KOMUNIKÁCIE
COMMUNICATIONS

https://doi.org/10.26552/com.C.2000.2.4-9

komunikácie
COMMUNICATIONS

Ivan Martinček - Ivan Turek - Milan Dado - Karol Grondžák - Slavomir Černický *

INTERFERENCIA MODOV V OPTICKÝCH VLÁKNACH

INTERMODAL INTERFERENCE IN OPTICAL FIBRES

V článku sú popísané výsledky teroetickej analýzy a meraní interferencie modov v málomodových optických vláknach. Je ukázané, že interferencia modov je citlivý parameter závisiaci od profilu indexu lomu a geometrických vlastností optického vlákna. Porovnanie teoretických a experimentálnych hodnôt interferencie modov ukazuje, že meranie interferencie by mohlo byť použité na zisťovanie miery korešpondencie profilu indexu lomu vyšetrovaného vlákna a ideálneho step-indexového vlákna.

Úvod

Monochromatickú elektromagnetickú vlnu šíriacu sa optickým vlnovodom (vláknom) je možné popísať lineárnou kombináciou istých „vlastných" funkcií (alebo „modov"). Parametre šírenia sa týchto modov (rýchlosť šírenia sa, koeficient absorpcie, oblasť frekvencií pre ktoré jednotlivé mody existujú) sú vo všeobecnosti odlišné. V dôsledku toho zmena fázy signálu vyvolaná prechodom vlny cez vybraný úsek vlnovodu závisí nie len od toho, akým dlhým úsekom vlna prešla a aká je jej frekvencia, ale i od toho, ktorým modom bol signál prenášaný. V prípade, že na prenose signálu sa podieľalo viacero modov, dôjde na konci vlákna k vytvoreniu optického stavu zloženého z odlišných modálnych funkcií s odlišnými fázami.

Pri zvážení rozdielnosti fáz jednotlivých modálnych funkcií signál vytvorený kvadratickým detektorom na konci vlnovodu dĺžky l môžeme vyjadriť nasledovne:

$$s = \int_S c(x,y) \cdot \sum_i \psi_i(x,y,t) \cdot \sum_i \psi_i^*(x,y,t) \cdot dxdy \qquad (1)$$

kde $c(x, y)$ je citlivosť detektora, $\psi_i(x,y,t)$ sú funkcie popisujúce jednotlivé mody a rovnajú sa $\psi_{i,0}(x,y)$. exp $(j(\omega t - \beta_i z))$, kde $\psi_{i,0}$ sú modálne funkcie, β_i sú fázové konštanty jednotlivých modov, x a y sú súradnice kolmo na os vlnovodu, z je súradnica v osi vlnovodu, S je plocha, na ktorej sú modálne funkcie $\psi_{i,0}$ odlišné od nuly a „*" označuje komplexne združenú funkciu. Roznásobením naznačených súčtov a za predpokladu, že k detekcii bol použitý detektor s citlivosťou rovnou c_0 na celej ploche S, dostaneme po jednoduchých úpravách:

In this paper the results of a theoretical analysis and practical investigation of mode interference in single mode optical fibres are described. It is shown that inter-modal interference is a sensitive parameter depending on the quality of an index profile or geometrical parameters of optical fibre. The comparison of experimental and theoretical values of mode interference indicates that the measurement of interference could be used for testing how the real refractive index profile of investigated fibre corresponds with the refractive index profile assumed at the calculation (step-index profile).

Introduction

The monochrome electromagnetic wave propagating through the optic fibre can be described as a linear combination of eigenfunctions (modes) which are determined by the parameters of the fibres. The propagation parameters of such modes (propagation velocity, absorption coefficient, frequency band for which the mode exists) are in general different. So the change of the signal phase due to its passing through the fibre depends not only on the frequency of the signal and the length of the optic fibre, but also on the properties of the actual mode propagating through the fibre. If more than one mode propagates, the optical stage at the end of the fibre is a superposition of values with different phases.

Taking into account the different phase constants of the modes, we can express the output of the quadratic detector located at the end of the fibre of length l as follows:

$$s = \int_S c(x,y) \cdot \sum_i \psi_i(x,y,t) \cdot \sum_i \psi_i^*(x,y,t) \cdot dxdy \qquad (1)$$

where $c(x,y)$ is the detector sensitivity, $\psi_i(x,y,t)$ are the functions describing the propagating modes and are equal to $\psi_{i,0}(x,y)$. exp $(j(\omega t - \beta_i z))$, where $\psi_{i,0}$ are the modal functions, β_i are the phase constants of particular modes, x and y are the coordinates perpendicular to the direction of the propagation, z is the coordinate in the direction of propagation, S is an area in which the modal functions are nonzero and "*" denotes complex conjugation. Assuming the constant sensitivity of detector c_0 on whole area S after some manipulation we get:

$$S = c_0 \cdot \int_S \sum_i \psi_{i,0}(x,y) \cdot \psi_{i,0}^*(x,y) \cdot dxdy + c_0 \cdot \int_S \sum_{i \neq 0} \psi_{i,0}(x,y) \cdot \psi_{k,0}^*(x,y) \cdot exp(j(\beta_i - \beta_k)) \cdot dxdy \qquad (2)$$

* [1]Ivan Martinček, [1]Ivan Turek, [2]Milan Dado, [1]Karol Grondžák, [2]Slavomir Černický
[1] Department of Physics,
[2] Department of Telecommunication, Faculty of Electrotechnical Engineering, University of Žilina, Moyzesova 20, SK-01026 Žilina, Slovak Republic
E-mails: ivmar@fel.utc.sk, turek@fel.utc.sk, dado@bull.utc.sk, grondz@fel.utc.sk.

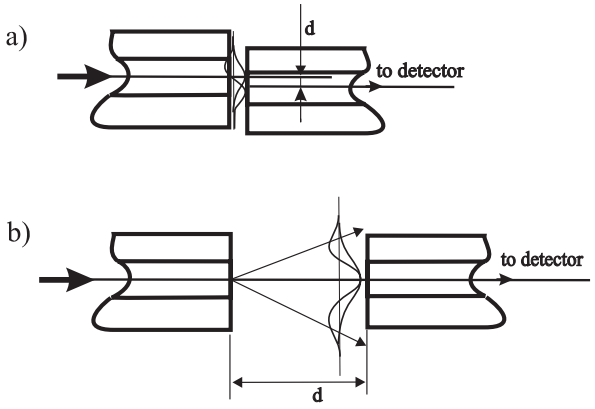
Prvý člen pravej strany rovnice (2) predstavuje signál vytvorený súčtom intenzít prenesených jednotlivými modmi, nezávisí od dĺžky vlnovodu a iba slabo závisí od vlnovej dĺžky svetla (od λ závisí iba v dôsledku závislosti modálnych funkcií od vlnovej dĺžky).

Druhý člen pravej strany rovnice (2) predstavuje „interferenčný" člen. Ako je z jeho tvaru uvedeného v rovnici (2) vidieť, jeho hodnota je rovná nule, pretože

$$\int_S \psi_{i,0}(x,y) \cdot \psi^*_{k,0}(x,y) \cdot dxdy = 0 \qquad (3)$$

pre $i \neq k$ (pretože modálne funkcie sú ortogonálne).

V prípade, že citlivosť detektora nie je v celej oblasti S rovnaká, interferenčný člen sa nemusí rovnať nule. Jeho hodnota bude závisieť od stupňa a typu jeho symetrie a charakteru príslušnej dvojice interferujúcich modov $\psi_{i,0}$ a $\psi_{k,0}$ a keď fázové konštanty interferujúcich modov nie sú rovnaké, relatívne prudko bude závisieť od dĺžky vlákna a od vlnovej dĺžky svetla, ktoré vláknom prechádza. Rovnaký výsledok by sme dosiahli, keby sa ortogonalita modov narušila zaradením vhodného priestorového filtra. Ako takýto filter sa môže použiť tienidlo s malou dierkou (pinhole) [1], vhodne umiestnené medzi vyšetrované vlákno a detektor, alebo ďalšie optické vlákno, ktorým sa signál z konca vyšetrovaného vlákna privedie na povrch detektora [2]. Nesúosovým uložením takéhoto priestorového filtra sa naruší ortogonalita symetrických a antisymetrických modov (takéto usporiadanie je vhodné pri vyšetrovaní interferencie modov LP_{01} a LP_{11}). Uložením filtra súosovo, ale vo väčšej vzdialenosti od konca vyšetrovaného vlákna sa naruší ortogonalita symetrických modov (usporiadanie vhodné napríklad pri vyšetrovaní interferencie modov LP_{01} a LP_{02}), ako to ilustrujú schémy uvedené na obr. 1.



Obr. 1 Schéma usporiadania pre vyšetrovanie interferencie a) symetrického a asymetrického; b) dvoch symetrických modov
Fig. 1 The set-up of investigation of intermodal interference of a) symmetric and astisymmetric; b) two symmetric modes

The first term on the right side of (2) represents the sum of particular mode intensities and does not depend on the length of waveguide and only weakly depends on the wavelength (the dependence is only because of the dependence of the modal functions on wavelength).

Second term on the right side of (2) is the interference term. It is clear that its value is zero, because

$$\int_S \psi_{i,0}(x,y) \cdot \psi^*_{k,0}(x,y) \cdot dxdy = 0 \qquad (3)$$

for $i \neq k$ (the modal functions are orthogonal).

In the case of non-uniform sensitivity of the detector the interference term can have non-zero value. The value will then depend on the symmetry of the interfering modes $\psi_{i,0}$ and $\psi_{k,0}$. When the phase constants (propagation velocity) of interfering modes are different, then the value will depend also on the length of the fibre and on the wavelength. For the detector with uniform sensitivity the same result can be achieved when the orthogonality of the modes is disturbed using a spatial filter which influences the spatial distribution the of the optical field amplitude. A pinhole located between the fibre and detector can be used as such a filter. A piece of another fibre transporting the signal from the end of the investigated fibre to the detector can be used for this purpose too [2].

Non axisymmetrical location of such a spatial filter will disturb the orthogonality of symmetric and anti- symmetric modes (such configuration is useful when studying the interference of LP_{01} and LP_{11} modes). When locating the spatial filter axisymmetrically, but further from the end of the fiber under study, the orthogonality of symmetric modes is disturbed (such configuration is useful for studying the interference of modes LP_{01} and LP_{02}), as can be seen in Fig. 1.

Určenie fázových konštánt

Fázovú konštantu modu pre vlákno so skokovým profilom indexu lomu možno podľa [3] vyjadriť:

$$\beta = k\left[n_{co}^2 - \frac{U^2(V)}{V^2}\,(n_{co}^2 - n_{cl}^2)\right]^{\frac{1}{2}} \qquad (4)$$

kde $U^2 = r^2\,(n_{co}^2\,k^2 - \beta^2)$, $V^2 = k^2\,r^2\,(n_{co}^2 - n_{cl}^2)$ je normovaná frekvencia, $k = 2\pi/\lambda$ je konštanta šírenia sa svetelnej vlny vo vákuu, n_{co} je index lomu jadra, n_{cl} je index lomu plášťa a r je polomer jadra.

Funkčná závislosť $U(V)$ sa získa riešením charakteristickej rovnice pre step-indexové vlákno. V priblížení slabovodivého vlákna ($n_{co} \approx n_{cl}$) pre step-indexové vlákno má táto charakteristická rovnica [4] tvar:

Mode phase constant determination

The mode phase constant for step-index optic fibre can be expressed [3]:

$$\beta = k\left[n_{co}^2 - \frac{U^2(V)}{V^2}\,(n_{co}^2 - n_{cl}^2)\right]^{\frac{1}{2}} \qquad (4)$$

where $U^2 = r^2\,(n_{co}^2\,k^2 - \beta^2)$, $V^2 = k^2\,r^2\,(n_{co}^2 - n_{cl}^2)$ is the normalised frequency, n_{co}, n_{cl} are the refractive indices of the core and the cladding respectively, r is the core radius and $k = 2\pi/\lambda$ is free-space propagation constant.

Function $U(V)$ can be obtained by solving a characteristic equation for the optic fibre. In a weakly-guided approximation ($n_{co} \approx n_{cl}$) for step-index fibre it has the form [4]:

$$U\frac{J_{m+1}(U)}{J_m(U)} = W\frac{K_{m+1}(W)}{K_m(W)} \qquad (5)$$

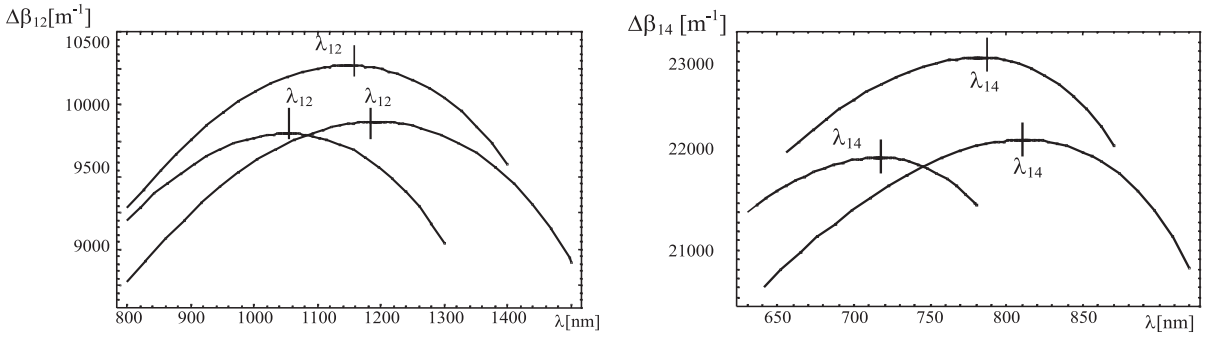
kde $m = 0,1,2,...$, J_m je Besselova funkcia l. druhu, K_m je modifikovaná Besselova funkcia 2. druhu, kde $W^2 = r^2(\beta^2 - n_{cl}^2\ k^2)$, $V^2 = U^2 + W^2$.

Ako vyplýva z rovnice (2), hodnota integrálu popisujúceho interferenčný člen dominantne závisí od rozdielu fázových konštánt jednotlivých modov $\Delta\beta_{pq}(\lambda) = \beta_p(\lambda) - \beta_q(\lambda)$, kde p, q sú rády modov (LP_{01} je mod 1. rádu, LP_{11} 2. rádu atď.) a dĺžky vlákna, na ktorej dochádza k interferencii. Rozdiel fázových konštánt $\Delta\beta_{12}(\lambda)$ pre mody LP_{01}, LP_{11} a $\Delta\beta_{14}(\lambda)$ pre mody LP_{01}, LP_{02}, vyplývajúce z rovníc (4) a (5) pre rozdielne parametre vlákien, je znázornený na obr. 2. Z tohto obrázku vidieť, že rozdiel fázových konštánt nadobúda extrém pre vlnové dĺžky λ_{pq}. Ako vyplýva z rovnice (4), hodnota λ_{pq} pre step-indexové vlákna závisí len od parametrov r, n_{co}, n_{cl} príslušného vlákna.

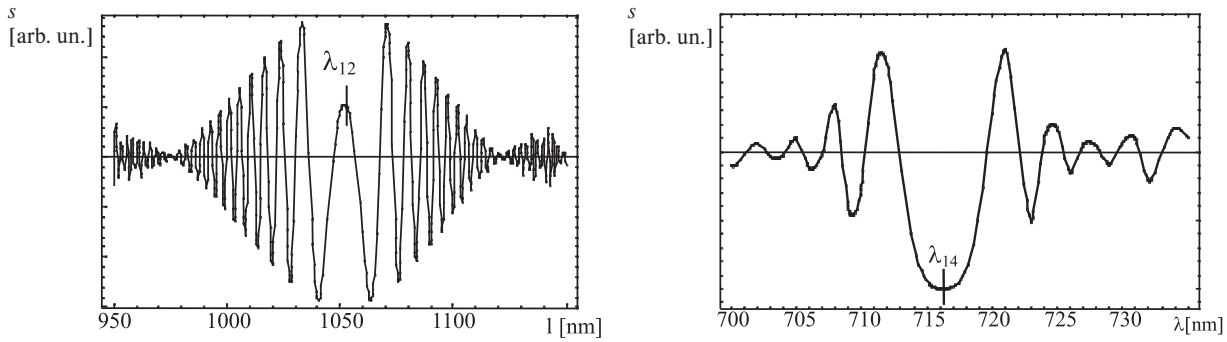
$$U\frac{J_{m+1}(U)}{J_m(U)} = W\frac{K_{m+1}(W)}{K_m(W)} \qquad (5)$$

where $m = 0,1,2,...$, J_m is Bessel function of the first kind, K_m is modified Bessel function of the second kind, $W^2 = r^2(\beta^2 - n_{cl}^2\ k^2)$, $V^2 = U^2 + W^2$.

According to (2), the value of the interference term depends dominantly on the difference of the phase constants of particular modes $\Delta\beta_{pq}(\lambda) = \beta_p(\lambda) - \beta_q(\lambda)$, (p, q are orders of the modes: 1 for LP_{01}, 2 for LP_{11}, and so on) and on the length of the fibre. The phase constant differences $\Delta\beta_{12}(\lambda)$ and $\Delta\beta_{14}(\lambda)$ following from a numerical solution of (4) and (5) for LP_{01}, LP_{11} modes and LP_{01}, LP_{02} modes respectively for different fibre parameters can be seen in Fig. 2. From this figure it is clear that there is an extreme value for wavelength λ_{pq}. It can be seen from (4) that this value depends only on the parameters of the fibre (r, n_{co}, n_{cl}).



Obr. 2 Závislosť rozdielu fázových konštánt modov a) LP_{0I} a LP_{1I}; b) LP_{0I} a LP_{02} od vlnovej dĺžky
Fig. 2 Wavelength dependence of the phase constant difference for a) LP_{0I} a LP_{1I}; b) LP_{0I} a LP_{02} modes



Obr. 3 Závislosť interferenčného člena od vlnovej dĺžky pri interferencii modov a) LP_{0I} a LP_{1I}; b) LP_{0I} a LP_{02}
Fig. 3 Wavelength dependence of the interference term when interfering a) LP_{0I} a LP_{1I}; b) LP_{01} a LP_{02}

Existencia extrému závislosti $\Delta\beta_{pq}(\lambda)$ sa prejaví i extrémom závislosti rozdielu fázy od vlnovej dĺžky, takže λ_{pq} sa dá chápať i ako interferenčný stred (obr. 3).

Pre step-indexové vlákno v priblížení slabovodivého vlákna je možné rozdiel fázových konštánt vyjadriť ako súčin dvoch funkcií. Prvá z týchto funkcií závisí len od parametrov vlákna r, n_{co}, n_{cl}

The existence of an extreme of $\Delta\beta_{pq}(\lambda)$ causes the existence of the extremal value of the phase difference as a function of wavelength so λ_{pq} can be taken as a centre of the interference (Fig. 3.).

In the weakly-guided approximation the phase constant difference can be expressed as a product of two functions. The

a druhá funkcia závisí iba od normovanej frekvencie V. Tak dostaneme:

first one depends only on the fibre parameters (r, n_{co}, n_{cl}) the second one only on the normalised frequency (V):

$$\Delta\beta_{pq} = \left[\frac{n_{co} - n_{cl}}{2r^2 n_{co}}\right]^{\frac{1}{2}} \frac{U_q^2(V) - U_p^2(V)}{V} = K(n_{co}, n_{cl}, r)\, f_{pq}(V) \qquad (6)$$

kde p, q sú rády modov.

Na obr. 3. je znázornená závislosť $f_{pq}(V)$ pre mody LP_{01}, LP_{11} a LP_{01}, LP_{02}. Ako vidieť z tohto obrázku, prvá funkcia nadobúda extrém pre hodnotu $V = 3{,}029$ [5] a druhá pre hodnotu $V = 4{,}448$. Ak uvážime, že tieto extrémy zodpovedajú interferenčným stredom príslúchajúcich danej dvojici modov, pomocou normovanej frekvencie pre hodnoty λ_{pq} dostávame:

$$\lambda_{12} = \frac{2\pi r}{3{,}029}\,(n_{co}^2 - n_{cl}^2)^{\frac{1}{2}} \quad \text{pre } LP_{01} \text{ a } LP_{11} \qquad (7)$$

$$\lambda_{14} = \frac{2\pi r}{4{,}448}\,(n_{co}^2 - n_{cl}^2)^{\frac{1}{2}} \quad \text{pre } LP_{01}, \text{ a } LP_{02} \qquad (8)$$

Porovnaním rovníc (7) a (8) dostaneme súvis medzi interferenčnými stredmi modov LP_{01}, LP_{11} a LP_{01}, LP_{02}:

$$\frac{\lambda_{12}}{\lambda_{14}} = 1{,}458 \qquad (9)$$

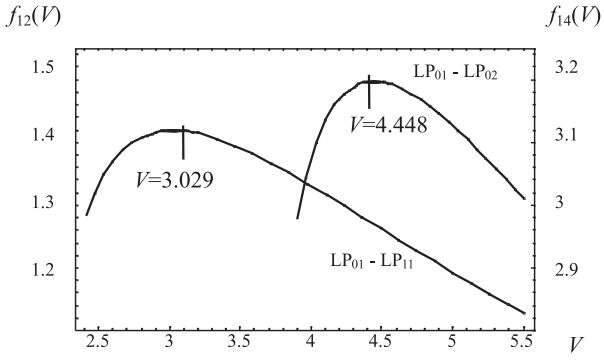
Ako sme už uviedli v prvom odseku, aby sme mohli pozorovať interferenciu modov, je nutné porušiť ortogonalitu jednotlivých modov. Dá sa to urobiť viacerými spôsobmi [1, 3, 6]. Pri našich meraniach optické pole vyšetrovaného vlákna snímame ďalším optickým vláknom (detekčným), ktoré je pri vyšetrovaní interferencie modov LP_{01} a LP_{11} osovo posunuté tak, ako ukazuje obr. 1a. Pri vyšetrovaní interferencie modov LP_{01} a LP_{02} je uložené súosovo, ale vzdialené tak, ako je zakreslené na obr. 1b.

Výsledky experimentu

Interferenciu modov optických vlákien sme experimentálne vyšetrovali v usporiadaní uvedenom v práci [3].

Výsledky získané pri vyšetrovaní interferencie prvých dvoch modov sú uvedené v prácach [2, 3], v ktorých bolo na základe nameraných priebehov poukázané na možnosť ich použitia pre určovanie homogenity optických vlákien.

Výsledky získané pri vyšetrovaní interferencie vyšších modov prezentujeme v tejto práci. Ide o výsledky získané pri vyšetrovaní interferencie dvojice modov LP_{01}a LP_{02} na telekomunikačných step-indexových vláknach firmy Pirelli a ich porovnanie s interferenciou modov LP_{01} a LP_{11}. Merania sme uskutočni-



Obr. 4 K určeniu extrému $\Delta\beta$ pre dvojicu modov LP_{01} - LP_{11} a dvojicu LP_{01} - LP_{02}
Fig. 4 On determination of extreme value of $\Delta\beta$ for modes LP_{01} - LP_{11} and LP_{01} - LP_{02} respectively

where p, q are the orders of modes.

