumstances under which the photos are taken such as lighting, focus and white balance affect the quality of the picture and the accuracy of face recognition.

2. Related works

2.1. Face detection algorithms

Face detection can be performed by observing different attributes. For finding faces skin color can be an indicator in colored pictures, in videos movement detection can be used, or generally we can look for face-like objects and structures, and of course the combinations of these. The most successful algorithms are appearance-based, without using additional cues.

Face detection algorithms can be classified into four categories, which do not have clear boundaries so they can overlap.

Knowledge based: These rule-based methods work with creating rules from human knowledge of what constitutes as a face. These rules describe the overall look of a face, details of facial features [22]. A face model can also be used, which describes the relative distances of facial features [18]. These methods are mainly used for face localization.

Feature invariant: These methods look for features that do not change when changes in illumination, position, perspective occur. Mainly used for face localization. These algorithms include: edge grouping [23], [10], texture analysis [6], skin color analysis [12], multiple feature analysis [7].

Template matching: Several patterns of a face are stored, describing the face as a whole or describing certain facial features. For face detection the correlation between the input image and stored patterns is computed. These methods can be used for face localization and detection alike. Methods include: predefined face templates [5], deformable face templates [9].

Appearance based, learning methods: These are the most successful algorithms. The models (or templates) are learned from a training set, which should capture the variability of facial appearances. Methods include: Eigenface [20], distribution based [19], Neural Network [16], Support Vector Machine [13], Hidden Markov-model [15], Bayes Classifier [17], AdaBoost learning based methods [21]. The latter are the most successful ones in terms of accuracy and speed [11].

2.2. Face recognition algorithms

For identifying faces feature extraction is essential. Depending on the face recognition system different features are required. The features needed to be found can be lines, or specific key points like the eyes, nose or mouth. Feature extraction can be performed during face detection, or after. In most cases face recognizers implement feature extraction. Feature extraction is also a key element in facial expression recognition.

Feature extraction is also necessary in systems that observe the faces in whole and create their own feature vectors. Algorithms like Eigenfaces [20] and Fisherfaces [3] need the exact positions of the eyes, nose or mouth in order to allow face normalization [24].

Face recognition algorithms can be divided into two groups, pose-dependent and pose-independent [11]. The difference between the two types is the representation of the face. Pose-dependent methods analyze faces from the viewer's point of view, while the pose-independent, object oriented methods work with 3D models of the face, thus becoming independent of the position of the face.

Pose-dependent algorithms can be further divided into two or three groups [24], [11].

Holistic algorithms: observing the faces as a whole. These include methods that use PCA like Eigenfaces and Fisherfaces. The local or analytic, feature based methods include geometric methods, HMMs, LBP histogram methods. The third class is the Hybrid methods: Elastic Bunch Graph Matching, Hybrid Local Feature Analysis, and modular Eigenfaces.

3. Improvement by face aligning

In order to align the pictures of the faces we need to determine the positions of the eyes. For finding the eyes we used two Viola-Jones detectors implemented in OpenCV [8], trained with "lefteye" and "righteye" cascade classifiers respectively [4]. We achieved the best results using these two detectors together from the OpenCV library.

These cascade classifiers were created by using 7000 positive samples for 18×12px size. The "lefteye" classifier is intended to find the left eye, as the "righteye" classifier is meant to find the right eye in the picture (as some examples can be seen in *Fig. 1*).

During testing, out of 591 cases the detectors found the eyes correctly only in one case, meaning the "lefteye" found the left eye and the "righteye" the right eye. In the majority of the test cases each detector found both of the eyes. In higher resolution pictures there were great amounts of false positives. Thus determining the true positives for each eye became difficult.