In Fig. 3. the function $f_{pq}(V)$ for LP_{01}, LP_{11} and LP_{01} ,LP_{02} modes are presented. From this picture it can be seen, that the extremal values for the first and second function are $V = 3{,}029$ [5] and $V = 4{,}448$ respectively. If we assume the correspondence of these extremal values with the interference centre of the corresponding modes, we obtain:

$$\lambda_{12} = \frac{2\pi r}{3.029}\,(n_{co}^2 - n_{cl}^2)^{\frac{1}{2}} \quad \text{for } LP_{01} \text{ and } LP_{11} \qquad (7)$$

$$\lambda_{14} = \frac{2\pi r}{4.448}\,(n_{co}^2 - n_{cl}^2)^{\frac{1}{2}} \quad \text{for } LP_{01} \text{ and } LP_{02} \qquad (8)$$

Comparing the equations (7) and (8) we obtain the relation of the interference centres LP_{01}, LP_{11} and LP_{01}, LP_{02}:

$$\frac{\lambda_{12}}{\lambda_{14}} = 1.458 \qquad (9)$$

As it has been mentioned it is necessary to disturb the orthogonality of the modes so that we observe the modes interference. It can be done in different ways [1, 3, 6]. In our measurement other fibre (detecting fibre) scans the optic field of the fibre under study. When studying the LP_{01} and LP_{11} interference, the detecting fibre is located non-axisymmetrically (Fig. 1a) and when studying the interference of LP_{01} and LP_{02} modes, the detecting fibre is located as Fig.1b shows.

Results of the experiment

The interference of the fibres was investigated at the experimental set up described in [3].

The results obtained when the first two modes were interfering are published in [2, 3]. It was noted there that the intermodal interference could be used for investigation of homogenity of optic fibres. The information obtained during investigation of the interference of higher modes is illustrated by the results measured when investigating the interference of modes LP_{01} and LP_{02} on telecommunication step-index fibres of the firm Pirelli are presented in this paper together with their comparison with interference of LP_{01}, LP_{11}. The measures were

li na ôsmich vláknach vybraných z jedného telekomunikačného kábla. Z každého vlákna sme vybrali dve vzorky dĺžky ≈ 30 cm, a to zo začiatku a konca kábla (ich vzdialenosť v kábli bola 6109 m), takže výsledky vypovedajú o pozdĺžnej homogenite vyšetrovaných vlákien. Namerané hodnoty stredov interferencie, λ_{12}, λ_{14} pre jednotlivé vlákna sú uvedené v tab. 1., kde sú uvedené i hodnoty $\lambda^T_{12} = 1{,}468\ \lambda_{14}$, vypočítané z nameraných hodnôt λ_{14}, ktoré sa podľa vzťahu (9) majú rovnať hodnotám λ_{12}, takže v tabuľke uvedený rozdiel $\lambda_{12} - \lambda^T_{12}$ informuje o zhode resp. nezhode nameraných hodnôt s hodnotami vyplývajúcimi z teoretického popisu vlákien. Nepresnosť nameraných hodnôt je po riadku 1 nm, takže i hodnoty vypočítaných vlnových dĺžok uvedených v tab. 1. (pri ich vyjadrení v nanometroch) sú zaokrúhlené na celé čísla.

carried out on eight fibre samples of length ≈ 30 cm picked up from the beginning and the end of one telecommunication cable (the distance of the samples in the cable was 6109 m) so the results give us the information about the longitudinal homogenity of the investigated fibres. The measured values of wavelengths λ_{12}, λ_{14} (centres of interference) for particular fibres are given in Tab. 1. The values $\lambda^T_{12} = 1{,}468\ \lambda_{14}$, calculated from the experimentally measured values λ_{14}, (which, according (9), should be equal to values (12) are also presented in Tab. 1. Thus the difference $\lambda_{12} - \lambda^T_{12}$ given inTab.1. informs about the agreement between the measured values and the values coming from the theoretical model of fibres. Inaccuracy of the measured values is of the order of 1 nm, so even the calculated values expressed in nm are rounded to integers.

Tab. 1.

		Začiatok vlákna Beginning of the fiber				Koniec vlákna End of the fiber			
vlákno fiber		λ_{14} [nm]	λ_{12} [nm]	λ^T_{12} [nm]	$\lambda_{12} - \lambda^T_{12}$ [nm]	λ_{14} [nm]	λ_{12} [nm]	λ^T_{12} [nm]	$\lambda_{12} - \lambda^T_{12}$ [nm]
hnedé	brown	670	1013	984	+29	669	1009	982	+27
sivé	gray	666	1048	978	+70	669	1051	982	+69
oranžové	orange	666	1004	978	+26	668	1009	981	+28
biele	white	677	1051	994	+57	676	1051	992	+59
čierne	black	658	987	966	+21	655	982	962	+20
červené	red	672	1055	986	+69	668	1053	981	+72
modré	blue	691	1075	1014	+61	696	1087	1022	+65
zelené	green	697	997	1023	-26	700	1003	1028	-25

Ako vidieť z tabuľky, rozdiel medzi λ_{12} a λ_{14} zo začiatku a konca vlákna je rádovo niekoľko nanometrov. Maximálny rozdiel bol nameraný pre modré vlákno, kde rozdiel medzi interferenčnými stredmi prvého a štvrtého modu nameranými na vzorkách vybraných zo začiatku a z konca tohto vlákna bol 5 nm a v prípade prvého a druhého modu tento rozdiel bol až 12 nm. Svedčí to o relatívne veľkej pozdĺžnej nehomogenite tohto vlákna.

It can be seen from the table that the difference between the values λ_{12} and λ_{14} respectively measured at the beginning and the end of the fibre is several nanometers. The blue fibre shows the maximal difference: the difference between the interference centres of LP_{01} and LP_{02} modes at the beginning and the end of the fibre is 5 nm and for LP_{01} and LP_{11} modes it is 12 nm which indicates a relatively high longitudinal inhomogenity of this fibre.

Záver

Z matematického popisu šírenia sa svetla v step-indexovom optickom vlákne bol odvodený súvis stredov interferencie základného modu a vyšších modov.

Rozdielne hodnoty interferenčných stredov (medzi jednotlivými vláknami) sú podľa teórie popisujúcej step-indexové vlákna spôsobené rozdielnymi hodnotami r, n_{co}, n_{cl} vlákien. Polohy stredov interferencie ale môžu byť ovplyvnené i tým, že rôzne vlákna majú odlišné profily indexu lomu. V dôsledku toho, pri vláknach s rovnakými r, n_{co}, n_{cl}, rozdiel vlnovej dĺžky prislúchajúcej stredu interferencie prvých dvoch modov (λ_{12}) a hodnoty λ^T_{12} vyplývajúcej z λ_{14} (t. j. hodnoty $\lambda_{12} - \lambda^T_{12}$) môže poskytovať informáciu o tom, do akej miery sa profil indexu lomu príslušného vlákna odlišuje od ideálneho step-indexového profilu.

Conclusion

The relation of the interference centres of the primary and the higher modes was derived from the mathematical description of the light propagation in the step-index fibres.

When assuming the theory of step-index fibres, different values of the interference centres for particular fibres are caused by different values of r, n_{co}, n_{cl} of the fibres. But the interference centres could be caused also by different refractive index profiles for particular fibres. For the fibers with the same values r, n_{co} and n_{cl} it means that the difference of interference center of the first two modes λ_{12} and value λ^T_{12} coming from the interference center of the first and the fourth mode (i.e. the value $\lambda_{12} - \lambda^T_{12}$ in Tab. 1.) gives information about how the refractive index profile of particular fibre differs from a step-index profile.

Experimentálne vyšetrenie interferencie modov niekoľkých optických vlákien ukázalo, že jednotlivé vlákna iba približne spĺňajú z teórie step-indexového vlákna vyplývajúci súvis λ_{12} a λ^{T}_{12}. Rozdiely medzi nameranými hodnotami stredov interferencie prvých dvoch modov λ_{12} a hodnotou λ^{T}_{12} vyplývajúcou zo stredu interferencie prvého a štvrtého modu sa líšia o hodnotu, ktorá sa pohybuje v rozpätí od - 26 až do +72 nm, takže nemohla byť dôsledkom chýb merania.

Pozorované rozdiely λ_{12} a λ^{T}_{12} teda svedčia o tom, že meranie interferencie modov bude možné využiť na kontrolu toho, nakoľko je profil indexu lomu vyšetrovaného vlákna zhodný, alebo odlišný od priebehu so skokovou zmenou indexu lomu.

Poďakovanie

Týmto by sme chceli poďakovať Slovenským telekomunikáciám za poskytnutie vzoriek vlákien. Zároveň potvrdzujeme, že táto práca bola z väčšej časti vykonaná v rámci programu COST 265.

Recenzenti: J. Turán, J. Štelina

The experimental investigation of mode interference carried out on a series of optic fibres shows that particular fibres exhibit the accordance with the step-index theory of optic fibres only partially. The difference between the measured values λ_{12} and values λ^{T}_{12} range from -26nm up to +72nm, so it cannot be a measurement error. So the observed differences indicate that the telecommunication fibres exhibit a significant difference from the ideal step-index fibre.

The observed disagreement of experimental and theoretical values indicates that the measurement of interference could be used to test the agreement of the refractive index profile of investigated fibre with the refractive index profile of ideal step-index fibre.

Acknowledgement

We would like to express our thanks to the Slovak Telecommunications which have provided us with the fibre samples. We also confirm that this work has been performed in the framework of the COST 265 programme.

Reviewed by: J. Turán, J. Štelina

Literatúra - References

[1] HLUBINA, P.: The mutual interference of modes of a few mode fibre waveguide analysed in the frequency domain, Journal of Modern Optics, Vol. 42, 1995, pp. 2385-2399
[2] TUREK, I., MARTINČEK, I., DADO, M., ČERNICKÝ, S.: Using of intermodal interference as a detector of optical fibre homogenity, (in Slovak, resume in English) Fine Mechanics and Optics, 1999, pp. 223-226
[3] TUREK, I., MARTINČEK, I., Stránsky, R.: Interference of Modes in Optical Fibres, Optical Engineering, Vol. 39, 2000, pp. 1304-1309
[4] SNYDER, A., LOVE, J.: Optipacl waveguide theory, London, Chapman and Hall, 1983
[5] BLAKE, J. N., KIM, B. Y., SHAW, H. J.: Fiber-optic modal coupler using periodic microbending, Optic Letters, Vol. 11, 1986, pp. 177-179
[6] EFTIMOV, T. A., BOCK, W. J.: Sensing with a LP_{01}-LP_{02} intermodal interferometer, Journal of Lightwave Technology, Vol. 11, 1993, pp. 2150-2156

https://doi.org/10.26552/com.C.2000.2.10-15

komunikácie
COMMUNICATIONS

Martin Klimo *

RIADENIE VÝSTUPNEJ PAMÄTE PRE CBR PREVÁDZKU

PLAYOUT BUFFER CONTROL FOR CBR TRAFFIC

Komunikačné siete založené na princípe prepojovania paketov prinášajú pri prenose nový druh skreslenia spojitého signálu, ktoré vzniká náhodným oneskorením paketov. Na elimináciu tohto skreslenia sa používa výstupná pamäť. Tento článok ukazuje spôsob, ako určiť zvyškové skreslenie vyjadrené odstupom signálu od šumu. Predpokladá sa prevádzka typu CBR a statické riadenie výstupnej pamäte.

1. Úvod

Ak je signál prenášaný sieťou s prepojovaním kanálov, hlavná časť skreslenia je spôsobená aditívnym šumom. Vplyv aditívneho šumu na signál bol študovaný už dlhú dobu nielen teoreticky, ale aj prakticky. Pre hodnotenie kvality služby prenosu sa všeobecne používa odstup signálu od šumu. V paketových sieťach, ktoré používajú štatistický multiplex, nevyhnutne vzniká počas prenosu náhodné oneskorenie. Skúsenosti z počítačových sietí ukazujú, že pre prenos dát je dostatočnou charakteristikou stredné oneskorenie. To však nie je dostatočné v prípade prenosu hlasu, kedy chvenie buniek alebo paketov prináša nový druh skreslenia, ktoré doposiaľ nebolo teoreticky študované. Väčšina literatúry hľadá optimálnu stratégiu prevádzky vyrovnávacích pamätí v uzloch. V princípe, zmeny oneskorenia nie je možné úplne eliminovať. Nejaké chvenie celkového oneskorenia vždy ostáva a posledná možnosť, kde je možné zmenšiť ho, je výstupná vyrovnávacia pamäť a jej riadenie. Hlavný rozdiel medzi touto poslednou vyrovnávacou pamäťou a medziľahlými pamäťami v uzloch spočíva v tom, že jedine výstupná vyrovnávacia pamäť spracúva koncovú prevádzku. Táto má však podstatne chudobnejšiu paletu riadiacich nástrojov. Nemá k dispozícii možnosť použitia priorít, zmeny poradia alebo smerovania. Jedinými možnosťami je vkladanie a rušenie buniek, prípadne zmena intervalu medzi vzorkami. Tento typ vyrovnávacej pamäte sa volá playout buffer a jeho riadenie zmenou intervalu medzi vysielanými vzorkami je predmetom štúdia v tomto článku.

2. Chvenie oneskorenia

Uvažujme jednoduchý prenos signálu, v ktorom signál $s(t)$ je pravidelne vzorkovaný (CBR) a vzorky $s(n\Delta)$ sú prenášané (neuvažujeme kvantizačné skreslenie) sieťou ATM navzájom nezávisle. Blízko k uvedeným predpokladom je napr. prenos reči, v ktorom

Networks based on the packet switching principle introduce a new type of the continuous signal distortion which is caused by a random delay of packets. To eliminate this distortion, playout buffers are introduced. The paper presents a method how to evaluate the rest of this distortion in terms of Signal-to-Noise Ratio, when static policy is used for sample rate control in the case of CBR traffic.

1. Introduction

When the voice signal is transmitted over a circuit switched communication network, the main part of distortion is caused by noise. The impact of noise to the signal has been studied theoretically and in practice for a long time and the Signal-to-Noise Ratio (SNR) is broadly used as the Quality of Service (QOS) parameter. In the packet switched networks, where statistical multiplexing is used, random delay is introduced during the transmission. Experience in computer networks showed that average delay is sufficient QOS parameter when data are transmitted. But it is not sufficient, when voice is transmitted, and cell delay variation introduces a new type of distortion, which has not been theoretically studied yet. Anyway, a lot of strategies reducing delay variation are studied in the literature. Most of them are looking for optimal queueing strategy in node buffers. In principle, there is no possibility to eliminate delay variation totally. Some end-to-end delay variation remains and the last possibility how to reduce it consists of the last output buffer and its control. The main difference between this last buffer and intermediate buffers in the nodes is that the last buffer handles only one end-to-end traffic. It implicates only a limited palette of control tools. There are no more available tools like a priority system, sequencers, routers and the only control tools are cell insertion/discarding and intersample interval control. This type of buffer is called “playout buffer“, and its control by sample interval tuning is studied in this paper.

2. Sample delay variation

We assume a simple signal transmission when signal $s(t)$ is regularly sampled (CBR), and samples $s(n\Delta)$ are transmitted independently (quantizing error is omitted) over ATM network. Near these assumptions is for example PCM voice transmission,

* **Martin Klimo**
University of Žilina, Department of Information networks, SK-01026 Žilina, Slovak Republic
Tel.: +421-89-762 329, Fax +421-89-655 530, E-mail martin@frkis.utc.sk

PCM kanál 2 Mbits/s (30 hlasových kanálov) je mapovaný do 47 bytov ATM bunky. Sieť vnáša do prenosu náhodné oneskorenie a vzorky nie sú prijaté v správnom čase. Rozdiel medzi ekvidištantnými referenčnými okamžikmi a okamžikmi, kedy sú vzorky posielané poslucháčovi, považujeme za oneskorenie vzoriek.

Predpoklad 1
Nech je ergodický náhodný proces stacionárny v širšom zmysle s
- ohraničenou strednou hodnotou $|e_S| = |E[s(t)]| < +\infty$,
- kovariančnou funkciou $R_S(t) = E[s(0)\, s(t)]$, $t \in \Re$,
- ohraničenou strednou energiou $\sigma_s^2 = R_S(0) = E[s^2(t)] < +\infty$,
- ohraničeným výkonovým spektrom $\hat{s}(\omega) = 0$, $|\omega| \geq \Omega > 0$,

kde $\hat{s}(\omega) \int_{-\infty}^{\infty} R_S(t) e^{-j\omega t}\, dt$, $\omega \in \Re$ je spektrálna výkonová hustota. Označenie $\Re = (-\infty, \infty)$ $Z = \{..., -1, 0, 1, ...\}$ znamená množinu reálnych resp. celých čísel.

Vzorkovacia teoréma [2] pre centrovaný náhodný proces:

if 2 Mbps sample stream (30 voice channels) is mapped into 47 bytes payload of ATM cells. The network introduces a random delay, and samples are received at improper time instants. The difference between equidistant reference instants, and instants when the samples are playing out are assumed as sample delays.

Assumption 1
Let the second order be stationary ergodic stochastic process with
- limited average $|e_S| = |E[s(t)]| < +\infty$,
- covariation function $R_S(t) = E[s(0)\, s(t)]$, $t \in \Re$,
- limited energy $\sigma_s^2 = R_S(0) = E[s^2(t)] < +\infty$,
- limited bandwidth $\hat{s}(\omega) = 0$, $|\omega| \geq \Omega > 0$,

where $\hat{s}(\omega) \int_{-\infty}^{\infty} R_S(t) e^{-j\omega t}\, dt$, $\omega \in \Re$ is the power density function.

The notation $\Re = (-\infty, \infty)$ $Z = \{..., -1, 0, 1, ...\}$ and is used in the paper.

The sampling theorem [2] for a zero-mean stochastic process:

$$\dot{s}(t) = \underset{N\to\infty}{l.i.m} \sum_{-N}^{N} \dot{s}(k\Delta)\Phi(t - k\Delta),\ \Delta = \frac{\pi}{\Omega},$$

$$\Phi(t) = \frac{sin\Omega t}{\Omega t} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{\Phi}(\omega) e^{j\omega t}\, d\omega = \frac{1}{2\Omega} \int_{-\Omega}^{\Omega} e^{j\omega t}\, d\omega,\ t \in \Re$$

ukazuje, ako môže byť náhodný proces rekonštruovaný zo vzoriek, ktoré sú z náhodného procesu odoberané pravidelne so vzorkovacím intervalom Δ. V takomto prípade má chybový signál nulovú strednú energiu. Ak sú však vzorky prenášané sieťou s komutáciou paketov, ktorá vnáša do prenosu náhodné oneskorenie, vzorky sú používané pre rekonštrukciu signálu nepravidelne (prinajmenšom ak výstupná pamäť je prázdna v okamžiku, kedy sa mala použiť vzorka na rekonštrukciu). Predpokladajme, že táto nepravidelnosť, ktorú vyjadrujeme odchylkou (oneskorením) od pravidelných vzorkovacích okamžikov, vytvára bodový proces $\{\tau_k, k \in Z\}$.

Predpoklad 2
Nech je $\{\tau_k, k \in Z\}$ ergodický, stacionárny náhodný bodový proces s charakteristikami prvého rádu:
- distribučná funkcia $F(t) = p\ \{\tau_k \leq t\}$, $t \in \Re$, $k \in Z$,
- charakteristická funkcia $G(\omega) = E[e^{j\omega\tau_\kappa}] = \int_{-\infty}^{\infty} e^{j\omega t}\, dF(t)$, $\omega \in \Re$, $k \in Z$.

Vysielaný (rekonštruovaný) signál bude $r(t) = \sum_k s(k\Delta)\Phi$ $(t - k\Delta - \tau k)$, a rozdiel medzi obnoveným a originálnym signálom vytvorí šum

$$n(t) = r(t) - s(t) = \sum_k s(k\Delta)[\Phi(t - k\Delta - \tau k) - \Phi(t - k\Delta)].$$

Celkové hodnotenie kvality je založené na rozdiele pôvodného $s(t)$ a rekonštruovaného signálu $r(t)$. Ak je miera tohto hodnotenia lineárna, potom celková kvalita môže byť vyjadrená charakteristikami $n(t)$. Hlavné zjednodušenie, ktoré používame

shows how the random process can be reconstructed from the samples shifted over time regularly with a sampling interval Δ. In this case the error signal has zero energy. Due to a random delay in the packet switched network the samples are taken irregularly by the reconstruction procedure (at least, if the playout buffer is empty at the regular interval). Let us suppose that this irregularity, expressed by differences (delays) from regular sampling points, creates the point process $\{\tau_k, k \in Z\}$.

Assumption 2
Let $\{\tau_k, k \in Z\}$ be an ergodic, stationary, random point process with first order characteristics:
- distribution $F(t) = p\ \{\tau_k \leq t\}$, $t \in \Re$, $k \in Z$,
- characteristic function $G(\omega) = E[e^{j\omega\tau_\kappa}] = \int_{-\infty}^{\infty} e^{j\omega t}\, dF(t)$, $\omega \in \Re$, $k \in Z$.

The playout (reconstructed) signal will be $r(t) = \sum_k s(k\Delta)\Phi$ $(t - k\Delta - \tau k)$, and the difference between the playout signal and the original one creates the so-called noise

$$n(t) = r(t) - s(t) = \sum_k s(k\Delta)[\Phi(t - k\Delta - \tau k) - \Phi(t - k\Delta)].$$

End-to-end quality measures are based on the difference between the original signal $s(t)$ and the playout signal $r(t)$. If this measure is linear, the end-to-end quality may be expressed in terms of the noise $n(t)$.

The main simplification used in this paper is based on the sample independence assumption, when the original signal is

v tomto článku spočíva v predpoklade nezávislosti vzoriek t. j. v predpoklade, že originálny signál tvorí „biely šum s ohraničeným spektrom“. V skutočnosti žiadna aplikácia takýto signál nevytvára, avšak toto zjednušenie môže dať priamo použiteľné výsledky z dvoch dôvodov: kompresia signálu zmenšuje koreláciu medzi vzorkami a signál blízky bielemu šumu sa často používa na účely testovania.

Nech $s(t)$, $t \in \Re$ a $\{\tau_k, k \in Z\}$ sú stochastické procesy podľa Predpokladu 1 a Predpokladu 2. Ak $s(k\Delta)$ a τ_k, $k \in Z$ sú dvojice nezávislých premenných a, $\{s(n\Delta), s(m\Delta) \mid n, m \in Z, n \neq m\}$ sú tiež nezávislé premenné, t. j.

$$R[(n-m)\Delta] = \begin{cases} \sigma_s^2, & n = m \\ 0, & n \neq m \end{cases},$$

potom [1] spektrálna výkonová hustota $\hat{n}(\omega) = \int_{-\infty}^{\infty} R_N(t) e^{-j\omega t}\, d\omega$,

$R_N(t) = E[n(0)n(t)]$, je

$$\hat{n}(\omega) = 2\sigma_S^2 \hat{\Phi}(\omega)[1 - Re\, G(\omega)],\ \omega \in \Re \qquad (1)$$

a stredný výkon šumu $\sigma_N^2 = E[n^2(0)] = R_N(0) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{n}(\omega) d\omega$ je

$$\sigma_N^2 = 2\sigma_S^2 \left(1 - \frac{1}{2\Omega}\int_{-\Omega}^{\Omega} Re\, G(\omega) d\omega\right). \qquad (2)$$

Ako vidíme, použitie „spektrálne ohraničeného bieleho šumu“ ako testovacieho signálu dáva veľmi jednoduchý vzťah na výpočet stredného výkonu šumu. To môže byť zaujímavé pre inžinierske výpočty, v ktorých sa odstup signálu od šumu všeobecne používa na hodnotenie kvality. Ak použijeme definíciu *SNR*

$$SNR = 10\log \frac{\sigma_S^2}{\sigma_N^2}, \quad [\text{dB}] \qquad (3)$$

vo vzťahu (2), potom dostávame

$$SNR = -10\log 2\left(1 - \frac{1}{2\Omega}\int_{-\Omega}^{\Omega} Re\, G(\omega) d\omega\right), [\text{dB}].$$

Nasledujúci príklad ukazuje vplyv náhodného oneskorenia vzoriek na signál typu „spektrálne ohraničený biely šum“, ak predpokladáme, že oneskorenie má exponenciálne rozdelenie. Takéto rozdelenie je aproximáciou rozdelenia oneskorenia paketov vo veľkých datagramových sieťach s veľkým počtom zdrojov, v ktorých výstupný tok zo siete je blízky Poissonovmu procesu. Nech $\{s(n\Delta), s(m\Delta) \mid n.m \in Z,\ n \neq m\}$ sú nezávislé veličiny, oneskorenie je náhodná veličina s exponenciálnym rozdelením s distribučnou funkciou $F(t) = 1 - e^{-\mu t}$, $t \geq 0$, $\mu \geq 0$ strednou hodnotou $\bar{\tau} = \frac{1}{\mu}$, a charakteristickou funkciou $G(\omega) = \frac{\mu}{\mu + j\omega}$, $\omega \in \Re$.

Potom je spektrálna výkonová hustota šumu

$$\hat{n}(\omega)_{exp} = 2\sigma_S^2 \hat{\Phi}(\omega) \frac{\omega^2}{\mu^2 + \omega^2},$$

stredný výkon šumu

“band-limited white noise“. Of course, no real application can produce such a signal. Anyhow, this simplification may be interesting for two reasons: signal compression before sampling decreases the covariation between samples, and it is also very common to have a defined test input signal and the white noise is often used for testing communication networks.

Let $s(t)$, $t \in \Re$ and $\{\tau_k, k \in Z\}$ be the stochastic processes according to Assumption 1 and Assumption 2 respectively. If $s(k\Delta)$ and τ_k, $k \in Z$ are pairs of independent variables, $\{s(n\Delta), s(m\Delta) \mid n, m \in Z, n \neq m\}$ are independent variables, i.e.

$$R[(n-m)\Delta] = \begin{cases} \sigma_s^2, & n = m \\ 0, & n \neq m \end{cases},$$

then [1] the power spectral density $\hat{n}(\omega) = \int_{-\infty}^{\infty} R_N(t) e^{-j\omega t}\, d\omega$,

$R_N(t) = E[n(0)n(t)]$, is

$$\hat{n}(\omega) = 2\sigma_s^2 \hat{\Phi}(\omega)[1 - Re\, G(\omega)],\ \omega \in \Re \qquad (1)$$

and the average noise power $\sigma_N^2 = E[n^2(0)] = R_N(0) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{n}(\omega) d\omega$ is

$$\sigma_N^2 = 2\sigma_s^2 \left(1 - \frac{1}{2\Omega}\int_{-\Omega}^{\Omega} ReG(\omega) d\omega\right). \qquad (2)$$

As we can see, the band-limited white noise as a test signal leads to a very simple formula for the noise power calculations. This makes it interesting for engineering calculations, where signal-to-noise ratio (SNR) is broadly used as a signal quality measure. By defining *SNR*

$$SNR = 10\log \frac{\sigma_S^2}{\sigma_N^2}, \quad [\text{dB}] \qquad (3)$$

formula (2) becomes to

$$SNR = -10\log 2\left(1 - \frac{1}{2\Omega}\int_{-\Omega}^{\Omega} Re\, G(\omega) d\omega\right), [\text{dB}].$$

The following example shows the impact of the sample delay to the band-limited white noise-like original signal, when the delay has an exponential distribution. This distribution is an approximation of the packet delay in a large datagram network loaded by many sources, where the output stream is near to the Poisson process. Let $\{s(n\Delta), s(m\Delta) \mid n.m \in Z,\ n \neq m\}$ be independent variables, delay is exponentially distributed with distribution $F(t) = 1 - e^{-\mu t}$, $t \geq 0$, $\mu \geq 0$ average value $\bar{\tau} = \frac{1}{\mu}$, and characteristic function $G(\omega) = \frac{\mu}{\mu + j\omega}$, $\omega \in \Re$.

Then the noise power spectral density is

$$\hat{n}(\omega)_{exp} = 2\sigma_S^2 \hat{\Phi}(\omega) \frac{\omega^2}{\mu^2 + \omega^2},$$

the average noise power

$$\sigma^2_{N_{exp}} = 2\sigma^2_S\left(1 - \frac{\Delta}{\pi\bar{\tau}}\ \text{arctg}\ \frac{\pi\bar{\tau}}{\Delta}\right) \qquad (4)$$

a odstup signálu od šumu

$$SNR_{exp} = 10\log 2\left(1 - \frac{\Delta}{\pi\bar{\tau}}\ \text{arctg}\ \frac{\pi\bar{\tau}}{\Delta}\right), \text{[dB]}.$$

3. Výstupná pamäť

Výstupná pamäť sa používa na elimináciu chvenia oneskorenia vzoriek. V závislosti od akceptovateľného celkového oneskorenia čakajú vzorky vo výstupnej pamäti a potom sú vysielané von v pravidelných časových intervaloch. Ak v okamžiku, kedy mala byť vysielaná vzorka, je výstupná pamäť prázdna, vznikne výpadok signálu, až pokiaľ nie je k dispozícii oneskorená vzorka. Toto zvyškové oneskorenie opatovne spôsobuje šum a na jeho štúdium môžeme použiť predchádzajúce výsledky. Nech p_0 je pravdepodobnosť, že výstupná pamäť je prázdna v čase, keď má byť vzorka hraná von. $F_{(t)_{res}}$ a distribučná funkcia zvyškového oneskorenia t. j. intervalu od okamžiku, kedy vzorka mala byť vysielaná do okamžiku, kedy je skutočne vysielaná. Potom oneskorenie nadobúda nulovú hodnotu s pravdepodobnosťou (1_{-p0}), kladnú hodnotu zvyškového oneskorenia τ_{res} s pravdepodobnosťou p_0, a spektrálna výkonová hustota šumu a odstup signálu od šumu sú:

$$\hat{n}(\omega) = p_0\hat{n}(\omega)_{res},$$
$$SNR = SNR_{res} - 10\log(p_0).$$

Index „res“ označuje funkciu počítanú pre prípad, že zvyškové chvenie vzniká s pravdepodobnosťou 1.

4. Riadenie výstupnej pamäte

Predchádzajúce vzťahy ukazujú, ako sa zlepší odstup signálu od šumu v prípade, že na oneskorené vzorky čakáme, miesto toho, aby sme chýbajúce (oneskorené) vzorky síce vysielali včas, ale nahradené nulovou hodnotou. Existuje niekoľko spôsobov, ako tento výsledok ešte zlepšiť. Po prvé, môžeme použiť iné spôsoby náhrady chýbajúcej vzorky miesto náhrady nulovou hodnotou.To má zmysel v prípade, že zvýšenie odstupu signálu od šumu bude väčšie, ako je príspevok SNR_{res}. Po druhé, môže byť zmenšená pravdepodobnosť vyprázdnenia pamäte p_0 a to tak, že vzorky budú vysielané von pomalšie ako udáva vzorkovací interval, ak sa pamäť blíži k vyprázdneniu. V tomto článku budeme uvažovať statické riadenie výstupnej rýchlosti vzoriek, t. j. budeme predpokladať, že bude použitý interval medzi vysielanými vzorkami $\Delta_n = \Delta + \delta_n$, $n = 1, 2, ...$, ak v čase po odvysielaní predchádzajúcej vzorky je v pamäti práve n vzoriek (predpokladáme neohraničenú veľkosť vyrovnávacej pamäte). Ak je v okamžiku, kedy mala byť vysielaná vzorka pamäť prázdna, vzorka bude odvysielaná von akonáhle príde, t. j. interval medzi vzorkami bude $\Delta_1 + \tau_{res}$. Marginálna charakteristická funkcia oneskorenia bude

$$G(\omega) = E[e^{j\omega t}] = \sum_{n=1}^{\infty} p_n e^{j\omega\delta_n} + p_0\, E[e^{j\omega\tau_{res}}],\ \omega \in \Re$$

$$\sigma^2_{N_{exp}} = 2\sigma^2_S\left(1 - \frac{\Delta}{\pi\bar{\tau}}\ \text{arctg}\ \frac{\pi\bar{\tau}}{\Delta}\right) \qquad (4)$$

and then the signal-to-noise ratio is

$$SNR_{exp} = 10\log 2\left(1 - \frac{\Delta}{\pi\bar{\tau}}\ \text{arctg}\ \frac{\pi\bar{\tau}}{\Delta}\right), \text{[dB]}.$$

3. Playout buffers

To eliminate sample delay variation, playout buffers are used. Depending on an acceptable average delay, samples are waiting in the output queue, and they are “played out“ with regular intervals. If the buffer is empty when a sample should be sent, a signal gap occurs until the delayed sample is available. This residual delay again produces noise, and the previous results may be applied. Let p_0 be the probability that a buffer is empty when the sample should be played out, and $F_{(t)_{res}}$ the distribution of the residual delay i.e. the interval since the sample should be played out. Then the delay τ takes a zero value with probability (1_{-p0}), value of the residual delay τ_{res} with probability p_0, and

$$\hat{n}(\omega) = p_0\hat{n}(\omega)_{res},$$

$$SNR = SNR_{res} - 10\log(p_0).$$

Here “res“ indexes functions, where only residual sample jitter is taken into account.

4. Playout buffer control

The previous formula shows how waiting for the delayed sample improves SNR, compared to the policy when the empty buffer generates zero samples, but just in time. There are several methods how to improve this result. Firstly, other lost sample replacing methods bring an advantage, but of course, only if their gain in SNR, compared to the zero stuffing is greater than SNR_{res}. Secondly, probability of an empty buffer p_0 can be decreased if samples are played out slower than the sampling rate, when the buffer is coming to be empty. In this paper we assume static control policy of the playout sample rate, i.e. we assume that $\Delta_n = \Delta + \delta_n$, $n = 1, 2, ...$ intersample interval is used, when the sample should be played out, and the buffer contains n samples (infinite buffer capacity is assumed). If the buffer is empty at the playing out instant, the sample will be played out as soon as it comes, i.e. intersample interval $\Delta_1 + \tau_{res}$ occurs. The marginal characteristic function of delay is

$$G(\omega) = E[e^{j\omega t}] = \sum_{n=1}^{\infty} p_n e^{j\omega\delta_n} + p_0\, E[e^{j\omega\tau_{res}}],\ \omega \in \Re$$

a spektrálna výkonová hustota šumu

$$\hat{n}(\omega) = 2\sigma_S^2 \hat{\Phi}(\omega)\left[1 - p_0 - \sum_{n=1}^{\infty} p_n \cos\omega\delta_n\right] + p_0\hat{n}(\omega)_{res},\ \omega \in \Re.$$

Stredný výkon šumu (2) je

$$\sigma_N^2 = \frac{2\sigma_S^2}{\Delta}\left[1 - p_0 - \sum_{n=1}^{\infty} pn\,\frac{\sin\pi\epsilon_n}{\pi\epsilon_n}\right] + p_0\sigma_{N_{res}}^2 \qquad (5)$$

kde $\epsilon_n = \frac{\delta_n}{\Delta}$, $n = 1, 2, ...$ je relatívne oneskorenie vzorky.

Vzťah (3) môžeme použiť pre výpočet odstupu signálu od šumu.

Ako príklad uvažujme neohraničenú FIFO vyrovnávaciu pamäť, do ktorej prichádzajú vzorky tak, že vytvárajú Poissonov proces s intenzitou $1/\Delta$ vzoriek za sekundu, teda so stredným intervalom medzi vzorkami $\bar{\tau} = \Delta$. Statické riadenie nech spočíva v tom, že je zadaná hranica $N \geq 1$ nasledujúco:

- $\delta_n = \delta$, $n = 1, ..., N$
- $\delta_n = -\delta$, $n = N + 1, N + 2, ...$

V tomto prípade aj zvyškové oneskorenie má exponenciálne rozdelenie so strednou hodnotou Δ. Použitím vzťahu (4) v (5) dostávame

and the noise power spectral density, is

The average noise power (2) is

$$\sigma_N^2 = \frac{2\sigma_S^2}{\Delta}\left[1 - p_0 - \sum_{n=1}^{\infty} pn\,\frac{\sin\pi\epsilon_n}{\pi\epsilon_n}\right] + p_0\sigma_{N_{res}}^2 \qquad (5)$$

where $\epsilon_n = \frac{\delta_n}{\Delta}$, $n = 1, 2, ...$ is a relative sample delay.

Formula (3) can be used for SNR calculation.

As an example we assume an infinite FIFO buffer with Poisson input sample stream at the rate of $1/\Delta$ samples per second, with the average sampling interval $\bar{\tau} = \Delta$. The static control strategy is given by threshold $N \geq 1$ as follows:

- $\delta_n = \delta$, $n = 1, ..., N$
- $\delta_n = -\delta$, $n = N + 1, N + 2, ...$

In this case, the rest delay has exponential distribution with an average rest delay Δ. Then applying (4) in (5) gives

$$\frac{\sigma_N^2}{\sigma_S^2} = \frac{2}{\Delta}\left(1 - \frac{sin\pi\epsilon}{\pi\epsilon}\right)(1 - p_0) + 2\left(1 - \frac{1}{\pi}\,\mathrm{arctg}\,\pi\right)p_0 \qquad (6)$$

kde $\epsilon = \frac{\delta}{\Delta}$. Formálne môžeme písať $p_0 = 1 - \rho = 1 - \sum_{n=1}^{\infty} p_n\rho_n$, kde $\rho_n = \frac{\Delta + \delta_n}{\Delta} = 1 + \epsilon_n$, rozdelenie pravdepodobnosti p_n, $n = 0, 1, ...$ však musíme aj tak vypočítať. Môžeme ho dostať ako invariantné rozdelenie pravdepodobnosti stavov Markovovho reťazca vnoreného do okamžikov po odvysielaní vzoriek. Môže byť počítané napríklad pomocou rekurzie

where $\epsilon = \frac{\delta}{\Delta}$. Formally, we can write $p_0 = 1 - \rho = 1 - \sum_{n=1}^{\infty} p_n\rho_n$, where $\rho_n = \frac{\Delta + \delta_n}{\Delta} = 1 + \epsilon_n$, anyhow, the distribution p_n, $n = 0, 1, ...$ should be calculated. It can be obtained as an invariant distribution of an embedded Markov chain to the instants when samples should be played out. It can be calculated, for example, by recursion

$$p_{n+1} = e^{\rho_{n+1}}\,p_n - \frac{\rho_1^n}{n!}\,e^{\rho_{n+1}-\rho^1}\,p_0 - \sum_{k=1}^{n}\frac{\rho_k^{n=k+1}}{(n-k+1)}\,e^{\rho_{n+1}\,-\rho_k}\,p_k,\ n = 0, 1, ...,$$

s použitím normujúcej rovnice $\sum_{n=0}^{\infty} p_n = 1$, pričom predpokladáme $0 \leq \rho < 1$, t. j. $0 \leq p_0 - \sum_{n=1}^{\infty} p_n\epsilon_n < 1$

Je prirodzené očakávať, že šum spôsobený chvením oneskorenia bude tým menší, čím je menšia pravdepodobnosť vyprázdnenia vyrovnávacej pamäte. Znižovanie pravdepodobnosti vyprázdnenia pamäte sa dá dosiahnuť oneskorením prvých vzoriek v slove, to však môže spôsobiť neakceptovateľné stredné oneskorenie pre aplikácie bežiace v reálnom čase. Preto výsledok (6) musí byť analyzovaný súčasne so stredným oneskorením, ktoré vypočítame z Chinčin-Pollaczekovej formuly

with the norm equation $\sum_{n=0}^{\infty} p_n = 1$. We also assume $0 \leq \rho < 1$, i.e. $0 \leq p_0 - \sum_{n=1}^{\infty} p_n\epsilon_n < 1$. It is quite natural that noise caused by delay variation will be less when the buffer will be empty with very low probability. On the other hand, it leads to a very high average delay, which may be unacceptable for application. Then the result (6) must be analysed together with the defined average delay, which may be obtained from Khinchine-Pollaczek formula

$$E\{T_w\} = \frac{\Delta}{2}\left(\frac{1}{p_0 - \sum_{n=1}^{\infty} p_n \epsilon_n} - 1\right) = konst.,$$

čo v našom prípade dáva

$$E\{T_w\} = \frac{\Delta}{2}\left(\frac{1}{p_0 - (p_+ - p_-)\epsilon} - 1\right) = konst.,$$

kde $p_+ = \sum_{n=1}^{N} p_n$ a $p_- = \sum_{n=N+1}^{\infty} p_n$.

5. Záver

Potreba prenosu spojitých signálov paketovými sieťami v reálnom čase vedie k riešeniu problému, ako zabezpečiť prenášaný signál voči zhoršeniu kvality spôsobenej náhodným oneskorením vzoriek. Posledná možnosť v informačnom reťazci ako to urobiť je riadenie výstupnej vyrovnávacej pamäte. Článok uvádza vzťahy, ktoré môžu byť použité pri ladení parametrov výstupnej vyrovnávacej pamäte, najmä intervalu medzi vysielaním vzoriek tak, aby sa maximalizoval odstup signálu od šumu, ktorý je použitý ako miera kvality prenosu signálu za podmienky, že vzorky v pôvodnom signále sú navzájom nezávislé.

Recenzenti: P. Podhradský, K. Blunár

$$E\{T_w\} = \frac{\Delta}{2}\left(\frac{1}{p_0 - \sum_{n=1}^{\infty} p_n \epsilon_n} - 1\right) = konst.,$$

or, in our particular case,

$$E\{T_w\} = \frac{\Delta}{2}\left(\frac{1}{p_0 - (p_+ - p_-)\epsilon} - 1\right) = konst.,$$

where $p_+ = \sum_{n=1}^{N} p_n$ and $p_- = \sum_{n=N+1}^{\infty} p_n$.

5. Conclusions

A need for real-time transmission of continuous signals over a packet-switched network leads to the problem how to save the quality of the transmitted signal against the degradation caused by the random delay of samples. The last possibility is to control the playout buffer. There are several formulas presented in this paper, which can be used for tuning parameters of static policy for intersample intervals used by the buffer to play out the samples. Signal-to-Noise Ratio is used as a Quality of Service parameter, and signal with independent samples is taken as an original signal.

Reviewed by: P. Podhradský, K. Blunár

Literatúra - References

[1] KLIMO, M., KORNER, U.: Sample Jitter in the Real-Time Applications, to be published.
[2] JERRI, A. J.: The Shannon Sampling Theorem - Its Various Extensions and Applications: A Tutorial Review, Proc. IEEE, vol. 65, no. 11, pp. 1565 - 1596, Nov. 1977,

https://doi.org/10.26552/com.C.2000.2.16-20

komunikácie
COMMUNICATIONS

Milan Karpf - Boris Šimák *

VPLYV ŠUMU TYPU SOĽ A KORENIE NA KÓDOVANIE OBRAZU ZALOŽENOM NA ŠTIEPENÍ OBRAZU

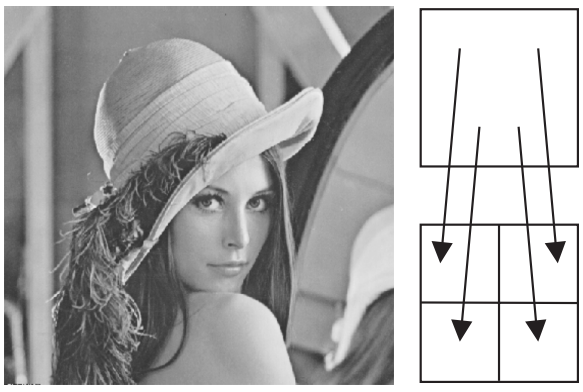
SALT & PEPPER NOISE IMPACT ON IMAGE CODING BASED ON IMAGE SPLITTING

Tento článok sa zaoberá vplyvom šumu a riadiacich parametrov (segmentačné kritériá) na zvolenú metódu segmentácie. Boli použité dva riadiace parametre. Dynamické kritérium a kritérium vyhodnocujúce varianciu oblasti. Ako segmentačná metóda bolo použité štiepenie obrazu. Vplyv šumu a riadiacich parametrov na segmentačnú metódu a kódovánie bol vyhodnotený pomocou objektívneho kritéria kvality obrazu (vrcholový odstup signál šum) a počtu oblastí segmentovaného obrazu.

The paper deals with the influence of noise and two control parameters on a segmentation method for image coding. Dynamic min_max criterion and variance criterion were used as segmentation control parameters. For the segmentation of images we have used the method of image splitting. The influence of the value of variance and dynamic criterion on the quality of the segmentation coded image is demonstrated. We have evaluated the objective criterion of the quality of the image (peak signal-to-noise ratio) and number of regions in the image for different values of segmentation control parameters and noise density.

Introduction

During the segmentation for the segmentation-based coding [3] we analyse a given image and we try to find areas homogenous in a certain sense (for example colour or grey level). The internal parts of certain areas (textures) and borders are coded separately. These segmentation methods involve the region growing, combinations of growing and splitting and a whole number of other methods adopted from the scientific discipline of computer vision and medical data processing.



(a) *(b)*

Figure 1 (a) Image Lena-original. (b) One splitting step of quadtree decomposition algorithm.

Quadtree decomposition

For the segmentation of image we have used the method of quadtree decomposition which is one of image splitting methods. The advantages of quadtree decomposition compared to other segmentation methods are speed and simple coding of area borders [2]. Quadtree decomposition divides a square image into four equal-sized square blocks (see Figure 1 (b)) and then tests whether each block fulfils the criterion of homogenity. If not, it is divided again into four equal-sized square blocks. This procedure is repeated again for every block in the image until the criterion for the tested block is met or until the size of the block equals one pixel or a set minimum value (our minimum set value is one pixel).

The first criterion of homogenity that we used (1) splits a block when the difference between the maximum value of the block pixels and the minimum value of the block pixels is greater than the set value (threshold) (2). The threshold is specified as a value between 0 and 1, where 1 corresponds to the maximal possible value (255) for the eight bits representation of a grey level image. Therefore, the threshold controls the segmentation method. The splitting procedure is defined as follows:

Let X_i be an area (block) of the image and $x(m,n)$ are points in this area $x(m,n) \in X$. Then, the criterion of homogenity is

$$min\_max(X_i) = \max_{X_i} (x(m,n)) - \min_{X_i} (x(m,n) \qquad (1)$$

* **Ing. Milan Karpf, Doc. Ing. Boris Šimák, CSc.**
Department of Telecommunication Engineering, Faculty of Electrical Engineering, Czech Technical University, Technická 2, CZ-16627 Praha 6, Czech Republic; E-mail: karpf@feld.cvut.cz, simak@feld.cvut.cz

and the splitting condition is

$$min\_max(X_i) > threshold_i \begin{cases} TRUE & split\ area\ X_i \\ FALSE & do\ not\ split\ area\ X_i \end{cases} \quad (2)$$

This criterion is simple and fast and recommended in [1], [4]. We code the value of particular pixels by the mean value across the area to which pixels were assigned by the segmentation algorithm. The influence of the value of threshold for *min_max* criterion (dynamic criterion [1]) on the quality of the segmentation coding is depicted in Figure 2 (a).

The second criterion of homogenity that we used is the variance of area. Variance is defined as follows.