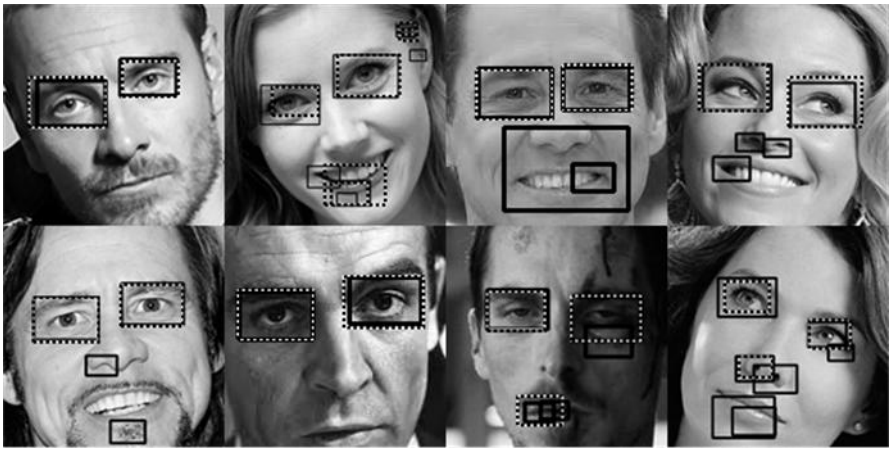


Figure 1: Results of "lefteye" and "righteye" classifiers

The eye-detectors are run in the face's upper 4/7 area; this immediately lowers the number of false positives. An eye was considered detected only when the two detectors found the same area: the center of one rectangle is within the other, and vice versa. The face is rotated when the following conditions are met: exactly 2 eyes are detected, the distance of the eyes is larger than the 1/9 of the face's width, and the rotation angle is less than ±35°. Using this method the detected eye-pairs were all true positives, out of the 591 test cases 437 correct rotations and eye-pair detections were achieved.

We tested our method on OpenCV's face recognition implementations. The test set consisted of 591 test photos of 20 persons, with one face on each photo. Beside the test set, the training set contained 10 photos for each person, aligned at eye level. The results can be seen in *Table 1*.

Table 1: Results of face recognition on 20 subjects with different preprocessing

	Correct identification	Incorrect identification	Accuracy
Eigenfaces	97	494	0.164
Fisherfaces	169	422	0.285
LBPH	129	462	0.218

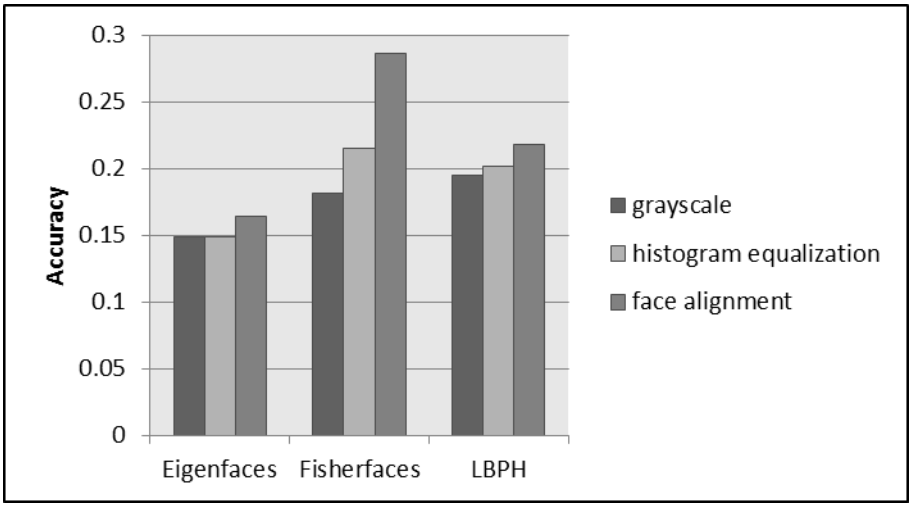


Figure 2: Results of face recognition on 20 subjects with different preprocessing

Fig. 2 shows the results of the different preprocessing methods, grayscale, histogram equalization (on grayscale images), and face alignment (on grayscale and histogram equalized images).

All three algorithms produced better results with face alignment than without. Correlation between the improvement or failure in recognition and the angle of rotation could not be observed.