Let $\bar{x}_i = \frac{1}{S_i} \sum_{\forall x \in X_i} x(m,n)$, where

$i = 1, 2, \ldots$ *number of areas in an image* (3)

and S_i = the number of pixels in area X_i. Then, the variance of area is

$$\sigma_i^2 = \sigma^2(X_i) = \frac{1}{S_i} \sum_{\forall x \in X_i} (x(m,n) - \bar{x})^2 \quad (4)$$

and the splitting condition is

$$\sigma_i^2(X_i) > variance\ threshold$$

$$unnormalized_i \begin{cases} TRUE & split\ area\ X_i \\ FALSE & do\ not\ split\ area\ X_i \end{cases} \quad (5)$$

The variance threshold is specified as a normalized value $\in (0, \infty)$, where 1 corresponds to the maximal possible value (255) for the eight bits representation of a grey level image. In other words:

$$variance\ threshold = \frac{variance\ threshold\ unnormalized}{255}. \quad (6)$$

The influence of the value of *variance threshold* on the quality of the segmentation coding is depicted in Figure 2 (b).

When we code areas of image using the mean values of areas, we calculate the value of *PSNR*. Given that *MxN* is the size of the image and $y(m,n)$ is the value of pixel of that processed image.

$$e(m,n) = y(m,n) - x(m,n)$$
where $m = 1, 2, \ldots, M$ and $n = 1, 2, \ldots N$. (7)

$$Mean - squared - error$$

$$e_{MSE} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} e(m,n)^2 \text{ and} \quad (8)$$

$$the\ peak\ signal - to - noise\ ratio$$

$$PSNR = 10\log_{10} \frac{255^2}{e_{MSE}}. \text{ [dB]} \quad (9)$$

The influence of the value of *threshold* and *variance threshold* on the distribution of areas in segmented image Lena is depicted in Table 1. We selected very closed values of *PSNR* to compare the dynamic criterion and the variance criterion each other. We have calculated *normalized # of areas*.

(a) threshold = 0.3, PSNR = 26.96 dB

(b) variance threshold = 1.2745, PSNR = 27.01 dB

Fig. 2 Images segmented by (a) min_max criterion and (b) variance criterion. In this figure, each region is filled with the corresponding grey level.

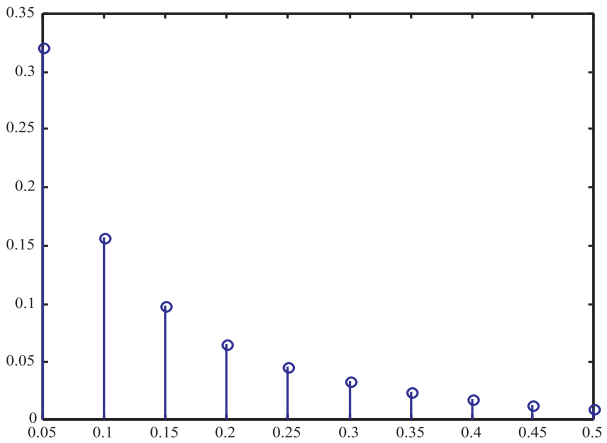
$$normalized\ \#\ of\ areas = \frac{\#\ of\ areas}{maximal\ \#\ of\ areas}, \quad (10)$$

where *maximal # of areas* = *M*N*. (11)

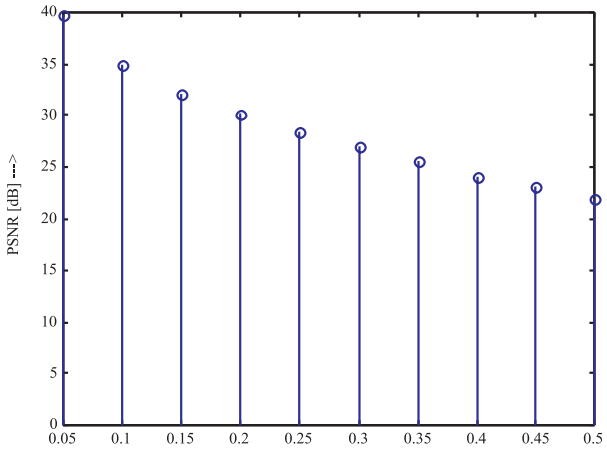
In our case: *maximal # of areas* = 512*512 = 262144.

The distribution of areas in the segmented image Lena for dynamic and variance criterions. Table 1

	Dynamic criterion - threshold			Variance - variance_threshold		
Size of Area [pixels x pixels]	0.1	0.3	0.5	0.2157 (55)	1.2745 (325)	3.4314 (875)
512 x 512	0	0	0	0	0	0
256 x 256	0	0	0	0	0	0
128 x 128	0	0	0	0	0	2
64 x 64	0	4	14	3	11	16
32 x 32	14	65	98	30	69	79
16 x 16	166	283	234	207	215	161
8 x 8	1041	886	486	815	631	378
4 x 4	3909	2140	655	2540	1688	736
2 x 2	13329	3542	712	9630	4382	1157
1 x 1	22828	1640	112	34824	6472	1132
Normalized # of Areas	0.1575 (41287)	0.0327 (8560)	0.0088 (2311)	0.1833 (48049)	0.0514 (13468)	0.0140 (3799)
PSNR [dB]	34.9021	26.9573	21.9891	34.5415	27.0141	22.0196

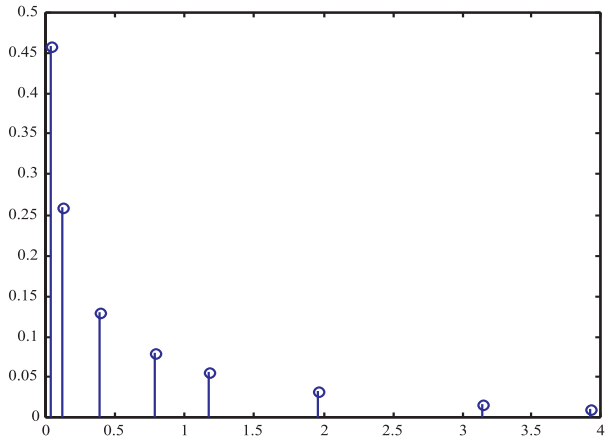


(a)

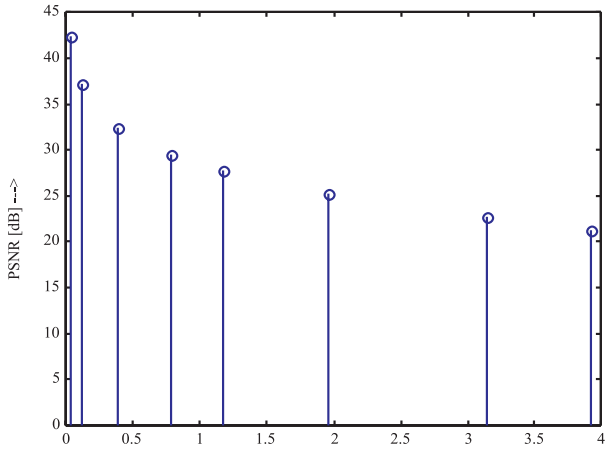


(b)

Fig. 3 (a) normalized # of areas versus threshold, (b) PSNR versus threshold.



(a)



(b)

Fig. 4 (a) normalized # of areas versus variance threshold. (b) PSNR versus variance threshold.

PSNR and *normalized # of areas* versus *threshold* and *variance threshold* are shown in Figure 3 and Figure 4, respectively.

We have calculated the correlation coefficients (9), (10), which are normalized measures of the strength of the linear relationship [7] between variable *PSNR* and variable *normalized # of areas*.

$$corrcoef\,(PSNR,\ normalized\ \#\ of\ areas) = \\ = \begin{bmatrix} 1 & 0.9352 \\ 0.9352 & 1 \end{bmatrix} \qquad (12)$$

It is seen that the *PSNR* and the *normalized # of areas* are highly correlated data (nearly equivalent data) for *min_max* criterion (12) and *variance criterion* (13). Therefore, it is enough to evaluate only the influence of noise on *normalized # of areas* and not to calculate *PSNR*. We have to be aware of the fact that the *PSNR* criterion depends also on the methods which are used for the coding of areas. If we evaluate only *normalized # of areas* and do not evaluate *PSNR* we will probably eliminate the problem of area coding.

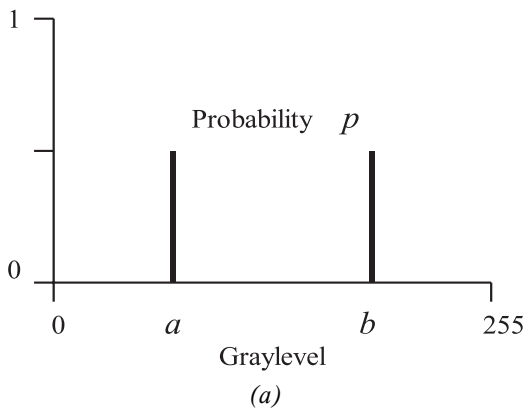
$$corrcoef\,(PSNR,\ normalized\ \#\ of\ areas) = \\ = \begin{bmatrix} 1 & 0.9544 \\ 0.9544 & 1 \end{bmatrix} \qquad (13)$$

Noise

Image capture mechanisms are not ideal. Therefore, noise is found in the image. This noise hinders the ability to effectively process and compress the image. The noise affects also the image segmentation and the image coding.

In our experiment we have added salt and pepper noise to the image. The salt and pepper noise occurs e.g. in images which are obtained by camera containing malfunctioning pixels or can occur due to a random bit error in a communication channel [6]. Its histogram is defined as

$$h_i = \begin{cases} \text{pepper noise with probability } p & \text{for } G_i = a \\ \text{salt noise with probability } p & \text{for } G_i = b \\ 0 & \text{elsewhere} \end{cases}, \qquad (14)$$



(a)

Fig. 5 (a) A histogram of salt and pepper noise. (b) The salt and pepper noise degraded image with a combined probability (density) of 5 %. (MATLAB's default value) PSNR = 18.5 dB.

where G_i is the grey level value of the image. The salt and pepper noise each occur at grey level values a and b with probability p (see Figure 5 (a)). The influence of noise on the image is shown in Figure 5 (b).

The graphs of *normalized # of areas* versus *threshold* (dynamic criterion) and *variance threshold* (variance criterion) for different values of noise density are shown in Figure 6.

Conclusion

We conclude that the number of areas in the image and PSNR for coding of areas by mean values of appropriate areas are highly correlated data. Therefore, we think that it is enough to evaluate only number of areas in the image and not to evaluate PSNR for different values of noise density and splitting control parameters.

The number of areas, which is produced by the dynamic splitting criterion, is smaller than the one produced by the variance criterion for equal quality (i.e. PSNR). Thus, the coding of borders of image areas which are obtained by dynamic splitting criterion spends less bits than the coding of areas borders obtained by variance splitting criterion (see Table 1).

The noise affects significantly the image segmentation method. When the noise density is increasing the number of areas in image is also increasing (see Figure 6). Nevertheless, we have to be aware that the quality of image does not improve when the number of areas in the image increases, because we do not have the original noiseless image for coding of image.

Reviewed by: J. Polec, M. Brezňan

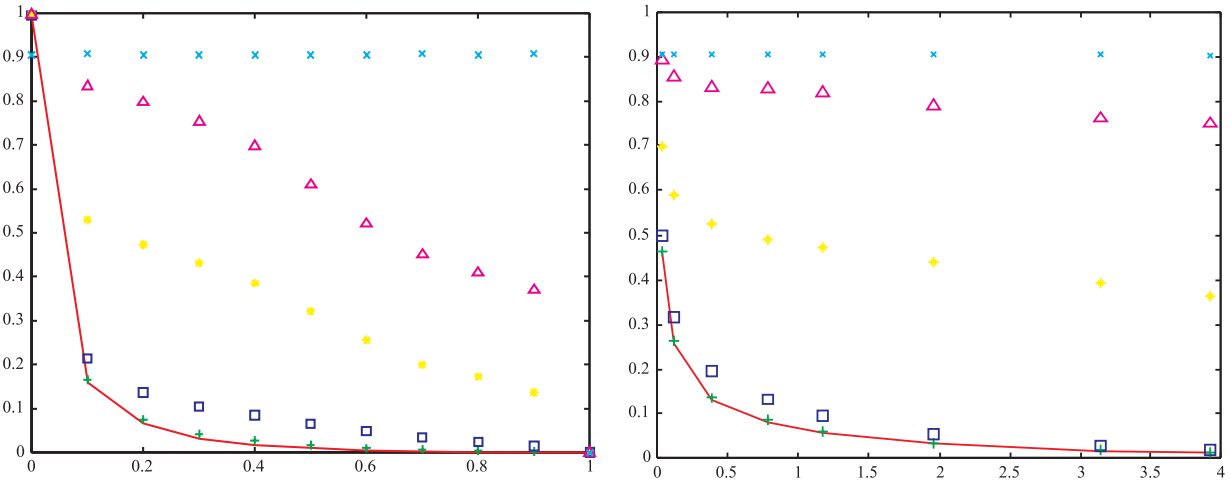


Figure 6 normalized # of areas versus (a) threshold and (b) variance threshold for quadtree decomposition and different noise densities. Density = 0 (full line), 0.001(+++), 0.01 (square), 0.1 (***), 0.3 (triangle), 1 (xxx). Noise density ∈ ⟨0,1⟩.

Literatúra - References

[1] CORTEZ, D., NUNES, P., SEQUEIRA, M. M., PEREIRA, F.: Image Segmentation Towards New Image Representation Methods. Signal Processing: Image Comm., vol. 6, Nov. 1995, pp. 485-498

[2] VAISEY, J., GERSHO, A.: Image Compression with Variable Block Size Segmentation. IEEE Trans. Signal Processing, vol. 40, Aug. 1992, pp. 2040-2060

[3] KARPF, M.: Video Coding, Second Generation Methods, journal T&P, ISSN 1211-5525, July 1999, pp. 14-15

[4] CHEEVASUVIT, F., MAITRE, H., VIDAL-MADJAR, D.: A Robust Method for Picture Segmentation Based on a Split and Merge Procedure. Département I.S.S.V. - Laboratoire Image. ENST - 84C006, Aout 1984, 25 p.

[5] WILLEMIN, P., REED, T. R., KUNT, M.: Image Sequence Coding by Split and Merge. IEEE Trans. on Communications, vol. 39, No. 12., December 1991, pp. 1845-1855

[6] LIM, J.S., LIM J.S.: Two-Dimensional Signal and Image Processing. Prentice Hall, November 1989, 880 p.

[7] Using MATLAB. The MathWorks, Inc, June 1997, 410 p.

https://doi.org/10.26552/com.C.2000.2.21-29

Pavol Tomašov - Karol Rástočný - Jiří Zahradník *

ČASTI MODELU INFORMAČNÝCH A ZABEZPEČOVACÍCH SYSTÉMOV

PARTS OF MODEL OF INFORMATION AND SAFETY SYSTEMS

Modelovacie techniky pre rozsiahle a zložité systémy sa vyberajú s ohľadom na očakávaný výsledok modelovania. Pre rutinný postup syntézy vo väčšine prípadov postačí skúsenostný prístup. Model systému s exaktným opisom jeho atribútov vyžaduje aplikáciu niektorých teoretických záverov. Pre informačné a zabezpečovacie systémy je zvolený model, vychádzajúci z teórie informácií. Tento model možno ďalej precízovať pomocou systémovej a nakoniec obvodovej teórie. V predkladanom článku sú opísané dva nástroje aparátu teórie informácií pre spracovanie modelu systému. Ide o vyjadrenie nerovnosti pre spracovanie informácií a o kvantifikáciu chýb pri manipulácii s informáciou.

Úvod

Pri analýze a syntéze informačného a zabezpečovacieho systému existujú dva základné postupy:

- pre dohodnuté funkcie sa vyberá štruktúra technických a programových prostriedkov systému na základe skúseností projektanta z predošlých aplikácií. Výber je závislý od technologickej úrovne komponentov systému a od celkovej sumy, ktorú je odberateľ ochotný zaplatiť. Úloha riadenia systému je zložená zo „štandardných elementov“,
- podrobne sa špecifikujú požadované funkcie systému vo väzbe na riadený systém bez ohľadu na budúcu skladbu HW a SW komponentov. V ďalšom kroku sa vytvorí model funkcií, na ktorom sa definuje úloha riadenia ako čiastková úloha radenia primárneho procesu. Pri tvorbe modelu funkcií možno zobrať za základ niektorý z referenčných modelov, ak ide o obvyklý sortiment služieb. Pre osobitný sortiment služieb treba vykonať podobný postup, aký sa použil pri tvorbe referenčných modelov. Účinným prostriedkom je vrstvenie funkcií. Tento postup umožňuje rozklad úlohy syntézy na jednoduchšie moduly (vrstvy funkcií). Ak sa vyrieši spolupráca vrstiev, možno pri syntéze použiť postup, známy z objektového programovania. Na realizáciu funkcií vrstiev sa v poslednej fáze vyberú HW a SW komponenty. Charakteristiky vykonávania funkcie vrstiev (časové, objemové, spoľahlivostné, bezpečnostné, ...) sú závislé od tech-

Modelling techniques for large and composite systems are chosen with regard to the expected result of modelling. For routine procedure of synthesis the empirical method is sufficient in most cases. System model with exact description of its attributes demands application of some theoretical conclusions. For information and safety systems a model derived from Information theory is selected. This model can then be specified with the help of the system and, eventually, circuit theory. In the presented paper two tools of Information theory apparatus for model system processing are described. They express information processing inequality and quantification of faults occurring by information manipulation.

Introduction

When analysing and synthesising an information and safety system two ground lines exist:

- for denominated functions a structure of technical and program elements of the system is selected based on a designer's experience from the last application. The selection depends on technology level of system components and on general sums which the customer is willing to pay. The task of system control consists of "standard elements",
- the demanded functions of the system are thoroughly specified in connection with the controlled system regardless of the future composition of HW and SW components. In the next step the function model is created on the basis of which a control task is defined as a partial task of the primary process control. When creating a function model it is possible to use some of the reference models as a basis provided that a usual range of services is demanded. For a special range of services a similar procedure to the one used during the creation of reference models, has to be performed. The interleaving of functions is an effective tool. This procedure enables decomposition of synthesis responsibilities on simple modules (layers of the functions). Provided that the co-operation of the layers is solved, the procedure well-known from the object programming can be used in the synthesis. HW and SW components are chosen for

* **[1]Prof. Ing. Pavol Tomašov, PhD., [2]Doc. Ing. Karol Rástočný, PhD., [3]Doc. Ing. Jiří Zahradník, PhD.**
[1]University of Žilina, Faculty of Electrical Engineering, Veľký diel NF, SK-01026 Žilina, Slovak Republic,
Tel.: +42-89-5133262, Fax:+421-89-652241, E-mail: tomas@fpedas.utc.sk
[2]Tel.: +42-89-5655559, Fax: +421-89-652241, E-mail: rastoc@fel.utc.sk
[3]Tel.: +42-89-5655559, Fax: +421-89-652241, E-mail: zahra@fel.utc.sk

nologického stupňa použitých komponentov a od „šikovnosti“ vytvorenia modelu funkcií.

V mnohých aplikáciách je postačujúci prvý postup. V zložitých a rozsiahlych informačných systémoch sa niekedy vyžaduje osobitný sortiment služieb. Príkladom sú zabezpečovacie systémy, použité pri riadení kritických procesov. Typickým kritickým procesom je dopravný proces. Pri výskyte poruchy v riadení takého systému môže dôjsť k stratám na majetku, zdraví alebo živote. Porucha, ktorá vyvolá nebezpečný stav, môže byť aj produktom použitého systému. Je zrejmé, že funkcie takéhoto systému treba špecifikovať druhým z uvedených postupov. Skupinu funkcií v jednotlivých skupinách (vrstvách) treba zostaviť tak, aby bola zaručená stabilita, kauzalita a bezpečnosť. Výsledkom špecifikácie funkcií celého systému má byť „safety case“ pre konkrétnu aplikáciu.

Činnosť informačného aj zabezpečovacieho systému môže byť rozložená na štyri základné druhy služieb: získavanie, úschova, prenos a transformácia relevantných, aktuálnych a garantovaných informácií (obr. 1). Každá z čiastkových služieb je spojená so štruktúrou HW a SW prostriedkov, ktoré vykonávajú „technológiu“ príslušných operácií. Aby systém poskytoval požadovaný sortiment služieb, musia byť elementy služieb zostavené do sekvencií, ktorých vykonávanie je riadené. Charakteristiky vykonávania funkcií sú silne závislé aj od spôsobu riadenia. Už pri zostavovaní sekvencie čiastkových služieb musí byť zohľadnený technologický stupeň elementov systému. Napríklad výkonnosť počítačovej siete ovplyvňuje distribuovanosť DB systémov, spôsob a intervaly aktualizácie replík DB, atď. Jadrom tejto tézy je vytvorenie riadiaceho postupu, ktorý začína rozkladom služieb systému na primitívy, a končí špecifikáciou protokolu a stanovením príslušného formalizmu na jeho konštrukciu.

Za jadro problému analýzy a syntézy informačných a zabezpečovacích systémov s osobitným sortimentom služieb možno považovať zostavenie modelu, ktorý pokryje čo najväčšie množstvo funkcií a stavov systému. V predkladanom článku je pokus o jednotiaci prístup k modelovaniu takých systémov s využitím niektorých prvkov teórie informácií.

Výber nástrojov pre modelovanie úlohy riadenia

Výber modelovacích techník pre doterajšie technologické stupne informačných a zabezpečovacích systémov vychádza zo skúsenostného postupu. Tento prístup pokrýval takmer všetky požiadavky na riešenie úloh syntézy a analýzy. Išlo napríklad o modely funkcií, model dátových tokov, entitno-relačný model, protokolový model, model porúch a ďalšie modifikované modely. Tieto modely sa dajú zvládnuť špecializovanými programovými balíkmi, takže sú v praxi aj dostatočne efektívne. Majú však aj spoločnú nevýhodu, ktorá obmedzuje ich použitie a efektívnosť pre informačné a zabezpečovacie systémy posledného technologického stupňa. Touto nevýhodou je fakt, že v reťazi: Teória informácií, Teória systémov, Teória riadenia, Teória obvodov, vynechávajú závery teórie informácií, teórie systémov a niektoré aj závery teórie

the realisation of layer functions in the last phase. Characteristics of layer functions execution (up to date, performance, reliability, safety, ...) depend on a technological degree of service components and on the skill of function model creation.

In many applications the first procedure is sufficient. Sometimes a special range of services is demanded in the composite and extensive information systems. An example is the safety systems used for critical processes control. Transportation process is a typical critical process. When a failure occurs in the control of such a system, it may lead to the loss of property, health, or life. The failure that invokes a danger state can be a product of the used service system, too. It is evident that functions of such a system need to be specialised by the second procedure. The function group in single groups (layers) has to be formed in such a way that it guarantees stability, causality and safety. The result of function specification of the whole system has to be the „safety case“ for concrete application.

The activity of information and safety system can be divided into four basic kinds of services: *obtaining, safekeeping, transmission and transformation* of relevant, current and guaranteed information (Fig. 1). Each of the partial services is connected with the structure of HW and SW means performing competent operation “technology“. To enable the system to offer the required range of services the service elements have to be formed into sequences, whose execution is controlled. Characteristics of functions execution are strongly dependent on the kind of control, too. Even when forming the partial services sequence the technological degree of system elements has to be respected. For example the efficiency of computer network is influenced by the level of distribution of DB system, the way and time interval of actualisation of DB replicas, etc. The core of this thesis to create a control procedure, which begins by with decomposition of system services to primitives, and ends by with protocol specification and with estimating the competent formalism on its construction.

Formation of the model that covers the greatest possible number of functions and system states is considered to be the main problem of analysis and synthesis of information and safety system with special range of services. In our paper we try to present a unifying approach to modelling such systems using some elements of information theory.

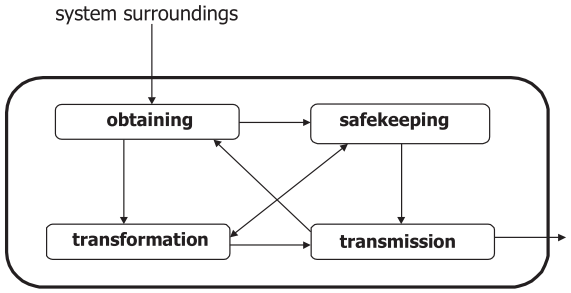
Selection of tools for modelling of control task

The selection of modelling techniques for up to present technological degrees of information and safety systems is based on empirical method. This approach covered almost all of the requirements for solution of synthesis and analysis tasks. Examples include: function models, data flow model, entity-relational model, protocol model, failure model and further modified models. These models can be managed by special software packets, and thus they are sufficiently effective even in praxis. However they all have a shared disadvantage which limits their application and effectiveness for information and safety systems of the last technological degree. In the chain of the Information theory System theory Control theory Circuit theory, all mentioned models fall to

riadenia. Pre rozsiahle a zložité systémy by sa mali modelovacie techniky doplniť najmenej o informačný model. Informácia je pre všetky takéto systémy primárnym „substrátom“, s ktorým sa v systéme manipuluje. Informačný model má preto ambície byť jednotiacim pre doterajšie jednoúčelové modely.

Zabezpečovací systém je podmnožinou informačného systému. Jeho úlohou je narábanie s informáciou (akvizícia, transformácia, prenos a úschova) osobitným spôsobom, odlišne od ostatných podobných činností v informačnom systéme. Otázky štruktúry a správania zabezpečovacieho systému možno rozložiť na riešenie jeho základných procesov.

Na opis systému (štruktúra a správanie) bez započítania dynamiky jeho stavov možno použiť statickú štrukturálnu funkciu. Ak je systém zložený z nezávislých a nekorelovaných elementov, ide o monotónnu štrukturálnu funkciu. Opis systému platí pre vybranú dvojicu jeho stavov (napríklad prevádzkový bezpečný stav a stav s nebezpečnou poruchou).



Obr. 1 Základné operácie pri manipulácii s informáciou
Fig. 1 Basic operations at manipulating with information

Od štrukturálnej funkcie možno prejsť jednoducho k pravdepodobnostnej funkcii, ktorá opisuje pravdepodobnosť jednotlivých stavov systému v niektorom bode časovej osi.

Asi za najdôležitejší model možno považovať ten, ktorý opisuje zabezpečovací systém v procese jeho starnutia. Takýto model musí dovoliť výber rozdelenia pravdepodobnosti výskytu príslušného náhodného parametra (poruchy) a výpočet pravdepodobnosti výskytu zvoleného stavu v potrebnom časovom intervale.

Zabezpečovací systém je súčasťou (subsystémom systému riadenia železničnej dopravy. Pri riadení dopravného procesu možno rozlíšiť tri hierarchické úrovne: procesnú, operatívnu a manažérsku. Riziko vzniku nebezpečenstva je najväčšie na procesnej úrovni. Na tomto riziku sa podieľajú všetky časti systému riadenia dopravy. Zabezpečovací systém má v tomto ohľade významný podiel, pretože stanovuje („vypočítava“) väčšinu povelov na zmenu stavu dopravného procesu. Ak je povel korektný, ide o prevádzkový stav. Ak z nejakých príčin dôjde k nesprávnemu vytvoreniu alebo interpretácii povelu na zmenu stavu, ide o poruchový stav. Tento stav môže, ale nemusí viesť k realizácii ohrozenia, pri ktorom vznikajú škody na majetku, zdraví, živote a životnom prostredí.

Úroveň bezpečnosti preto musí byť odvodená od prípustnej (akceptovateľnej) miery ohrozenia dopravného procesu. Táto úroveň je závislá od chránenej hodnoty a od intenzity dopravného procesu.

Predpokladajme v prvom priblížení, že existuje mechanizmus rozdelenia rizika medzi zabezpečovací systém a ostatné subsystémy riadenia dopravy. Potom možno hovoriť o úrovni bezpeč-

include the conclusions of Information theory System theory and some of them even the conclusions of Control theory. Modelling techniques for large and composite systems should be enlarged by the data model at least. For all information is such systems the primary “sub-slope“ which the system manipulates with. Information model has therefore an ambition to be the unifying one for present special purpose models.

Safety system is a subset of the information system. Its task is to handle the information (acquisition, transformation, transmission and safekeeping) in a specific way, different from other similar activities in the information system. Questions of the structure and behaviour of the safety system can be divided into solution of its basic processes.

Stationary structural function can be used to describe the system (structure and behaviour) without including dynamics of its states. If the system contains independent and non correlated elements, it is a monotonous structural function. Description of the system is valid for a chosen pair of its states (the safe operating state and dangerous failure state).

It is simple to pass from structural function to probability function, which describes the probability of single system state at some point of the time axis.

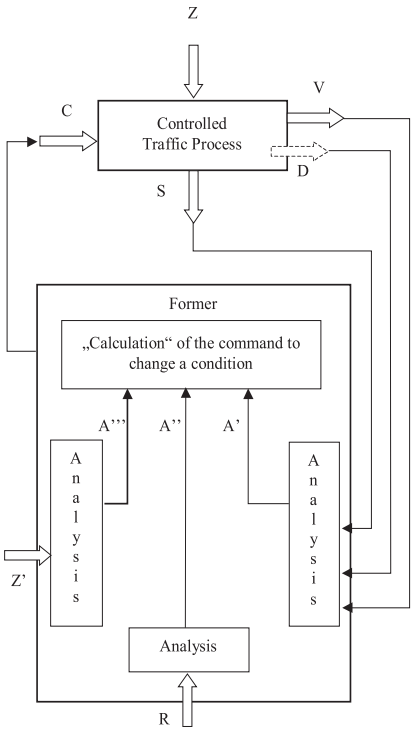
As the most important model can be regarded the one that describes the safety system in its ageing process. Such a model has to permit the selection of probability distribution of occurrence of competent random parameter (failure) and calculation of appearance probability of a chosen state in the necessary time period.

The safety system is a part (subsystem) of the railway traffic control system. In the control of the traffic process three hierarchical levels can be distinguished: procedural, operational and managerial. The highest risk of hazard occurrence exists on the procedural level. This risk is shared by all parts of the traffic control system. A significant role is played by the safety system since it sets (“calculates“) most of the commands given to change the conditions of the traffic process. Provided the command is correct, it results in an operational state. The state is considered faulty if for any reason the command given to change the condition is made incorrectly or misinterpreted. Such a condition can lead (but not in all cases) to an accident which damages property, health, lives and environment. The safety level must therefore be derived from the acceptable hazard rates for the traffic process. This level depends on the protected value and on the traffic process intensity.

Let us first assume that there is a mechanism of risk distribution between the safety system and other subsystems of the traffic control. Then the safety level of the safety system can be regarded as the level of risk of incorrect production and

nosti zabezpečovacieho systému ako o miere, ktorou sa dá vyjadriť riziko nesprávneho vytvorenia a nesprávnej interpretácie tých povelov, ktoré vytvára zabezpečovací systém. Za nesprávne vytvorenie povelu sa pritom považuje aj vytvorenie povelu správnym postupom, ale na podklade nesprávnych vstupných veličín pre jeho vytvorenie. Základná schéma jednostupňového riadenia procesu je na obr. 2.

Pre úlohy analýzy a syntézy systému s definovanou úrovňou bezpečnosti treba stanoviť postupy na zaistenie správania sa systému vo všetkých jeho predvídateľných stavoch. Tieto postupy sa realizujú cez ochranné mechanizmy zabezpečovacieho systému. Ochranné mechanizmy systému musia zaistiť, že aj v prípade výskytu poruchy systém vykonáva svoje funkcie presne podľa vopred definovaného algoritmu. Opatrenia na zaistenie takéhoto správania sa systému možno aplikovať na systémovej úrovni a na úrovni funkčných jednotiek a prvkov systému. Na systémovej úrovni ide predovšetkým o voľbu vhodnej štruktúry systému. Opatrenia na úrovni funkčných jednotiek a prvkov sú zamerané najmä na detekciu poruchy a negáciu jej účinkov.



Legend:
C – the space of control quantities
Z – the space of external effects
V – the space of output quantities
S – the space of state quantities
D – the space of diagnostic quantities
A – the space of the output quantities of the analyser
R – the space of order quantities
Z’ – the space of interference quantities

Obr. 2 Základná schéma jednostupňového riadenia procesu.
Fig. 2 The principal scheme of one-stage control process

Klasifikácia chýb

Zabezpečovací systém sa podieľa na týchto operáciách schémy riadenia podľa obr. 2.:
- získavanie veličín V, D, R, S,
- analýza veličín R, Z′, V, D, S,
- tvorba veličín C,
- prenos potrebných veličín medzi dvoma miestami.

Pri všetkých týchto operáciách môže vzniknúť chyba. Chyby vedú k nesprávnemu vytvoreniu riadiacej veličiny C (povelu na zmenu stavu), alebo k nesprávnej interpretácii riadiacej veličiny (prechod do neprísľušného stavu v priestore veličín S). Pre definovanie úrovne bezpečnosti treba tieto chyby klasifikovať a nájsť opatrenia na zaručenie akceptovateľného výskytu (pravdepodobnosti alebo intenzity) nezistených, alebo neošetrených chýb.

misinterpretation of the commands produced by the safety system. Commands resulting from a correct procedure but using incorrect input quantities are also considered incorrect commands. The principal scheme of the one-stage process control is shown in Fig. 2.

For the tasks of analysis and synthesis of the system with a defined level of safety the procedures ensuring the behaviour of the system in all its predictable states must be defined.

These procedures are realised through the defence mechanisms of the safety system. They have to ensure fulfilling of the demanded functions according to the pre-defined algorithm even in the case of failure. Precautions taken to ensure such system behaviour can be applied on the system level as well as on the level of functional units and system components. On the system level the choice of the appropriate system structure is involved above all. Precautions taken on the level of functional units and components aim mainly at fault detection and negation of fault effects.

Fault Classification

The safety system takes part in the following operations of the control scheme shown in Fig. 2:
- Obtaining V, D, R, S quantities
- Analysing R, Z′, V, D, S quantities
- Producing C quantities
- Transmission of required quantities between 2 places.

A fault may occur in all of these operations. Faults result in an incorrect production of the control C quantity (the command for a change of state) or in misinterpretation of the control quantity (transition to an unauthorised state in the area of S quantities). To define the safety level these faults must be classified and precautions that can guarantee acceptable occurrence (probability or rate) of unidentified or unattended faults must be taken.

Všetky časti zabezpečovacieho systému, ktoré získavajú veličiny V, D, R, S, analyzujú veličiny R, Z′, V, D, S, vytvárajú veličiny C a podieľajú sa na prenose všetkých veličín riadeného systému, možno takmer bez výnimky považovať za nejakú podobu konečného automatu (KA). Chyby, ktoré môžu vzniknúť v činnosti KA, možno klasifikovať do tried:

a) chyby v jazyku,
b) chyby v prenose zo vstupu na výstup KA,
c) chyby formátu,
d) chyby správavania (kauzality vykonávania čiastkových funkcií),
e) chyby poskytovania služby vyššiemu systému (chyby kauzality služieb).

Na kompletný opis systému treba zostaviť model, ktorý okrem uvedených skutočnosti umožní zarátať aj účinky operačného prostredia, teda kombinovať dva a viac náhodných procesov, ktoré môžu mať rozdielny charakter rozdelenia pravdepodobnosti.

Ak predpokladáme, že stavy objektu sa dajú opísať ako náhodné premenné, potom možno s výhodou použiť aparát teórie informácií na vytvorenie charakteristík toku porúch (pri analýze), alebo na opis modifikácie takého toku (pri syntéze). Ide o nasledujúce časti informačnej teórie.

Stavy objektu ako náhodné premenné:

Nech sú stavy objektu považované za náhodné premenné X_1, X_2, ... X_n. Ich vlastnosti sú dostatočne opísané funkciou rozdelenia pravdepodobnosti $p(x_1, x_2, ... x_n)$. Premenné $X_1, X_2, ... X_n$ môžu byť identicky rozdelené podľa niektorého typu rozdelenia pravdepodobnosti. Môžu byť nezávislé, podmienene závislé, alebo štatisticky závislé. Pri známom rozdelení pravdepodobnosti náhodných premenných sa dá stanoviť entropia stavov objektu.

Entropia umožňuje opísať objekt v potrebnej forme, napríklad pri tvorbe kódu, ktorým je opísaný celkový stav objektu.

Opis stavov objektu pomocou nerovnosti pre spracovanie dát

Predpokladajme, že stavy objektu tvoria Markovovu reťaz. Nerovnosť pre spracovanie dát sa použije na demonštráciu, že žiadna „šikovná“ manipulácia s údajmi nemôže zlepšiť výpočet stavových charakteristík.

Definícia: Náhodné premenné *X*, *Y*, *Z* tvoria Markovovu reťaz v tomto poradí, ak podmienené rozdelenie *Z* závisí len od *Y* a je podmienene nezávislé od *X*. Premenné *X*, *Y*, *Z* tvoria Markovovu reťaz $X \rightarrow Y \rightarrow Z$, ak spoločná pravdepodobnostná funkcia sa dá napísať takto:

$$p(x, y, z) = p(x).\ p(y \mid x).p(z \mid y). \qquad (1)$$

Z toho vyplývajú niektoré jednoduché dôsledky:

- $X \rightarrow Y \rightarrow Z$ vtedy a len vtedy, ak *X* a *Z* sú podmienene nezávislé pre dané *Y*. Implicitne je v tom podmienená nezávislosť, pretože

All parts of the safety system that obtain V, D, R, S quantities, analyse R, Z′, V, D, S, quantities produce C quantities and take part in the transmission of all the controlled system quantities can be regarded (almost without exception) as a certain kind of the finite automaton. Faults that can occur in operation of the finite automaton may be classified into the following classes:

a) Language faults
b) Faults in transmission from the input to the output of the finite automaton
c) Format faults
d) Behaviour faults (faults in causality of performing partial functions)
e) Faults in providing services to the superior system (faults in service causality).

For a complete system description we need to create a model that, apart from the mentioned facts, enables to incorporate the effects of the operational surroundings as well- thus to combine two and more random processes, which can have different character of probability distribution.

Supposing that the object states can be described as random variables, the information theory apparatus can be conveniently used to create characteristics of fault flow (during analysis) or to describe the modification of such a flow (during synthesis). The following parts of information theory are involved.

Object states as random variables

Let the object states be regarded as the random variables X_1, X_2, ... X_n. Their characteristics are sufficiently described by the probability distribution function $p(x_1, x_2, ... x_n)$. Variables X_1, X_2, ... X_n can be identically sorted by some type of probability distribution. They can be independent, conditionally dependent or statistically dependent. When the probability distribution of the random variables is known, entropy of the object states can be estimated.

Entropy enables to describe the object in the necessary form, e.g. during the creation of the code, by which is the comprehensive object state described.

Let us suppose that object states create a Markov chain. The data processing inequality can be used to show that clever manipulation with the data cannot improve the computation of state characteristics.

Definition: Random variables *X*, *Y*, *Z* form a Markov chain in this order if the conditional distribution of *Z* depends only on *Y* and is conditionally independent from *X*. Specifically, variables *X, Y* and *Z* form a Markov chain $X \rightarrow Y \rightarrow Z$ if the joint probability mass function can be written as:

$$p(x, y, z) = p(x).\ p(y \mid x).p(z \mid y). \qquad (1)$$

Some simple consequences result:

- $X \rightarrow Y \rightarrow Z$ if and only if *X* and *Z* are conditionally independent for given *Y*. Markovity implies conditional independence because

$$p(x, y \mid z) = \frac{p(x, y, z)}{p(y)} = \frac{p(x, y)p(z \mid y)}{p(y)} = p(x \mid y).p(z \mid y) \quad (2)$$

Takto sú charakterizované Markovove reťaze, ktoré môžu byť rozšírené na definované Markovove polia. Sú to n-rozmerné náhodné procesy, v ktorých vonkajšok a vnútrajšok je nezávislý voči danej hranici.

- $X \rightarrow Y \rightarrow Z$ implikuje $Z \rightarrow Y \rightarrow X$. Tieto podmienky možno zapísať aj takto: $X \leftrightarrow Y \leftrightarrow Z$.
- Ak $Z = f(Y)$, potom $X \rightarrow Y \rightarrow Z$.

Teraz už možno dokázať dôležitú teorému, demonštrujúcu, že žiadne spracovanie Y (determinované alebo náhodné) nemôže zvýšiť informáciu, že Y vypovedá o X.

Teoréma 1: Ak $X \rightarrow Y \rightarrow Z$, potom $I(X;Y) \geq I(X;Z)$.

Dôkaz: Podľa reťazového pravidla môžeme rozšíriť vzájomnú informáciu dvoma spôsobmi:

$$I(X;Y,Z) = I(X;Z) + I(X;Y \mid Z) \quad (3)$$
$$= I(X;Y) + I(X;Z \mid Y). \quad (4)$$

Pretože X a Z sú podmienene nezávislé pre dané Y, je $I(X;Z \mid Y) = 0$. Pretože $I(X;Y \mid Z) \geq 0$, dostávame

$$I(X;Y) \geq I(X;Z). \quad (5)$$

Rovnosť platí vtedy a len vtedy, ak $I(X;Y \mid Z) = 0$, t. j. $X \rightarrow Y \rightarrow Z$ vytvára Markovovu reťaz. Podobne sa dá dokázať, že $I(Y;Z) \geq I(X;Z)$.

Dôsledok: Pre zvláštny prípad, ak $Z = g(Y)$, je $I(X;Y) \geq I(X;g(Y))$.

Dôkaz: $X \rightarrow Y \rightarrow g(Y)$ tvorí Markovovu reťaz. Funkcia $g(Y)$ nemôže zvýšiť informáciu o premennej X.

Dôsledok: Ak $X \rightarrow Y \rightarrow Z$, potom $I(X;Y \mid Z) \leq I(X;Y)$.

Dôkaz: Z rovnice (3) a (4) a použitím faktu, že $I(X;Z \mid Y) = 0$ a $I(X;Z) \geq 0$, dostávame $I(X;Y \mid Z) \leq I(X;Y)$

Závislosť X a Y je zmenšená (alebo aspoň nezmenená) pozorovaním „poklesu“ náhodnej premennej Z.

Všimnime si, že môže byť aj $I(X;Y \mid Z) \geq I(X;Y)$ vtedy, ak X, Y, Z netvoria Markovovu reťaz. Napríklad nech X a Y sú nezávislé lineárne náhodné premenné a nech $Z = X + Y$. Potom $I(X;Y) = 0$, ale $I(X;Y \mid Z) = H(X \mid Z) - H(X \mid Y,Z) = H(X \mid Z) = P(Z_{=1})$. $H(X \mid Z_{=1}) = 0{,}5$ bit.

Použitie chybovej nerovnosti pre analýzu bezpečnosti

Použitie chybovej nerovnosti je kľúčové pri rozbore bezpečnosti zabezpečovacieho systému a jeho elementov. Dá sa očaká-

$$p(x, y \mid z) = \frac{p(x, y, z)}{p(y)} = \frac{p(x, y)p(z \mid y)}{p(y)} = p(x \mid y).p(z \mid y) \quad (2)$$

This is the characterisation of Markov chains that can be extended to define Markov fields, which are n-dimensional random processes in which the interior and exterior are independent from the given values of the boundary.

- $X \rightarrow Y \rightarrow Z$ implies that $Z \rightarrow Y \rightarrow X$. Thus the condition is sometimes written $X \leftrightarrow Y \leftrightarrow Z$.
- If $Z = f(Y)$, then $X \rightarrow Y \rightarrow Z$.

We can now prove an important and useful theorem demonstrating that no processing of Y, deterministic or random, can increase the information that Y states about X.

Theorem 1: If $X \rightarrow Y \rightarrow Z$, then mutual information $I(X;Y) \geq I(X;Z)$.

Proof: According to the chain rule, we can expand mutual information in two different ways

$$I(X;Y,Z) = I(X;Z) + I(X;Y \mid Z) \quad (3)$$
$$= I(X;Y) + I(X;Z \mid Y). \quad (4)$$

Since X and Y are conditionally independent for the given Y, $I(X;Z \mid Y) = 0$. Since $I(X;Y \mid Z) \geq 0$, we get

$$I(X;Y) \geq I(X;Z). \quad (5)$$

Equality is valid if and only if $I(X;Y \mid Z) = 0$, i.e. $X \rightarrow Y \rightarrow Z$ forms a Markov chain. Similarly can be proven that $I(Y;Z) \geq I(X;Z)$.

Corollary: In specific case, if $Z = g(Y)$, we get $I(X;Y) \geq I(X;g(Y))$.

Proof: $X \rightarrow Y \rightarrow g(Y)$ forms a Markov chain. Function $g(Y)$ cannot increase the information about variable X.

Corollary: If $X \rightarrow Y \rightarrow Z$, then $I(X;Y \mid Z) \leq I(X;Y)$.

Proof: From (3) and (4), and using the fact that $I(X;Z \mid Y) = 0$ by Markovity and $I(X;Z) \geq 0$, we get $I(X;Y \mid Z) \leq I(X;Y)$

Thus the dependence of X and Y is decreased (or remains unchanged) by the observation of the decrease of random variable Z.

Note that it is also possible that $I(X;Y \mid Z) \geq I(X;Y)$ when X, Y and Z do not form a Markov chain. For example, let X and Y be independent linear random variables, and let $Z = X + Y$. Then $I(X;Y) = 0$, but $I(X;Y \mid Z) = H(X \mid Z) - H(X \mid Y,Z) = H(X \mid Z) = P(Z_{=1})$. $H(X \mid Z_{=1}) = 0{,}5$ bit.

Application of error inequality for analysing the safety

The application of error inequality is key-important when analysing safety system and its components. The estimation of X

vať, že odhad X (vybraného stavu objektu) je možný s malou pravdepodobnosťou chyby len vtedy, ak podmienená entropia $H(X|Y)$ je malá (Y je vyjadrenie pozorovaného stavu X cez prostredníka, ktorým môže byť signál na výstupe obvodu). Chybová nerovnosť túto myšlienku kvantifikuje.

Rozšírime dôkaz kódov s nulovou pravdepodobnosťou chyby aj na kódy s malou pravdepodobnosťou chyby. Novou zložkou bude chybová nerovnosť, ktorá stanovuje spodnú hranicu pravdepodobnosti chyby v pojmoch podmienenej entropie.