4. Character identification

The key question regarding the identification of fictional characters in videos is whether there is visual information available on the characters or not. If this information can be used then the faces of actors should be gathered (e.g. from images or from the movie) and the training image set can be determined by

drawing the bounding boxes of the faces. The supervised character identification can be processed after training, and the time of appearances and disappearances of the characters can be determined in the film.

With unsupervised character identification there is no visual information about characters. In such case the identification can be achieved using information extracted only from the movie, labeling characters simply as "character A", "character B", etc. Differentiating between characters could be accomplished by using the first detected face as a training example, comparing subsequent faces to it and adding the faces to the training set to a new or existing character. But this method is practically unusable due to the low number of training images per person. Instead, in order to distinguish between different characters and to group the detected faces of same character together, our solution is based clustering. In an ideal situation the number of the clusters is equal to the number of the characters in the video. The time stamps of the clusters can be used for determining the duration of each character's presence (for annotation). For the unsupervised character identification problem we have developed a method – so called fully automatic character identification method – by clustering.

Our solution is divided into two parts; the first part extracts all the faces of the characters and stores them, the second part creates feature vectors from them, after which these vectors are clustered.

In the second part our method extracts LBP (Local Binary Patterns) feature vectors. In the basic methodology for LBP based face description (Ahonen et al., 2006) the part of the image where the face is located is divided into local regions and LBP descriptors are extracted from each region independently. The occurrences of the LBP codes in a region are collected into a histogram, which creates the local LBP descriptor. These local descriptors are then concatenated to form a global description of the face. These local feature based methods are more robust against variations in pose or illumination than holistic methods. This histogram effectively has a description of the face on three different levels of locality: the LBP labels for the histogram depict the patterns on pixel-level, the labels are summed over a small region to produce information on a regional level and the regional histograms are concatenated to build a global description of the face. The sample ratio parameter was 8 and the surrounding parameter of LBP was 2 in our application, which is implemented in OpenCV (Laganière, 2011).

A critical issue when clustering the LBP vectors in the next step was determining the number of clusters. The most often used clustering algorithm is the k-means, but as a parameter it requires the number of the clusters. Instead of this well-known algorithm we have used hierarchical k-means (Arai and Barakbah, 2007), which is the combination of k-means and hierarchical

clustering algorithm. The advantage of this hierarchical version of k-means is that it defines the initial centroids for standard k-means. In our solution the distance function of the clustering was the Euclidean distance (L2 norm) of LBP vector. For determining the number of the clusters we have used a rule of thumb, i.e. the root square of n/2, where n is the number of the vectors.

5. User supported and automatic annotation

Based on our fully automatic character identification method we have implemented an automatic annotation program. For the solution of the automatic annotation we have applied the method in *Fig. 3*, described in the previous section. The annotation program uses the generated clusters of the faces for the annotation of the characters. Based on the time stamps of a character's images the program collects and calculates the times of appearances and disappearances. These time durations are then stored in hh:mm:ss format, and other additional information (e.g. bounding box of the face) is also stored for possibilities of later analysis. The final annotation contains the screen time duration for each character.

With a half-automatic annotation program the user could provide external (additional) information and can improve the character identification task in the annotation, so we have developed half automatic character identification method with possibility of extra information. This extra information can be the character's name; the end user can input the names, and can map these to the list of durations. A further possibility in our half-automatic annotation program is quick mapping and checking. We have implemented an average image generating process for facial image clusters, in which the process converts the images into the same size and accumulates the pixel values in different pictures in the same positions, then divides it by the number of images in the cluster. The average image gives the possibility to quick mapping among the characters' names and the list of durations, because the end user does not need to examine numerous facial images, the average image is representative of all of them. Another advantage of using an average image is the possibility of correction. If the image cluster would contain more than one character, then the average image would not be recognized. In this case the end user can mark the unrecognized facial image, and the corresponding list of durations will be excluded from the annotation. The annotation can be improved by repairing these incorrect records in the list of durations. A further manual improvement possibility is merging a character's list of durations based on the average images.