Index W je rovnomerne rozdelený na množine $W = \{1, 2, ..., 2n^R\}$ a sekvencia Y^n je pravdepodobnostne zviazaná s W. Zo sekvencie Y^n odhadujeme vyslaný index W. Nech je odhad $\hat{W} = g(Y^n)$. Definujme pravdepodobnosť chyby

$$P^{(n)}_e = Pr(\hat{W} \neq W). \tag{6}$$

Ďalej definujme

$$E = 1, \text{ak } (\hat{W} \neq W), \quad E = 0, \text{ak sa } \hat{W} = W. \tag{7}$$

Použijeme reťazové pravidlo pre entropiu na rozšírenie $H(E,W|Y^n)$. Dostaneme:

$$H(E,W|Y^n) = H(W|Y^n) + H(E|W,Y^n) \tag{8}$$
$$= H(E|Y^n) + H(W|E,Y^n). \tag{9}$$

Pretože E je funkciou W a $g(Y^n)$ musí byť $H(E|W,Yn) = 0$. Tiež $H(E) \leq 1$, pretože E je binárna náhodná premenná. Posledný termín $H(W|E,Y^n)$ možno ohraničiť takto:

$$H(W|E,Y^n) = P(E=0)H(W|(Y^n,E=0) + P(E=1)H(W|Y^n,E=1) \tag{10}$$
$$\leq (1-P^{(n)}_e(E=0) + P^{(n)}_e \log(W-1) \tag{11}$$
$$\leq P^{(n)}_e nR, \tag{12}$$

pretože pri danom $E = 0$, $W = g(Y^n)$ a keď je $E = 1$, môžeme dostať hornú hranicu podmienenej entropie. Kombinovaním týchto výsledkov dôjdeme k chybovej nerovnosti:

$$H(W|Y^n) \leq 1 + P^{(n)}_e nR \tag{13}$$

Pretože pre pevný kód je $X^n(W)$ funkciou W, platí

$$H(X^n(W)(Y^n) \leq H(W(Y^n)). \tag{14}$$

Lema: (chybová nerovnosť): Pre diskrétny kanál bez pamäti s kódovou knihou ξ a s rovnomerne rozdelenými vstupnými správami nech platí: $P^{(n)}_e = Pr(W \neq g(Y^n))$.

Potom je

$$H(X^n|Y^n) \leq 1 + P^{(n)}_e nR. \tag{15}$$

Teraz dokážeme túto lemu, ktorá ukazuje, že kapacita kanála na jeden prenos sa nezvyšuje, ak použijeme diskrétny kanál bez pamäti viackrát.

(chosen object state) with small error probability is possible only if the conditioned entropy $H(X|Y)$ is small (Y is the observed state of X expressed through an intermediary, which can be a circuit output signal). Error inequality quantifies this idea.

We now extend the proof that was derived for zero-error codes to the case of codes with very small error probability. The new ingredient will be error inequality, which defines a lower boundary of the error probability in terms of the conditional entropy.

The index W is uniformly distributed on the set $W = \{1, 2, ..., 2n^R\}$, and the sequence Y^n is probabilistically related to W. From Y^n, we estimate the index W that was sent. Let the estimation be $\hat{W} = g(Y^n)$. Let us define the error probability

$$P^{(n)}_e = Pr(\hat{W} \neq W). \tag{6}$$

Next, we define

$$E = 1, \text{ak } (\hat{W} \neq W), \quad E = 0, \text{ak sa } \hat{W} = W. \tag{7}$$

Then using the chain rule for entropies to expand $H(E,W|Y^n)$. we get

$$H(E,W|Y^n) = H(W|Y^n) + H(E|W,Y^n) \tag{8}$$
$$= H(E|Y^n) + H(W|E,Y^n). \tag{9}$$

Now, since E is a function of W and $g(Y^n)$, inevitably $H(E|W,Yn) = 0$. Also $H(E) \leq 1$, since E is a binary valued random variable. The remaining term, $H(W|E,Y^n)$, can be bounded as follows:

$$H(W|E,Y^n) = P(E=0)H(W|(Y^n,E=0) + P(E=1)H(W|Y^n,E=1) \tag{10}$$
$$\leq (1-P^{(n)}_e(E=0) + P^{(n)}_e \log(W-1) \tag{11}$$
$$\leq P^{(n)}_e nR, \tag{12}$$

since by given $E = 0$, $W = g(Y^n)$, and when $E = 1$, we can get the upper boundary of the conditional entropy. Combining these results, we obtain error inequality:

$$H(W|Y^n) \leq 1 + P^{(n)}_e nR \tag{13}$$

Since for a fixed code $X^n(W)$ is a function of W,

$$H(X^n(W)(Y^n) \leq H(W(Y^n)). \tag{14}$$

Lema: (error inequality). For a discrete memoryless channel with a codeboock (and the input messages uniformly distributed, let $P^{(n)}_e = Pr(W \neq g(Y^n))$.

Then

$$H(X^n|Y^n) \leq 1 + P^{(n)}_e nR. \tag{15}$$

We will now prove this lema which shows that the channel capacity per one transmission is not increased if we use a discrete memoryless channel many times.

Lema: Nech Y^n je výsledok prispôsobenia X^n na diskrétny kanál bez pamäti. Potom

$$I(X^n;Y^n) \leq nC, \text{ pre všetky } p(x^n). \tag{16}$$

Dôkaz:

$$I(X^n;Y^n) = H(Y^n) - H(Y^n \mid X^n) \tag{17}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid Y_1, ..., Y_{i-1}, X^n) \tag{18}$$

$$= \sum_{i=1}^{n} H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i), \tag{19}$$

pretože podľa definície diskrétneho kanála bez pamäti Y_i závisí len od X_i a je podmienene nezávislé od všetkých ostatných. Pokračovaním série nerovností je:

$$I(X^n;Y^n) = H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i) \tag{20}$$

$$\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i \mid X_i) \tag{21}$$

$$= I(X_i;Y_i) \tag{22}$$

$$\leq nC, \tag{23}$$

kde (21) vyplýva z faktu, že entropia súboru náhodných premenných je menšia ako súčet ich individuálnych entropií. Výsledok (23) vyplýva z definície kapacity. Tým je lema dokázaná.

Teraz treba dokázať konverziu kódovacej teorémy.

Dôkaz: Máme ukázať, že akákoľvek sekvencia $(2^{nR},n)$ kódov so $\lambda^{(n)} \to 0$ musí mať $R \leq C$.

Ak sa maximálna pravdepodobnosť chyby blíži k nule, potom priemerná pravdepodobnosť chyby pre sekvenciu kódov sa tiež blíži k nule, t. j. $\lambda^{(n)} \to 0$ implikuje $P^{(n)}_e \to 0$, kde $P^{(n)}_e$ je definovaná ako priemerná pravdepodobnosť chyby pre kód (M,n): $P^{(n)}_e = \frac{1}{M} \sum_{i=1}^{M} \lambda_i$ Nech je pre každé n zobrazené W podľa rovnomerného rozdelenia cez $\{1, 2, ..., 2^{nR}\}$. Pretože W má rovnomerné rozdelenie, je $P^{(n)}_e = Pr(\hat{W} \neq W)$.

Preto je

$$nR = H(W) = H(W \mid Y^n) + I(W;Y^n) \tag{24}$$

$$\leq H(W \mid Y^n) + I(X^n(W);Y^n) \tag{25}$$

$$\leq 1 + P^{(n)}_e nR + I(X^n(W);Y^n)\, P^{(n)}_e \tag{26}$$

$$\leq 1 + P^{(n)}_e nR + nC \tag{27}$$

Po vydelení n bude:

$$R \leq P^{(n)}_e R + \frac{1}{n} + C \tag{28}$$

Pre $n \to \infty$ idú prvé dva členy na pravej strane k nule a preto

$$R \leq C. \tag{29}$$

Lema: Let Y^n be the result of passing X^n through a discrete memoryless channel. Then

$$I(X^n;Y^n) \leq nC, \text{ pre všetky } p(x^n). \tag{16}$$

Proof:

$$I(X^n;Y^n) = H(Y^n) - H(Y^n \mid X^n) \tag{17}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid Y_1, ..., Y_{i-1}, X^n) \tag{18}$$

$$= \sum_{i=1}^{n} H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i), \tag{19}$$

since by the definition of a discrete memoryless channel, Y_i depends only on X_i and is conditionally independent from everything else. Continuing the series of inequalities, we get

$$I(X^n;Y^n) = H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i) \tag{20}$$

$$\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i \mid X_i) \tag{21}$$

$$= I(X_i;Y_i) \tag{22}$$

$$\leq nC, \tag{23}$$

where (21) follows from the fact that the entropy of a collection of random variables is less than the sum of their individual entropies, and (23) follows from the definition of capacity. Thus we have proved that using the channel many times does not increase the information capacity in bits per transmission. We are now in position to prove the conversion to the channel coding theorem.

Proof: (conversion to channel coding theorem). We have to show that any sequence of $(2^{nR},n)$, codes with $\lambda^{(n)} \to 0$ must have $R \leq C$.

If the maximal probability of error is close to zero, then the average probability of error sequence of codes also goes to zero, i.e., $\lambda^{(n)} \to 0$ implies $P^{(n)}_e \to 0$, where $P^{(n)}_e$ is defined as average probability of error for an *(M,n)* code: $P^{(n)}_e = \frac{1}{M} \sum_{i=1}^{M} \lambda_i$.For each n, let W be drawn according to a uniform distribution, $P^{(n)}_e = Pr(\hat{W} \neq W)$.

Hence

$$nR = H(W) = H(W \mid Y^n) + I(W;Y^n) \tag{24}$$

$$\leq H(W \mid Y^n) + I(X^n(W);Y^n) \tag{25}$$

$$\leq 1 + P^{(n)}_e nR + I(X^n(W);Y^n)\, P^{(n)}_e \tag{26}$$

$$\leq 1 + P^{(n)}_e nR + nC \tag{27}$$

Dividing by n, we obtain

$$R \leq P^{(n)}_e R + \frac{1}{n} + C \tag{28}$$

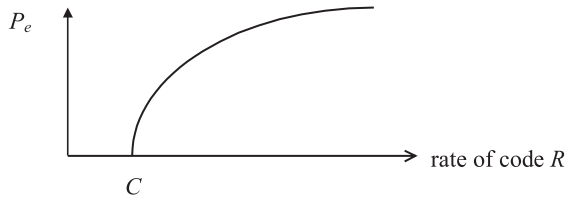
Now letting $n \to \infty$, we can see that the first two terms on the right hand side go to 0, and hence

$$R \leq C. \tag{29}$$

Takže po prepísaní (28) je

$$P^{(n)}_e \geq 1 - \frac{C}{R} - \frac{1}{nR}. \qquad (30)$$

Táto rovnica ukazuje, že ak $R > C$, pravdepodobnosť chyby sa vzdiaľuje od nuly so zväčšujúcim sa n. Preto sa pri takých rýchlostiach nedá dosiahnuť ľubovoľne malá pravdepodobnosť chyby (obr. 3).



Obr. 3 Dolná hranica pravdepodobnosti chyby
Fig. 3 Lower boundary of the error probability

Táto konverzia je slabou konverziou kódovacej teorémy. Dá sa dokázať aj silná konverzia, ktorá tvrdí, že pri zvyšovaní rýchlosti nad hodnotu kapacity sa pravdepodobnosť chyby blíži exponenciálne k 0,5. Hodnota kapacity je hraničným bodom, v ktorom sa pravdepodobnosť chyby mení.

Záver

V závere je formulovaný postup pre opis bezpečnosti z hľadiska pravdepodobnosti nebezpečnej poruchy ako pravdepodobnosti výskytu náhodnej premennej X_{ij}. Pravdepodobnosť $P(X_{ij})$ pritom znamená pravdepodobnosť prechodu objektu zo stavu i do stavu j. Stav X_i patrí do množiny bezpečných stavov, stav X_j patrí do množiny nebezpečných stavov. Pre obidve množiny stavov možno použiť aj jemnejšie delenie, pričom sa použije stromová štruktúra zobrazenia stavov do náhodných premenných.

- Bezpečnosť sa opisuje štruktúrou stavov. Túto štruktúru možno použiť na opis stavov celého objektu, alebo v hierarchickom usporiadaní na opis stavov jednotlivých prvkov , alebo funkcií objektu.
- Pre jednotlivé typy objektov (zabezpečovacích systémov, alebo dopravnej cesty ako celku) sa stanoví druh závislosti stavov, ktoré sú reprezentované náhodnými premennými. V prvom priblížení sa dá predpokladať, že náhodné premenné (stavy objektu) tvoria Markovovu reťaz.
- Pri manipulácii s náhodnými premennými treba rešpektovať fakty, vychádzajúce z nerovnosti pre spracovanie dát.
- Miera bezpečnosti objektu je závislá od pravdepodobnosti chybného odhadu náhodnej premennej X na základe znalosti premennej Y. Využitím chybovej nerovnosti sa dajú stanoviť požiadavky na logickú reprezentáciu zabezpečovacích funkcií.

Recenzenti: P. Peniak, L. Skyva

We can rewrite (28) as

$$P^{(n)}_e \geq 1 - \frac{C}{R} - \frac{1}{nR}. \qquad (30)$$

This equation shows that if $R > C$, the error probability is moving away from 0 for sufficiently large n. Hence we cannot achieve an arbitrarily low error probability at rates above capacity. This inequality is illustrated in Fig. 3.

This conversion is a weak conversion of the channel coding theorem. It is also possible to prove a strong conversion, which states that for the rates above capacity, the error probability nears exponentially to 0,5. Hence, the capacity is a very clear dividing point in which the error probability is changing.

Conclusion

In the conclusion a procedure for description of safety from the point of dangerous error probability as a probability of random variable Xij appearance is formulated. Probability P(Xij) means the probability of transition of the object from state i to state j. State Xi belongs to the set of safe states, state Xj belongs to the set of hazardous states. For both sets of states a more precise division can be used, by which a tree-type structure of displaying states into random variables is used.

- safety is described by the states structure. This structure can be used to describe states of the whole object, alternatively in hierarchical order to describe the state of single elements, or object functions.
- for individual object types (safety system, or transport route as a whole) a state dependency type is determined (of the states represented by random variables). In the first approach it can be assumed, that random variables (object states) create Markov chain.
- when manipulating with random variables, facts concluding from data processing inequality have to be respected.
- the rate of object safety depends on the probability of a wrong estimation of the random variable X based on knowing the variable Y. By using the error inequality the requirements for logical representation of safety functions can be estimated.

The paper was elaborated with the support of grants VEGA 1/5230/98 and 1/5255/98.

Reviewed by: P. Peniak, L. Skyva

https://doi.org/10.26552/com.C.2000.2.30-37

komunikácie
COMMUNICATIONS

Rachid Laalaoua - Tülin Atmaca *

ŠTÚDIUM MECHANIZMU DYNAMICKÉHO RIADENIA FRONTOV PRE ZABEZPEČENIE QoS

A STUDY OF A DYNAMIC SCHEDULING MECHANISM TO GUARANTEE QoS

Riadenie výberu z frontov je jedným z kľúčových mechanizmov, ktorý bude používaný v sieťových prvkoch (prepínače a smerovače) pre podporu aplikácií v reálnom čase v širokopásmových sieťach. Garancia kvality služby (QoS) je dôležitou výzvou pri návrhu paketových sietí s integrovanými službami. Spôsoby výberu z frontov sú integrálnou súčasťou problému a úzko súvisia s ďalšími aspektmi modelovania sietí, akými sú charakteristiky toku a QoS špecifikácia. V tomto článku diskutujeme o dvoch spôsoboch výberu paketov z frontov s použitím priorít: Head of Line (HoL) a preemptívna disciplína - pre ktoré navrhujeme analytický popis a predkladáme numerické výsledky získané pomocou Markovho reťazca. Je navrhnutý nový mechanizmus nazývaný Dynamic-Weighted Fair Queuing (Dynamic-WFQ), v ktorom výber do frontu závisí od triedy prevádzky a dĺžky frontu. Je uvedený príklad funkcie výberu, ktorý je riešený Markovým reťazcom a výsledky sú potvrdené simuláciou. Nakoniec porovnáme hodnotenie priepustnosti pre multimediálnu komunikáciu s použitím popísaného spôsobu výberu z frontov.

Packet scheduling is one of the key mechanisms that will be used in network elements (switches and routers) for supporting real-time applications in broadband networks. Provision of Quality-of-Service (QoS) guarantees is an important and challenging issue in the design of integrated service packet networks. Scheduling disciplines are an integral part of the problem and are closely related to other aspects of network modeling such as traffic charaterization and QoS specification. In this paper we discuss two priority scheduling mechanisms: Head of Line (HoL) and preemptive discipline; for both, we propose an analytical description and present numerical results obtained by Markov chain. A novel mechanism called Dynamic-Weighted Fair Queuing (Dynamic-WFQ) which depends on the Classes of Service and the queue occupancies is proposed. An example of selection function is solved by Markov Chain and the results are validated by simulation. Finally, we compare the performance evaluation of multimedia communication scheduling algorithms described above.

KEYWORDS: Queuing, QoS, Dynamic-WFQ, HoL, Markov Chain, Simulation, Class of Service.

1. Introduction

High speed networks are currently expected to carry a wide range of traffic types, including data, voice, broadcast and interactive video. Each of these traffic types requires a different service from the network. The design of networks that provide a good Quality of Service (QoS) to the large variety of expected users is an open and interesting research area.

Network performance is quite sensitive to the queue service discipline implemented at the output trunks of routers and switches. While most current implementations are of the FCFS type, recent works have shown that other disciplines in which the priorities are taken into account such as HoL, Preemptive discipline, WFQ (Weighted Fair Queuing) provide better performance [1][6].

Traditionally, the FCFS discipline is used at each output port of switches or routers. Also space priorities (threshold and push-out mechanisms) are considered to differ loss probabilities but not the response time between different Classes of Service [11]. With emerging Classes of Service, this discipline must change and different classes should be placed in separate queues with single server. With FCFS discipline, there is no particular treatment given to packets from flows that are of higher priority or that are more delay sensitive. In addition, small packets can be queued behind long packets, then FCFS queuing discipline results in a larger average delay per packet than if the shorter packets were transmitted before the longer packets. In general, flows of larger packets get better service.

Hence, priorities may be assigned to the basis of traffic type. For example, WFQ is based on a *hypothetical fluid-flow* system called the Generalized Processor Sharing (GPS) [12] [13]. In GPS, if there are N non-empty queues, the server treats all of the N queues simultaneously at a rate proportional to their "reserved" weights. GPS is hypothetical because it can serve N queues simultaneously.

WFQ has received a lot of attention in the research community. Parekh's work [16] shows that in the absence of link-sharing, the end-to-end delay bound provided by WFQ [15] [16], which is the standard packet approximation algorithm of GPS, is very close to that provided by GPS. While WFQ maintains the bounded delay property of GPS, its fairness property is much weaker than GPS.

* **Rachid Laalaoua, PhD Student, Tülin Atmaca, Professor**
Institut National des Télécommunications, 9 rue Charles Fourier 91011 Evry Cedex, France,
Tel.: +33-1-60764742, Fax: +33-1-60764780, E-mail: rachid.laalaoua@int-evry.fr, tulin.atmaca@int-evry.fr

As described in [14], WFQ can introduce substantial inaccuracy in GPS approximation. This inaccuracy significantly affects best-effort traffic management, real-time traffic management, and link-sharing algorithms.

WFQ, also known as the Packet-by-Packet Generalized Processor Sharing, is the most weel-known packet approximation algorithm for GPS. In WFQ, when the server is ready to transmit the next packet at time τ, it picks, among all the packets queued in the system at τ, the first packet that would complete service in the corresponding GPS system if no additional packets arrive after τ.

A queuing discipline is nothing more than a means for choosing which packet in the different queues (multi-queues) will be served next [10] . This decision may be based on one or all of the following criteria:
- different Classes of Service.
- queue occupancies.
- lenght of different packets.

In our work a three Class-of-Service (CoS) system is considered with various traffic types. A network node is modelled by three-queue system, each of which is dedicated to a set of traffic type. Most of the traffic, particulary best effort, remains bursty. Therefore, the aggregation of Markov Modulated sources used in this study attemps to generate realistic burstiness properties.

In this paper, we introduce a new scheduling algorithm that depends on the both first points.

Analytical models of priority queues, both preemptive and nonpreemptive are well developped [18][19], but are restricted to infinite queues. In the litterature, the performance of a switch or router that schedules a mixture of real-time, non-real-time and best-effort traffic is studied. The performance measurement of interest for real-time traffic is average delay, and for non-real-time traffic is loss probability.

In this paper, we consider two models. The first is described in section 2 and is represented by three queues with a single server. Three service algorithms are presented: HoL, preemptive discipline and dynamic-WFQ. Let us note that the results are obtained both using simulations and an analytical method based on Markov chain approach, for three disciplines mentioned above. In this section, a second model is presented too to compare the performance obtained by three mechanisms mentioned above (HoL, WFQ and Dynamic-WFQ). However the traffic characteristics are changed as well as the number of sources. In section 3, a complete description is given for this model. So, we present only the simulations results for this model. Finally, some concluding remarks are given in section 4.

2 Priority discipline

2.1 Model Description

The system model is depicted in Figure 1. We consider three queues served by a single server. We assume that the arrival streams are independent Poisson processes with intensity λ_i at queue i (i = 1, 2, 3). The service time is exponentially distributed with parameter μ.

This model will be studied with three service disciplines (HoL, Preemptive and Dynamic-WFQ).

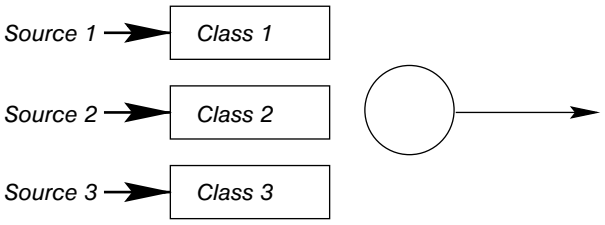


Fig. 1 Model to study

2.2 Head-of-Line (HoL)

In this scheduling policy, priority is always given to Class 1 traffic (real-time). Class 2 traffic is served only if there are no queued Class 1 packets. The last Class is served when both queues (queue 1 and queue 2) are empty. A service is not preempted. Within a class, packets are served with FCFS discipline.

2.2.1 Markovian Model

Markov chains are known to be powerful modelling tool for a variety of practical situations. For HoL discipline with three queues, the behaviour of the system described above is represented by a Markov Chain.

The state vector of this Markov chain is given by $n = (n_1, n_2, n_3, F)$ where n_i is the number of customers at queue i, and Flag F takes four values as follows:

$$F = \begin{cases} 0 \text{ if } n_1 = n_2 = n_3 = 0 \\ 1 \text{ if } n_1 > 0 \\ 2 \text{ if } n_1 = 0 \text{ and } n_2 > 0 \\ 3 \text{ if } n_1 = n_2 = 0 \text{ and } n_3 > 0 \end{cases}$$

The decision of the server depends on the value of the Flag. So the following algorithm shows how the transitions take place:

```
1  If (queue1 is not empty)
2          take customer from queue 1
3  else
4          if (queue2 is not empty)
5                  take customer from queue 2
6  else
7                  if (queue3 is not empty)
8                          take customer from queue 3
9  end
```

After generating Q matrix, we compute the steady state probabilities using Arnoldi's method [5].

we have:

$$\begin{cases} \pi Q = 0 \\ \Sigma \pi_j = 1 \end{cases}$$

$\pi_j = P(n = j)$ is the probability that the present state is j and the state vector of these probabilities is $\pi = (\pi_0; \pi_1; \ldots \pi_N)$.
Let $j = (n_1; n_2; n_3; F)$ be a state of Markov chain, and the probability that there is x customers in queue i is given as follows:

queue 1: $P^1(x) = \Sigma \pi_j$ such that $n_1 = x$.
queue 2: $P^2(x) = \Sigma \pi_j$ such that $n_2 = x$.
queue 3: $P^3(x) = \Sigma \pi_j$ such that $n_3 = x$.

Note that the probability that there is x customers in queue i is $P^i(x)$.

2.2.2 Results

This section gives the simulation and analytical results concerning the queue length distribution (probability density function (pdf)) and the loss probability. These parameters depend on the buffer capacity and on the distribution of global traffic. The results are obtained for $\lambda_1 = 0.27$ *packets/unit time* $\lambda_2 = 0.27$ *packets/unit time*; $\lambda_3 = 0.26$ *packets/unit time.* Simulation model is writen in C language programing.

Figure 2 shows that Markovian results are close to those obtained by simulation.