6. Testing the fully and half automatic character identification

We have tested our character identification system on the movie „The Breakfast Club". Before the automatic annotation, we have created a manual annotation (list of durations for every character's presence, the characters' „screen time"), and these were compared to evaluate the system's performance. The length of the film is 1:37:05, the frame-rate is 23.976, the resolution is 1280×688 pixel, and we have extracted 11638 images by half second sampling.

The movie has 11 characters (14 in total, but the faces of only 11 can be seen). 7 characters (Allison, Andrew, Bender, Brian, Carl, Claire, Richard) can be seen in the whole length of the movie, 3 characters (Brian's mother, Brian's sister, Claire's father) are only present in the beginning, and 1 character (Andrew's father) can be seen in the beginning and the end of the movie.

6.1. Testing the fully automatic character identification

The *fully automatic character identification* method created 59 clusters from 7093 images, and for evaluation we have calculated the purity, Rand index and F_1 indicators of clustering. The resulted clusters are pure (have high purity score), but the images belonging to the same character are distributed into more clusters.

If we compared it with the case where we have given manually the number of clusters (11), then it can be seen (in *Table 2*) that F_1 is better, but the other two indicators are worse.

Table 2: Results of the unsupervised character identification

	purity	Rand index	F_1
59 clusters	0.588	0.791	0.183
11 clusters	0.414	0.702	0.240

By lowering the number of clusters to the correct number of clusters the purity decreased, but the F_1 improved. This is due to the increasing recall measure of the characters' images. In order to see the trends of these indicators we have constructed a simplified task, where the sampling frequency in the video was lower, thus we could investigate more clustering. In *Fig. 3* the trends of the purity, Rand index and F_1 indicators can be seen, where the number of clusters was the parameter of our evaluation.

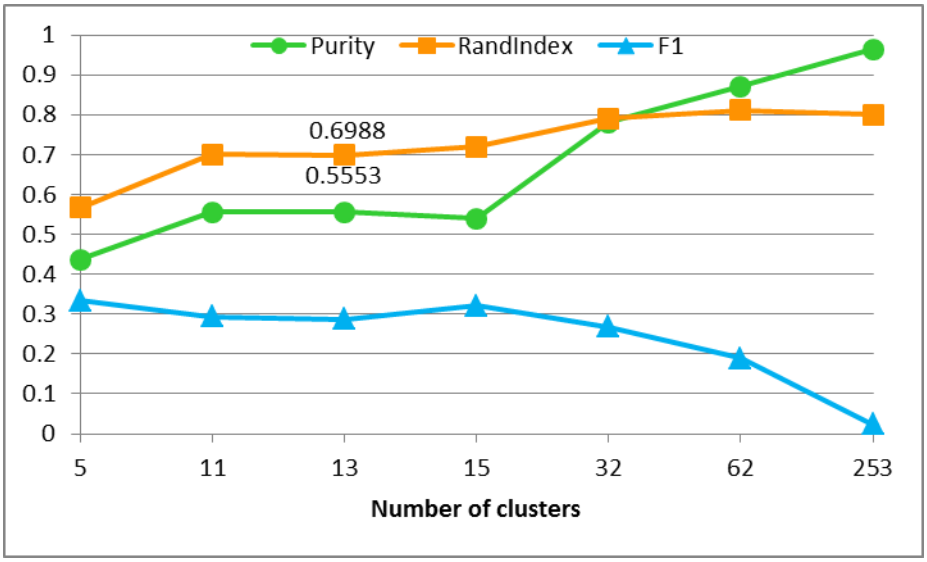


Figure 3: Comparison of clustering with different number of clusters

6.2. Half automatic character annotation – testing the supervised character identification

In order to test the supervised character identification method we created a learning set, in which there are 10 pictures for each character. These pictures were extracted from the movie to be tested, simulating the aforementioned method of drawing the bounding boxes of the characters and marking the positions of the eyes. These images were cropped and aligned using OpenCV's crop_face.py program to create the training set for the supervised character identification.

At the evaluation (as can be seen in *Table 3*) the manual annotation and the results of our character annotation system were compared. At the evaluation we have calculated the recall, the precision, and the harmonic mean of them (F_1), as the most frequent used indicators in the literature. Regarding a character the sum of the intersection of time of manual and automatic annotation system is the TP (true positive), the sum of only the manual ones is the P (i.e. this screen times consists of the sum of the true positives and false negatives: TP+FN), and the sum of intervals at only our system is the predicted P (i.e. TP+FP); F_1 can be determined from them.

Table 3: Results of the supervised character identification

Character	P	Recall	Precision	F_1
Allison	0:13:29	0.0816	0.1119	0.0944
Andrew	0:20:53	0.1979	0.7209	0.3106
Andrew's father	0:00:15	0.5333	0.0050	0.0099
Bender	0:22:51	0.2531	0.6686	0.3672
Brian	0:17:15	0.2676	0.4009	0.3210
Brian's mother	0:00:06	0.8333	0.0515	0.0971
Brian's sister	0:00:03	0	0	0
Carl	0:01:57	0.3419	0.5063	0.4082
Claire	0:13:10	0.3278	0.5396	0.4079
Claire's father	0:00:21	0.0476	0.0357	0.0408
Richard	0:07:04	0.4198	0.3090	0.3560

In order to compensate for the possible errors in the manual annotation, the detected faces were also individually inspected to calculate precision. Out of 7093 detections 6370 were correctly detected faces, the rest were the face detector's FP cases. Out of the 6370 faces 3085 were correctly identified by the Fisherfaces face recognizer, which means a precision score of 0.484.

Analyzing the *Table 3* and the collected faces we can assess our findings as follows. Brian's sister, the character with the shortest screen time was completely missed by the face detector, despite of her screen time being continuous, thus the sampling frequency being adequate, and also another character was detected on the same frame where Brian's sister was visible.

Brian's mother's and Andrew's father's recall values show that the program correctly identified their faces, the low precision scores show that there were many faces incorrectly identified as them. Claire's father's low recall value can be explained by that the face detector could only find his face 4 times during 21 seconds with 0.5 second sampling rate. Out of these 4 detected faces only 1 was correctly identified by the face recognizer, hence the low precision value. Allison's low scores can be explained by her appearance, her hair was usually obstructing her face, which made detecting and recognizing her face hard. The characters with the longest screen times, Andrew's and Brian's low recall values come from the error of the face detector, because their precision score is quite high, so when their faces were found, they were also identified correctly.

6.3. Half automatic character annotation with quick mapping and checking

Based on our *fully automatic character identification* method the half-automatic annotation program generates average images for facial image clusters for *quick mapping and checking*. An end user, who has seen the movie, is asked to map the average images to the characters. At the end of the mapping the images of the same characters were aggregated into one cluster, as shown in *Fig. 4* on the character called Bender. Only a few average images were mapped to nothing, because they were unrecognizable.

Figure 4: Average images of clusters mapped to a character (Bender)

After the user supported mapping the effectiveness of the system is expected to be better.

Table 4: Results of the half-automatic annotation after user supported mapping

Character	Number of mapped clusters	P	TP	TP+FP	Recall	Precision	F1
Allison	10	0:13:29	0:05:15	0:07:12	0.3894	0.7292	0.5077
Andrew	13	0:20:53	0:05:19	0:07:22	0.2546	0.7217	0.3764
Andrew's father	0	0:00:15	0:00:00	0:00:00	0	0	0
Bender	10	0:22:51	0:07:12	0:09:04	0.3151	0.7941	0.4512
Brian	5	0:17:15	0:02:13	0:03:18	0.1285	0.6717	0.2157
Brian's mother	0	0:00:06	0:00:00	0:00:00	0	0	0
Brian's sister	0	0:00:03	0:00:00	0:00:00	0	0	0
Carl	1	0:01:57	0:00:17	0:00:20	0.1453	0.8500	0.2482
Claire	10	0:13:10	0:04:47	0:07:51	0.3633	0.6093	0.4552
Claire's father	0	0:00:21	0:00:00	0:00:00	0	0	0
Richard	5	0:07:04	0:02:13	0:02:59	0.3137	0.7430	0.4411
Summarized	**54**	**1:37:24**	**0:27:16**	**0:38:06**	**0.2799**	**0.7157**	**0.4025**