It is clear that HoL gives good performance for Class 1.

Figure 3 presents the loss probabilities vs the proportion of traffic types obtained by simulation and by analytical method for real-time and non-real-time traffics. It appears that the loss probability increases when the Class 1 load increases too.

The second curve in Figure 3 shows the loss probabilty of Class 2 as a function of traffic load (Class1 and Class 2). We note that the Class 1 has an impact on the Class 2.

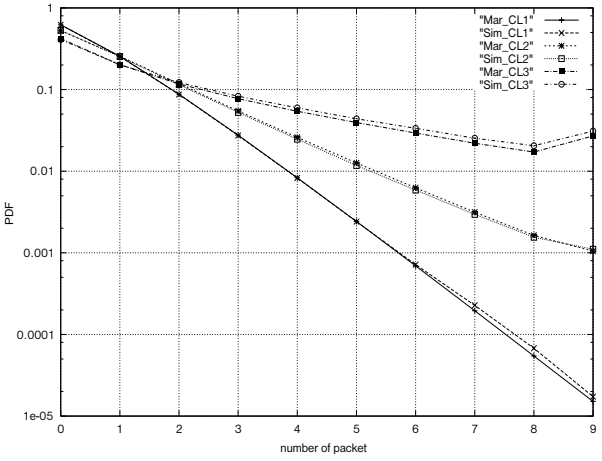


Fig. 2 Queue length distribution (HoL)

2.3 Preemptive discipline

Preemptive discipline gives absolute high priority to traffic of Class 1, because the traffic which has a high priority preemptes the customer with low priority in service. Therefore, in HoL discipline, they are served after the customer having lower priority has finished his service. For preemptive discipline, three cases are usually identified:

- *Preemptive resume:* customer picks up from where he left off.
- *Preemptive repeat without resampling:* when customer reenters service after having been preempted, he starts with the same total service time that he has lost previously.
- *Preemptive repeat with resampling:* this case assumes that a new service time is chosen for our reentering customer.

In this study, Preemptive resume is considered. In context of integrated service in Frame Relay, some constructor uses this scheme in the switches. The head of sub-frames that mother frame is preempted, are calculated automatically.

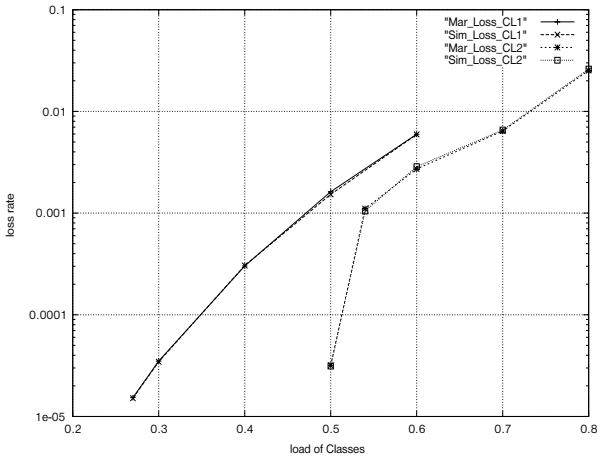


Fig. 3 Loss probabilty as a function of traffic load (HoL)

2.3.1 Markovian Model

In this section, we study the model presented in Figure 1 under Preemptive discipline using Markov chain approach. The state vector is given by

$n = (n_1, n_2, n_3, f_1, f_2, f_3)$ where n_i is the number of customers in queue i, and the value of $f_i \in \{0, 1\}$ select which customer will be served.

Transition rates are given in Figure 4. We compute steady state probabilities of this Markov chain using Arnoldi's method.

2.3.2 Results

We note that the results obtained by solving of Markov chain are close to those obtained by simulation. These results are shown in Figure 5, and are obtained for the same parameters used previ-

ously. We note also that Class 1 has greater loss when the HoL discipline are used because we can not preempt the service of lower priority customers. Therefore, loss probability of Class 2 depends on the proportion of each traffic type. Then, when proportion of Class 1 or Class 2 traffics increases, losses with preemptive discipline will be smaller than those with HoL discipline. Table 1 shows that the sejourn time increases when the proportion of traffic corresponding increases.

The results obtained using the simulation and an analytical method (HoL and Preemptive), have 1 % relative error.

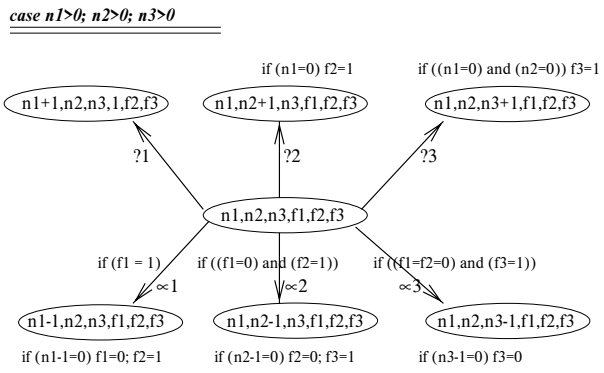


Fig. 4 Possible transitions

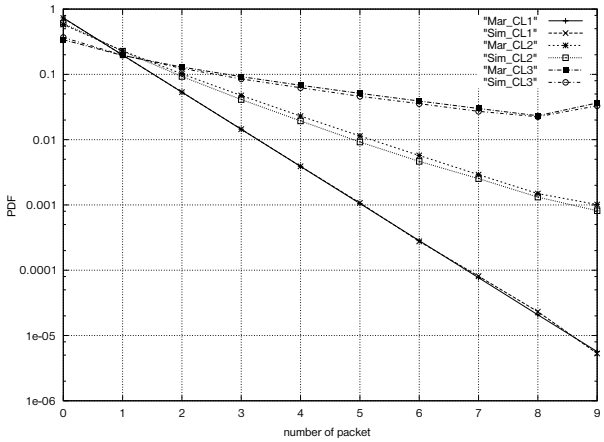


Fig. 5 Queue length distribution (Preemptive discipline)

Sejourn Time (second) as a function of traffic proportion Tab. 1

λ_1		0.27	0.3	0.4	0.5	0.6
λ_2		0.27	0.3	0.1	0.2	0.2
λ_3		0.26	0.2	0.3	0.1	0.1
Cl 1	Pr	1.37	1.43	1.67	1.99	2.41
	HoL	2.08	2.13	2.31	2.57	3.09
Cl 2	Pr	2.68	3.19	2.88	5.76	9.45
	HoL	3.25	3.66	3.57	5.79	9.58
Cl 3	Pr	8.18	9.63	7.41	13.54	27.81
	HoL	7.84	9.09	7.16	12.53	26.01

2.4 Dynamic-WFQ

From the discussion in the previous sections, it is seen clearly that the adapting scheduling algorithm based on queue occupancies and using weight for different Classes of Service can be benifical to network performance. Here, we present Dynamic-WFQ discipline in which we introduce the selection function which plays an important role for the packet transmission scheduling. This function computes a number of packets from different queues that will be transmitted to virtual queue in the next cycle (Figure 6).

A cycle can be determined as the total time spent to transmit all packets in the virtual queue. The behaviour of this function depends on two parameters (queue occupancies and priorities). When these parameters increase, the function must increase too. The selection function $f(c_i)$ is given as follows:

$$f(c_i) = w * oc_i + (1 - w) * pr_i$$

$$m_i = \left[QSize * \frac{f(c_i)}{\sum_{i=1}^{i=3} f(c_i)} \right]$$

such that $0 \leqslant (w; oc_i; pr_i) \leqslant 1$ and $\sum_{i=1}^{i=3} pr_i = 1$.

where m_i is a packet number of Class i, $QSize$ is the total packets that will be transmitted next in the virtual queue and oc_i is queue i occupancy.

In this study, the parameters w; pr_i are taken in the following values. $w = 0.4$; $pr_1 = 0.55$; $pr_2 = 0.35$; $pr_3 = 0.1$.

The packets already in the separate queues wait to be chosen to transmit (by the selection function) later. This service scheduling can be seen as cyclic service with priority discipline, but the packet numbers to transmit from each queue can be changed in each cycle. Packet numbers are calculated by the selection function.

2.4.1 Markovian Model

The state vector of Markov chain of Dynamic-WFQ is given by $(n_1, n_2, n_3, x_1, x_2, x_3)$ where n_i is the number of customers in queue i, and x_i is the number of class i customers in virtual queue. The selection function starts when $\sum_{i=1}^{i=3} x_i = 1$. This means that virtual queue contains a single customer. So the selection function must compute the number of packets from each seperate queues and transit with μ rate to the next state:

$(Max(n_1 - m_1, 0), Max(n_2 - m_2, 0), Max(n_3 - m_3, 0), Min(n_1, m_1), Min(n_2, m_2), Min(n_3, m_3))$

When all queues are empties, the first packet that arrives, transits directly to virtual queue.

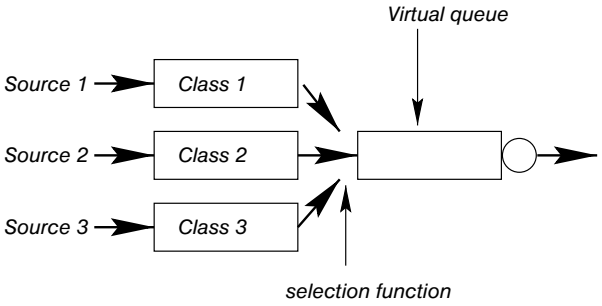


Figure 6: Dynamic-WFQ Model

2.4.2 Results

Given the number of states of related Markov chain increasing as function of buffer capacities, the computation time of the steady state probabilities becomes considerably long.

Therefore, we have reduced buffer sizes to $N_1 = N_2 = N_3 = 9$ in order to show that Markov results are close to simulation results. These results are shown in Figure 7.

In order to show the real performance of Dynamic-WFQ algorithm, we compare queue length distribution intended in Figure 8 of HoL and Dynamic-WFQ algorithms. We observe that the buffer occupancies for class 1 (higher priority) is longer in the case of Dynamic-WFQ scheduling than HoL scheduling. On the contrary, for both classes with lower priorities, Dynamic-WFQ provide better performance.

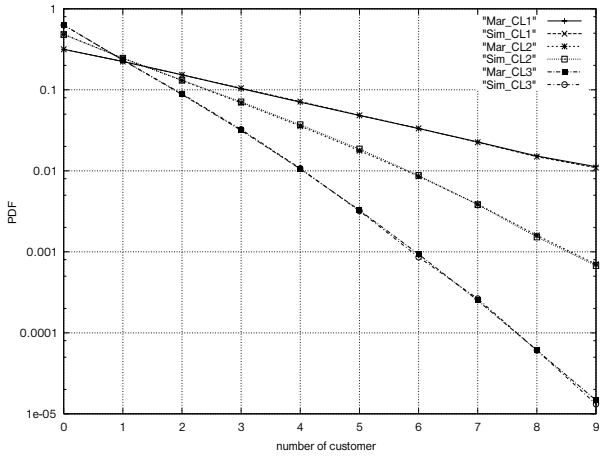


Fig. 7 Queue length distribution (Dynamic-WFQ)

3 Performance Comparison of HoL, WFQ and Dynamic-WFQ

3.1 Model

In multimedia communication, multiple data streams need to be multiplexed on a single transmission channel. Multiplexing of data streams using such simple mechanism may have undesirable results for multimedia real-time traffic. In this section, we analyze and compare three mechanisms with priority.

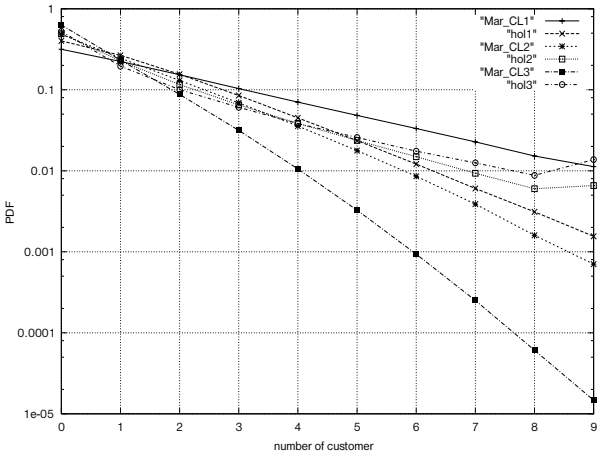


Fig. 8 Queue length distribution (HoL, Dynamic-WFQ)

The assumptions considered for the model under study aim to represent the context of a high speed Wide Area Network (WAN). This implies to take into account non negligeable transmission times between the sources, the bottleneck and the destinations.

We consider that in such backbone, the connectivity is reduced, thus we have modelled three groups of numerous sources whose traffics go through three high speed links (622 Mbit/s). Each set of sources includes numerous sources of different types of traffic (voice, videoconference, video, data with QoS, data best effort). The traffic profiles are disrupted by background traffics which transported in the same links but have other destinations.

This configuration depicted in Figure 9, is intended to be a test for the comparison between the three service scheduling disciplines described above, in the presence of different traffic types and different number of sources.

3.2 Traffic Characterization

We have three sets of sources which represent different traffic types and background flows that represent $\frac{2}{3}$ of global traffic. The drawback of this method is that it requires a realistic network simulator, and considerable amount of computing time. However, we approximate the set of background flows by using a server with vacation in multiplexer for each set of sources. Since the parasit proportion is $\frac{2}{3}$ of all traffic, the idle time of server is $\frac{0.8*2}{3} + 0.2$, and working time is $\frac{0.8*1}{3}$ (0.8 is load). These periods have an exponential distribution. Figure 10 shows the probability density function of the number of packets in the multiplexer using real model with classical server, and other method with vacation server using different time. We note that 0.01 second, which represents the sum of idle time and working time, is the best approximation. *idle time* $= 0.00267$ *and working time* $= 0.00733$.

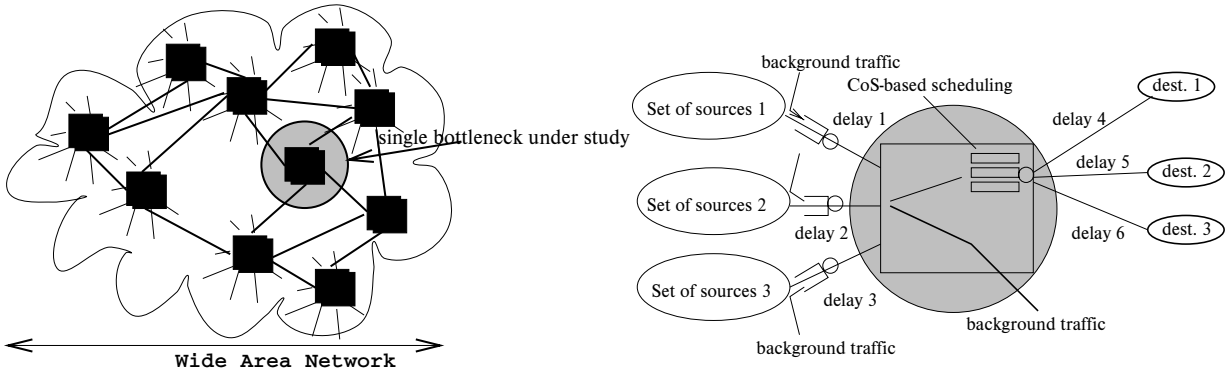


Fig. 9 WAN topology and single bottleneck model

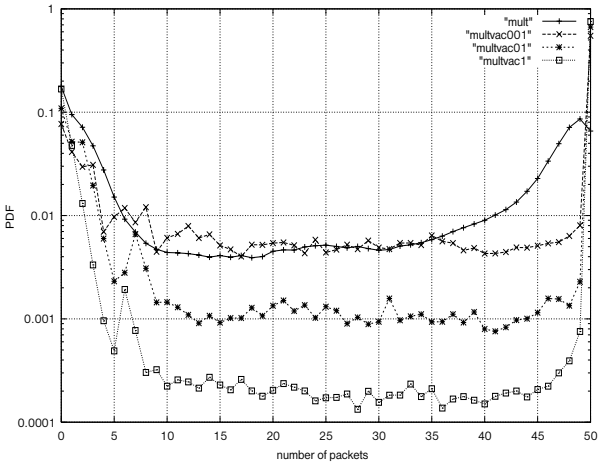


Figure 10: Queue length distribution at Multiplexer

In order to understand the effect of different mechanisms, we study a scenario (Figure 9) using a simulator written in C language. As shown in Figure 9, the simulated network consists of three multiplexers with vacation server which are linked to a router. The router is modeled by three queues with a single server using a priority scheduling mechanisms. The following table shows the parameters used in the model.

It appears that the end-to-end delay for voice traffic using Dynamic-WFQ mechanism is near to delay obtained by WFQ and HoL mechanisms. We remark that the peaks corresponding to 20 s in Figure 11, are due to the bursty arrivals. In all case, the end-to-end delay is less than 11 ms. The parameters which are taken in this section are:
$pr_1 = 0.6$, $pr_2 = 0.3$, $pr_3 = 0.1$ and $w = 0.35$.

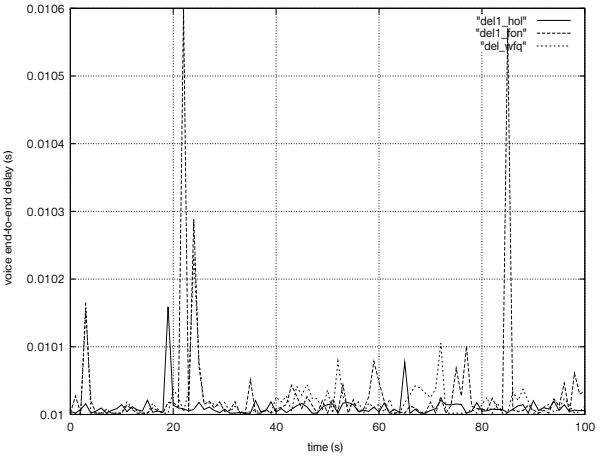


Figure 11: Voice End-to-end delay

TYPE	Model	number of sources per set	Average rate per source	% of global Set traffic	Packet size (bytes)
Voice	IDP	324	32 kbit/s	5 %	160
Videoconference	MMDP	50	256 kbit/s	9 %	256
Video (MPEG2)	MMDP	6	5 Mbit/s	16 %	512
Data with QoS	ON/OFF	3	9 Mbit/s	16 %	512
Data Best effort	ON/OFF	12	8 Mbit/s	51 %	1536

3.3 Results

This section gives the simulation results as the queue length distribution and voice end-to-end delay.

On the other hand, Figures 12 and 13 show the comparison of queue 2 and 3 packet distribution using different scheduling mechanisms. In Figure 12 we note that the line graphes which represent the pdf of visioconference traffic using HoL, WFQ and

Dynamic-WFQ, have the similair behaviour. Figure 13 shows the pdf of class 2 and Best-effort traffics, using Dynamic-WFQ and the results are better than those using HoL or WFQ disciplines.

4. Conclusion

In our studies we observed that the priority queuing mechanisms cannot isolate the impact of load between traffic streams. The basic property of assigning the bandwidth first to the high priority traffics help to maintain short delay and delay variance for high priority queues.

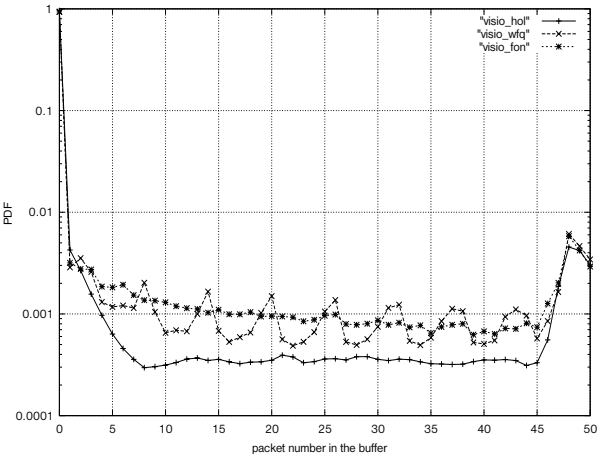


Fig. 12 Visioconference traffic PDF

With WFQ, the absolute weight are given for different traffic types. When congestion occurs for no real-time traffic, it remains for long time because the real-time traffic have a higher weight. The basic idea in Dynamic-WFQ is to change weight each cycle.

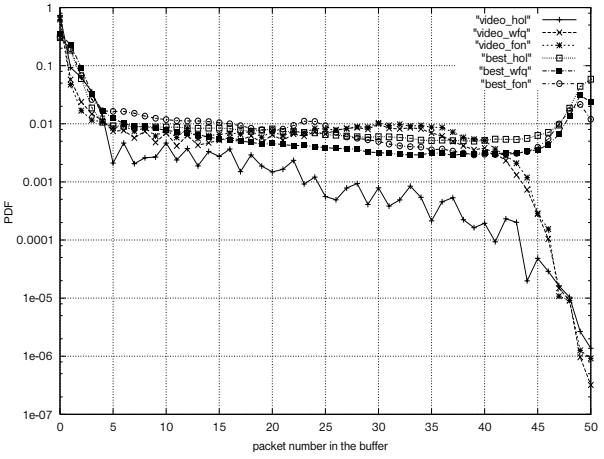


Fig. 13 Video and best-effort traffics PDF

Then, the absolute weight is given to each class (w_i * pr_i), and the remain proportion of bandwidth is shared. This sharing depends on queue occupancies. The congestion level in nodes (for no real-time traffics) is reduced.

As described in [6], FCFS scheduling mechanism which is used in most of the transport protocols today is not suitable for supporting the multimedia data streams. The real-time requirements as audio and video traffic, cannot be supported well in FCFS. It is shown in [14] that the inaccuracy introduced by WFQ can (a) signifcantly increases the delay bound for real-time sessions under hierarchical link-sharing; (b) cause end-to-end feedback algorithms for best-effort traffic to oscillate.

As shown above, Dynamic-WFQ using a simple linear function is performant. In future work, we will focus on an analytical method to obtain the better parameters such a w and pr_i.

Reviewed by: I. Baroňak, M. Klimo

Literatúra - References

[1] SPOHN, D.L.: second edition 1997. Data Network Design, McGraw-Hill.
[2] KLEINROCK, L. 1975. Queueing Systems, Volume 1 and 2, Wiley-Interscience.
[3] PERROS, H. G.: 1994. Queueing Networks with Blocking, Oxford.
[4] GROSS, D., Harris, C. M.: 1985. Fundamentals of Queueing Theory, John Wiley.
[5] RUEGG, A.: 1980. STOCHASTIC PROCESSES, John Wiley & Sons.
[6] JIM, M. Ng., WOODWARD, M. C.: 1998. "Evaluation of multimedia Communication Scheduling Algorithms for Host-based Multiplexing", In Proceeding of the 1998 ICT Conference (Porto-Carras Greece).
[7] GALLARDO, J. R., MAKRAKIS, D., L'OPEZ, M., OROZCO BARBOSA, L.: 1998 "Performnance Comparison of UT and CT under Realistic Traffic Models", In Proceeding of the 1998 ICT Conference (Porto-Carras Greece).
[8] TANG, H., SIMULA, O.: 1998 "Another Dimension of Flow Control for the intelligent node", In Proceeding of the 1998 ICT Conference (Porto-Carras Greece).
[9] SHAH-HEYDARI, S., LE-NGOC, T.: 1998 "Multiple-state MMPP Models for Multimedia ATM Traffic", In Proceeding of the 1998 ICT Conference (Porto-Carras Greece).
[10] LAALAOUA, R., ATMACA, T.: 1998 "Voice Over Frame Relay", In Proceeding of the 1998 ICT Conference (Porto-Carras Greece).
[11] HAMMA, S., PECKA, P., CZACHRSKI, T., Atmaca, T.: 1997 "Markovian analysis of threshold based priority mechanism for Frame-Relay networks", In Proceeding of Voice, Video, and Data Communications Conference on Performance and Control of Network Systems, Switching and Traffic Management in High Speed Networks, (Dallas,Texas, USA).