After the user supported mapping the clusters of unrecognized average images are filtered out, as it can be seen in Table 4; from 59 clusters only 54 clusters remained. The purity values of the 5 omitted clusters are low: 0.28, 0.42, 0.19, 0.3, 0.35. It can be concluded that during filtering clusters the false decisions (images not containing faces) are also filtered out, thus the system is more accurate (F_1 reaches higher value).

7. Summary

In books, as well as in films it is important to evoke reader sympathy in order to draw your audience into the story. But if the audience cannot follow the many characters in the movie, then this sympathy will not develop in the audience. Our work on character detection and identification (for annotation) in films is a large step in understanding and analyzing the story told.

In this paper we have presented a character annotation solution for videos. We have developed a fully automatic annotation system without any end user intervention and a half automatic one with possibility of end user's interaction. For this task we have used face detection, face recognition and we have improved them by face aligning. Another contribution is the developed clustering procedure for the extracted features of actors' faces.

For the half automatic annotation we have used Fisherfaces face recognition, as a supervised machine learning algorithm. For fully automatic annotation the possible method is clustering, and we have developed a *fully automatic face identification* method using it. In order to increase the efficiency of the system by user inputs a new feature, so called *quick mapping and checking* feature was added to the system.

References

[1] Ahonen, T., Hadid, A., Pietikäinen, M., "Face Description with Local Binary Patterns: Application to Face Recognition", *IEEE Trans. Pattern Analysis and Machine Intelligence* 28(12), 2006, pp. 2037–2041.

[2] Arai, K., Barakbah, A. R., "Hierarchical K-means: an algorithm for centroids initialization for K-means", *Reports of the Faculty of Science and Engineering*, 36(1), 2007, pp. 25–31.

[3] Belhumeur, P. N., Hespanha, J. P., Kriegman, D. J., "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, 1997, pp. 711–720.

[4] Castrillón-Santana, M., Déniz-Suárez, O., Antón-Canalís, L., Lorenzo-Navarro, J., "Face and Facial Feature Detection Evaluation", *Third International Conference on Computer Vision Theory and Applications*, VISAPP08, 2008.

[5] Craw, I., Tock, D., Bennett, A., "Finding Face Features", *Proc. Second European Conf. Computer Vision*, 1992, pp. 92–96.

[6] Dai, Y., Nakano, Y., "Face-Texture Model Based on SGLD and Its Application in Face Detection in a Color Scene", *Pattern Recognition*, Vol. 29, No. 6, 1996, pp. 1007–1017.
[7] Kjeldsen, R., Kender, J., "Finding Skin in Color Images", Proc. *Second Int'l Conf. Automatic Face and Gesture Recognition*, 1996, pp. 312–317.
[8] Laganière, R., "OpenCV 2 Computer Vision Application Programming Cookbook: Over 50 recipes to master this library of programming functions for real-time computer vision", Packt Publishing Ltd., 2011.
[9] Lanitis, A., Taylor, C. J., Cootes, T. F., "An Automatic Face Identification System Using Flexible Appearance Models", *Image and Vision Computing*, vol. 13, no. 5, 1995, pp. 393–401.
[10] Leung, T. K., Burl, M. C., Perona, P., "Finding Faces in Cluttered Scenes Using Random Labeled Graph Matching", *Proc. Fifth IEEE Int'l Conf. Computer Vision*, 1995, pp. 637–644.
[11] Li, S. Z., Jain, A. K., "Handbook of Face Recognition", New York: Springer, 2005.
[12] McKenna, S., Gong, S., Raja, Y., "Modelling Facial Colour and Identity with Gaussian Mixtures", *Pattern Recognition*, Vol. 31, No. 12, 1998, pp. 1883–1892.
[13] Osuna, E., Freund, R., Girosi, F., "Training Support Vector Machines: An Application to Face Detection", *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997, pp. 130–136.
[14] Parr, L. A., "The evolution of face processing in primates", *Philosophical Transactions of the Royal Society B: Biological Sciences*., Vol. 366 No. 1571, 2011, pp. 1764–1777.
[15] Rajagopalan, A., Kumar, K., Karlekar, J., Manivasakan, R., Patil, M., Desai, U., Poonacha, P., and Chaudhuri, S., "Finding Faces in Photographs", Proc. *Sixth IEEE Int'l Conf. Computer Vision*, 1998, pp. 640–645.
[16] Rowley, H., Baluja, S., Kanade, T., "Neural Network-Based Face Detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, 1998, pp. 23–38.
[17] Schneiderman, H., Kanade, T., "Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition", *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1998, pp. 45–51.
[18] Shinn-Ying, H., Hui-Ling, H., "Facial Modeling from an Uncalibrated Face Image Using a Coarse-to-Fine Genetic Algorithm", *Pattern Recognition*, Vol. 34, No. 9, 2001, pp. 1015–1031.
[19] Sung, K.-K., Poggio, T., "Example-Based Learning for View-Based Human Face Detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, 1998, pp. 39–51.
[20] Turk, M., Pentland, A., "Eigenfaces for Recognition", *Journal of Cognitive Neuroscience*, vol. 3, no. 1, 1991, pp. 71–86.
[21] Viola, P. and Jones, M., "Rapid object detection using a boosted cascade of simple features", *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, 2001, pp. 12–14.
[22] Yang, G., Huang, T. S., "Human Face Detection in Complex Background", *Pattern Recognition*, Vol. 27, No. 1, 1994, pp. 53–63.
[23] Yow, K. C., Cipolla R., "Feature-Based Human Face Detection", *Image and Vision Computing*, Vol. 15, No. 9, 1997, pp. 713–735.
[24] Zhao, W., Chellappa, R., Phillips, P. J., Rosenfeld, A., "Face Recognition: A literature Survey", ACM Computing Surveys, Vol. 35. No. 4, 2003.

Acta Universitatis Sapientiae

The scientific journal of Sapientia University publishes original papers and surveys
in several areas of sciences written in English.
Information about each series can be found at
http://www.acta.sapientia.ro.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering

Sapientia University

Scientia Publishing House

ISSN 2065-5916
http://www.acta.sapientia.ro

Information for authors

Acta Universitatis Sapientiae, Electrical and Mechanical Engineering publishes only original papers and surveys in various fields of Electrical and Mechanical Engineering. All papers are peer-reviewed.

Papers published in current and previous volumes can be found in Portable Document Format (PDF) form at the address: `http://www.acta.sapientia.ro`.

The submitted papers must not be considered to be published by other journals. The corresponding author is responsible to obtain the permission for publication of co-authors and of the authorities of institutes, if needed. The Editorial Board is disclaiming any responsibility.

The paper must be submitted both in MSWord document and PDF format. The submitted PDF document is used as reference. The camera-ready journal is prepared in PDF format by the editors. In order to reduce subsequent changes of aspect to minimum, an accurate formatting is required. The paper should be prepared on A4 paper (210×297 mm) and it must contain an abstract of 200-250 words.

The language of the journal is English. The paper must be prepared in single-column format, not exceeding 12 pages including figures, tables and references.

The template file from http://www.acta.sapientia.ro/acta-emeng/emeng-main.htm may be used for details.

Submission must be made only by e-mail (`acta-emeng@acta.sapientia.ro`).

One issue is offered to each author free of charge. No reprints are available.

Contact address and subscription:
Acta Universitatis Sapientiae, Electrical and Mechanical Engineering
RO 400112 Cluj-Napoca
Str. Matei Corvin nr. 4.
E-mail: acta-emeng@acta.sapientia.ro

Printed by Digital Color Company
www.digitalcolorcompany.ro