[12] PAREKH, A., GALLAGER, R G.: 1993. "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case" IEEE/ACM Transactions on Networking, Vol. 1, No. 3, June 1993.
[13] PAREKH, A., GALLAGER, R G.: 1994. "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case" IEEE/ACM Transactions on Networking, Vol. 2, No. 2, April 1994.
[14] ZHANG, Bennet, H.: 1990. "Why WFQ is not good enough for Integrated Services Networks", In Proceeding of NOSSDAV'96 Conference (April 1996).
[15] DEMERS, A., KESHAV, S., SHENKER, S.: 1996. "Analysis and Simulation of a fair queueing algorithm", Journal of Internetworking Research anf Experiece, pages 3-26, October 1990.
[16] PAREKH, A.: 1992. "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks" PhD dissertation, Massachusetts Institute of Technology, February 1992.
[17] ROBERTS, J. W.: 1994 "Virtual Spacing For Flexible Traffic Control", International Journal of Communication Systems, VOL. 7, 307-318.
[18] JAISWAL, K. N.: 1968 "Priority Queues", Academic Press, NewYork.
[19] LAVENBERG, S. S.: 1983 "Computer Performance Modelling Handbook", Academic Press, NewYork.

https://doi.org/10.26552/com.C.2000.2.38-55

komunikácie
COMMUNICATIONS

Peter Géczy - Shiro Usui *

INTELIGENTNÉ ADAPTABILNÉ SYSTEMY: PRÍSTUP PRVÉHO RÁDU

INTELLIGENT ADAPTABLE SYSTEMS: FIRST ORDER APPROACH

Budúcnosť komunikácií nevyhnutne vyžaduje technológie umožňujúce vysokú úroveň flexibility, adaptability a inteligencie. Inteligentné adaptabilné systémy sú obzvlášť vhodné pre túto úlohu. Väčšina adaptabilných systémov je založená na neurónových sieťach. Umelé neurónové siete sú systémy s enormnou interkonektivitou. Neurónové siete sa neprogramujú. Sú schopné nadobudnúť významné vlastnosti vďaka adaptačnému procesu nazývanému učenie. Predkladaný inteligentný adaptabilný systém využíva technológiu neurónových sietí. Systém umožňuje internú viacúrovňovú adaptabilitu. Vzhľadom na dostupné dáta autonómne adaptuje svoje parametre a štruktúru. Z externej perspektívy disponuje kontrolou vlastného vstupno-výstupného interfejsu. Systém je schopný vybrať si vhodné učiace exempláre z dostupného množstva dát tak, aby dosahoval optimálny progres učenia. Po naučení systém disponuje možnosťou výstupu v logickom formáte. Uvedený inteligentný adaptabilný systém pozostáva z niekoľkých modulov. Princíp a funkcia každého z modulov sú popísané a ilustratívne demonštrované.

Future of communications inevitably calls for technologies that feature high-level flexibility, adaptability, and intelligence. Intelligent adaptable systems are particularly suitable for this mission. Majority of adaptable systems utilize neural networks. Artificial neural networks are systems with huge network-like interconnectivity. They are not programmed. Neural Networks gain valuable properties through the process of adaptation called learning. Presented intelligent adaptable system utilizes neural network technology. The system incorporates internal adaptability at several levels. It autonomously adapts its parameters and structure to the presented data. Externally, it appropriately manages its input-output interfaces. The system is able to select suitable training exemplars from the available amount of data in order to achieve the optimal learning performance. After training the system provides logical output format of the task. Introduced intelligent adaptable system consists of several modules. Principle and functionality of each module is described and illustratively demonstrated.

1. Introduction

Rapid expansion of communication technologies is widely influencing our everyday life. Userbase of communication services grows at such a rate that technology development constantly faces challenges in order to sustain stability and reliability of services. Such state of the communications sector has led to the need of developing new techniques to enhance speed, expand bandwidth, and provide higher security, to mention a few.

Neural networks, as a result of their inherent learning and adaptive qualities, have enormous applicability in this explosive area of technology. Articial neural networks represent the technical abstraction of the biological structures observable in the nervous system of living creatures. The nervous system of biological entities consists of the elementary processing blocks-neurons. Neurons are interconnected by synapses and axons, enabling propagation of bio-signals and creation of bio-information pathways. Huge interconnectivity allows formation of wide networks with massive parallel processing capabilities. Just as neural networks, the com-munication systems are composed of units that process signal/information transmitted over cables or ether using well-defined protocols such as TCP/IP, ISDN, CDMA, TDMA, etc. This apparent parallel uncovers enormous potential of neural networks in communication technologies of future. It positions the neural networks into a place where a considerable advantage of their adaptability (and other properties) can be taken.

At present the communication technologies could be viewed as algorithm-oriented. Switching, routing, and packet transmitting is controlled by algorithms that feature none or very little adaptability. This trend has originated from the classical information science established in the early 50's by Shannon [1] and has further been strengthened by development of computer science. The nodes in communication infrastructure, though they are highly so-phisticated, automated, and computerized systems, utilize the algorithmic concepts lacking higher-order auto-adaptability to external and/or internal conditions. Lack of adaptability results in lower effectiveness of the global communication infrastructure as well as its elementary units.

The future of communication technologies inevitably calls for higher adaptability and/or learning, flexibility, and "intelligence". With the current advancement of neural networks all these qualities can be achieved using conventional computer systems without substantial investment in rebuilding the existing physical communication infrastructure. This economical solution is likely to determine the future course of communication technologies.

* **Peter Géczy, Shiro Usui**
Future Technology Research Center, Toyohashi University of Technology, Hibarigaoka, Tempaku-cho, Toyohashi 441-8580, Japan

Novel technologies will impact the communication sector in two major spheres: global and local. Globally influential future communication technologies are required in areas such as network management & control, resource allocation, market prediction, security, reliability, fault tolerance, and datamining. Intelligent adaptable systems (IAS) can be applied directly (or indirectly) to all these domains. In the local domain, IAS have already been applied to routing algorithms (classical traveling salesman problem), voice, image & character recognition, intelligent search & information filtering, and access control [2]. However, the already wide spectrum of IAS applicability is only the initial stage. Further expansion and progress of communication technologies and services will uncover numerous other areas where IAS will be the preferred technological choice.

2. Elements of Intelligent Adaptable Systems

Intelligent adaptable systems draw on the parallel of brain-style information processing. All the changes in the adaptivity of the brain as well as stimuli selectivity occur simultaneously. At the same time the brain is capable of processing incoming information from the receptors, producing adequate responses, and yet adapting itself synaptically and structurally. Although the processing speed of neurons is relatively slow (in the range of milliseconds) the massive parallelism in the brain sufficiently subsidizes this ineficiency. Due to the massive parallelism, the brain is continuously able to process an enormous amount of information.

Block structure of brain-like IAS is depicted in Fig. 1. The system manages its input interface by selecting appropriately the most suitable exemplars for learning. The central part of the system is an artificial neural network. The artificial neural network consists of elementary computational units-artificial neurons-interconnected by real valued weight connections. The artificial neural network should be adaptable both at the microstructure level (connection adaptation) and at the macrostructure level (structural adaptation). The concept displayed in Fig. 1, however, does not end only with the dynamic adaptability and exemplar selectivity. It represents a progressive step forward-towards the logical representation of knowledge that a network gains by training. This is the task of the knowledge acquisition module. The knowledge acquisition module should be able to extract knowledge that the network acquires during the learning. It is also important to note that the subsystems should dynamically co-operate and simultaneously operate on the artificial neural network during the process of learning in order to achieve satisfactory performance on the trained task.

Sample selection is the task of selecting suitable training exemplars from a training data set with the aim of improving the quality of training. Early approaches to sample selection have focused only on determining the informatively sucient subset of training set which was thereafter fixed and used for training. An informatively sufficient sample set was determined based on either the worst case analysis (VC dimensions) or the average case analysis (Bayesian statistics, information theory) [3]. More recent approaches, particularly those linked with on-line training, have

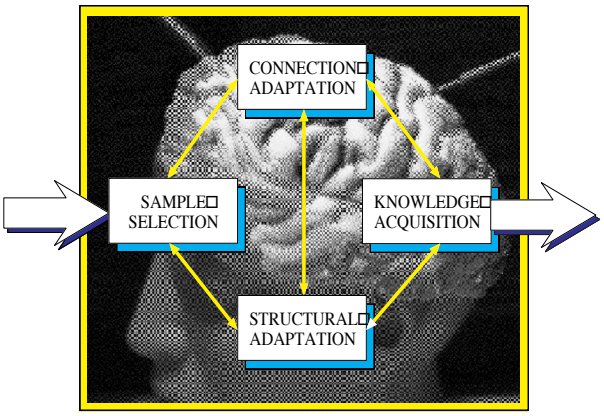


Fig. 1 Block scheme of brain-like IAS. Each module represents a specific underlying feature of learning. A knowledge acquisition module enables the extraction of knowledge from an artificial neural network trained on a specific task in the form of logical rules. All the modules should dynamically and simultaneously co-operate.

focused on determining the sample distribution with respect to which they actively sample data [4] – [8].

None of the aforementioned approaches are used in this study. The dynamic sample selection technique, presented in this study, does not impose any restrictions on the sample distribution and neither on the selected sample size. The number of the selected exemplars is allowed to vary at each iteration. Also the particular training set selected at one iteration may completely differ from the one selected at another iteration. The artificial neural network is given the freedom to select the exemplar set suitable for the fastest progress at each iteration.

The presented dynamic sample selection is capable of selecting an appropriate exemplar set that may vary in size and in the selected samples at each iteration of training. It is based on the controlled normed expression of error gradient that largely determines the search direction of the optimization technique. The error gradient is formed of gradients for each sample presented to a network. Thus the selection of training exemplars also plays a role in determining the search direction of an optimization technique. An appropriately determined search direction formed of suitably selected exemplars at each iteration may increase the convergence speed of optimization and hence also training.

Conection adaptation in neural networks is seen as an optimization task. The objective is to minimize the discrepancy between a network mapping and some true mapping with respect to the measure denoted as the objective function. True mapping is usually not available in its entirety. Having complete information about the true mapping would make the network's training meaningless unless a benchmark evaluation is under consideration. However, even in such cases it is preferred to use benchmark data sets of well-known real-world problems. True mapping is thus represented in its incomplete form, that is, in a form of a finite number of samples.

Relevance of the optimization field [9] in neural network training gives rise to the use of several optimization methods. In

this study preference is given to first order line search optimization techniques [10] – [19], due to their relative computational inexpensiveness, yet reasonable convergence speed. Second order optimization techniques may reach faster convergence rates at the expense of a larger number of calculations. They usually require second order information, that is, the Hessian matrix or its approximation [20] – [23]. Calculation of the Hessian matrix or its approximation becomes pointless for large data sets and/or large network structures due to unbearable computational expensiveness and memory requirements. The fastest achievable convergence rates of first order line search techniques are superlinear convergence rates. Computational complexities of first order approaches are linear, which is one order lower than that of second order techniques. First order approaches have also lower memory requirements. To be specific, the steepest descent methods have no memory requirements and conjugate gradient techniques have linear memory requirements.

Structural adaptation in neural networks aims at optimizing the structure and yet resulting in a network with the required properties. Structural optimality may have a positive impact on generalization property of neural networks. Non-optimality of the structure may cause several complications during and also after training. Underdetermined network structures are incapable of satisfactorily performing the task given by the training set. Overdetermined networks normally display unwanted overfitting properties after training. These and other complications can be avoided by properly fitting the structure of a neural network as well as its parameters.

Former approaches to the structural adaptation of neural networks were based either on regularization or smoothing. The application of regularization techniques results in the modification of the objective function with respect to which the network's parameters are adapted [24], [25]. The objective function then contains a penalizing term for weight connections. The penalizing term leads to connection value-spread. Some connections are forced to take higher values while others lower. Connections having low values are classified as irrelevant and thus eliminated from a structure. Connections with low values are statically irrelevant for mapping performed by a neural network, however, they may have a dynamic relevance during training which may help the network to progress faster. Smoothing approaches are essentially nondynamic [26] – [28]. The network undergoes structural adaptation after training. Curvature of the error surface is evaluated after training and the structural elements of a network corresponding to low curvature values are then removed.

The approach to structural adaptation presented in this study is based on dynamic performance measures [29] – [31]. Performance measures monitor the combined dynamic and static performance of a network up to its particular structural elements. They also detect disturbing features of error surface. Controlling the performance of a network by eliminating the low-performance structural elements and increasing the performance of a network by adding new structural elements represents a new concept for dynamic structural adaptation of neural networks.

Knowledge acquisition delineates a step towards logical representation of a network's "knowledge" gained by training. Since artificial neural networks outline an abstraction of brain structures, it has been a concern of researchers to identify how a network's gained knowledge can be extracted. Distributed representation of knowledge in neural networks makes it a difficult task. Artificial neural network structures do not correspond to any logical representation of knowledge known to humans. A particularly suitable representation of knowledge would be that of logical rules. Such an achievement may have wide impact not only on the design of a new generation of knowledge based systems, but also on our understanding of knowledge representation in the brain.

Currently available approaches to knowledge extraction from artificial neural networks can be viewed from the perspective of rule types and/or training approaches to neural networks predetermined to rule extraction [32] – [35]. The type of rules leads to the distinction of crisp from fuzzy rule extraction techniques. Training approach mainly separates the rule extraction techniques using structure adaptable training and static training. All of the aforementioned approaches, however, strongly precondition the training or the structure of a neural network for further rule extraction tasks. This means, for example, that a neural network may contain nodes representing logical functions or fuzzy membership functions, the structure of the network may be predetermined according to the intended rules to be extracted, a priori knowledge about the task can be implemented into artificial neural networks, etc. All of these techniques reduce the complexity of the principal problem of rule extraction from artificial neural networks.

This study approaches the rule extraction issue in its principal form, that is, independent of the training strategy for neural networks and without preconditioning either the network's structure or processing elements. It allows the neural network to learn freely the task given by the training set. Once the network achieved satisfactory performance in that it maps training data sufficiently correctly, the rule extraction method is applied to transform the network's knowledge into the form of logical rules. The rule extraction method is derived only on the basis of a network mapping and is independent of connection adaptation, sample selection, and structural adaptation [36] – [39].

3. Foundations of the Approach and Notation

Multilayer artificial neural networks, or multilayer perceptrons, are the primary interest in this study. Essentially, a multilayer network is a network which has one input layer, one output layer, and one or more hidden layers. The number of hidden layers can theoretically be unlimited. However, practically they do not exceed several layers. Although, for some application oriented purposes, one can find multilayer perceptron networks with more than one hidden layer, theoretically it has been proven that only one hidden layer is sufficient for universal approximation capabilities of neural networks [40], [41]. This means that an arbitrary functional dependency can be approximated to an arbitrary level of accuracy by a three-layer artificial neural network with an appropriate number of hidden units. Hence the focus on three-layer perceptron networks with the following structure (see Fig. 2).

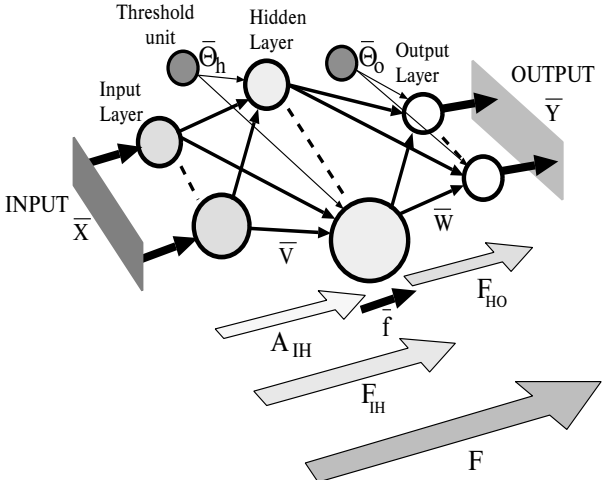


Fig. 2 A model of a three-layer artificial neural network with a mapping scheme. The network contains input, hidden, and output layers. The input layer distributes input signals into the hidden units. The hidden units are nonlinear elements with sigmoidal nonlinear transfer functions. The output units have linear transfer functions. The network's mapping F is decomposed into the input-to-hidden submapping F_{IH} and the hidden-to-output submapping F_{HO} ($F = F_{HO} \circ F_{IH}$). The input-to-hidden submapping F_{IH} ($F_{IH} = \bar{f} \circ A_{IH}$) is further decomposed into the affine input-to-hidden submapping A_{IH} and the nonlinear transformation $\bar{f}$

Mapping of a Three-Layer MLP Network

A mapping F is said to be a mapping of a three-layer *MLP* network defined as follows,

$$F = F_{HO} \circ F_{IH} \; (F : \Re^{N_I} \rightarrow \Re^{N_O}),$$

where N_I is the dimensionality of the input space and N_O is the dimensionality of the output space. F_{HO} is an affine mapping from N_H-dimensional subspace D^{N_H} of $\Re^{N_H}$ to $\Re^{N_O}$.

$$F_{HO} = (F_{HO_1}, \ldots, F_{HO_{N_O}}), F_{HO} : D^{N_H} \rightarrow \Re^{N_O}$$

$$F^{(p)}_{HO_k} = \sum_{j=1}^{N_H} w_{jk} F^{(p)}_{IH_j} - \Theta_{O_k}$$

where $F^{(p)}_{IH_k}$ is the output of the k-*th* hidden unit for the p-*th* training pattern, Θ_{O_k} is the threshold value ($\Theta_{O_k} \in R$) for the k-*th* output unit, w_{jk} is the real valued weight connection connecting the j-*th* hidden unit with the k-*th* output unit. F_{IH} is nonlinear multidimensional mapping

$$F_{IH} = \bar{f} \circ A_{IH} \qquad (F_{IH} : \Re^{N_I} \rightarrow D^{N_H}),$$

$$F^{(p)}_{IH_j} = f\left(\sum_{i=1}^{N_I} v_{ij} x^{(p)}_i - \Theta_{h_j}\right)$$

where Θ_{h_j} is the threshold value ($\Theta_{h_j} \in R$) for the j-*th* hidden unit, v_{ij} is the real valued weight connection connecting the i-*th* input unit with the j-*th* hidden unit, $x^{(p)}_i$ is the i-*th* coordinate of the p-*th* input vector $x^{(p)}$, $\bar{f}$ stands for a multidimensional nonlinear sigmoidal transformation in which each dimension of its N_H-dimensional domain vector is transformed by a sigmoidal transfer function f, ($\bar{f} : \Re^{N_H} \rightarrow D^{N_H}$), A_{IH} is an input-to-hidden affine submapping $A_{IH} : \Re^{N_I} \rightarrow \Re^{N_H}$.

Training in MLP Networks

Let T be a training set with cardinality N_P,

$$T = \{[x, y] \mid x \in \Re^{N_I} \wedge y \in \Re^{N_O}\}, \; |T| = N_P,$$

where each pair $[x, y]$ contains the input pattern x of the dimensionality N_I, and the expected output pattern y of the dimensionality N_O. Let u denote a set of free system parameters of a network (weights), $u = (w, h, v, \Theta)$, and the objective function E be defined as follows,

$$E(u, x) = \frac{1}{2N_P N_O} \sum_{p=1}^{N_P} \sum_{k=1}^{N_O} (F_k(u, x^{(p)}) - y^{(p)}_k)^2. \qquad (1)$$

Training in *MLP* networks is a process of minimization the objective function E,

$$\arg\min_u E(u, x),$$

given a finite number of samples $[x, y] \in T$ drawn from an arbitrary sample distribution.

Jacobean matrix, J_F, for a neural network is a matrix of the first derivatives of a network's mapping with respect to the free parameters. Analogously, error matrix, J_E, is a matrix of the first derivatives of an error function with respect to the free parameters. It can be expressed as a multiplication of the diagonal residual matrix Δ_O and Jacobean matrix J_F, $J_E = \Delta_O \cdot J_F$.

The task of rule extraction further requires to define classification and its realization by layered artificial neural networks.

Classification

Let $F = (F_1, \ldots, F_{N_O})$ be an arbitrary multidimensional mapping $F : \Re^{N_I} \rightarrow \Re^{N_O}$, and set $CLASS = \{CLS_1, \ldots, CLS_{N_O}\}$ be a set of classes $CLS_c \in \Re$. Classification CLS with respect to mapping F, $CLS(F)$, is defined as follows,

$$CLS : \Re^{N_I} \rightarrow CLASS$$

$$index(CLS_c) = index(F_c) \text{ such that}$$
$$F_c(x) = \max_l [F_l(x)], \; l = 1, \ldots, N_O,$$

where x is an input vector, F_l is the l-*th* coordinate mapping of F, and the function *index* returns an index of an indexed operand. The classification $CLS(F)$ defines a partitioning of the input space $\Re^{N_I}$ denoted as $PCLS(F)$.

Classification task, given by a training set T having N_O classes: $[CLS_1, \ldots, CLS_{N_O}]$, is performed by a neural network in such a way that each unit in the output layer is a representative of one class CLS_c, $c = 1, \ldots, N_O$. After presenting an input pattern, the strongest response in the output layer is selected as the classification answer of a network.

4. Dynamic Sample Selection

Dynamic sample selection techniques presented here focus on controlling the normed progressive search direction of the optimization technique by appropriate selection of exemplars. This enables determining an exemplar subset of training set that may vary at each iteration of optimization. Hence the name "dynamic sample selection" (DSS). Theoretical foundations of DSS have general validity. However, the specifics resulting from the use of first order line search optimization techniques and the type of the objective function are addressed.

4.1 General Functions & Dynamic Sample Selection

In this subsection the optimization case of general functions by first order line search techniques employing dynamic sample selection is addressed. The only requirement on optimized function E is the existence of the first partial derivatives. For this class of functions, practically very suitable expressions for dependencies of normed search directions on selected set of exemplars at each iteration of optimization procedure are shown. Theoretical details on derivation of the expressions can be found in [42] – [46].

Squared l_2 norm of the gradient is approximately expressed as,

$$\left\| \nabla E(u^{(k)}) \right\|_2^2 \approx \frac{(1-a)}{\left| \alpha^{(k)} \right|} \left| E(u^*) - E(u^{(k)}) \right|,$$

where a is a rate of convergence of steepest descent optimization technique, $\alpha^{(k)}$ is a scaling factor of the search direction at the state $u^{(k)}$, $\nabla E(u^{(k)})$ is a gradient vector at the given state $u^{(k)}$, and u^* is the optimum point.

The expressions for normed vector of search direction formulated above have particularly suitable form not only for the implementation purposes, but also for further analysis. The next intention is to observe dependence of squared l_2 norm of search direction on selected training set at a given step of learning:

$$\left\| \nabla E(u^{(k)}) \right\|_2^2 - \left\| \nabla E_{T^{(k)}}(u^{(k)}) \right\|_2^2 \approx \frac{(1-a)}{\left| \alpha^{(k)} \right|} \cdot \Xi$$

$$\Xi = [\, \left| E(u^*) - E(u^{(k)}) \right| - \left| E_{T^{(k)}}(u^*) - E_{T^{(k)}}(u^{(k)}) \right| \,], \quad (2)$$

where $\nabla E_{T^{(k)}}(u^{(k)})$ is a gradient vector at the state $(u^{(k)})$ for the selected set $T^{(k)}$, and $E_{T^{(k)}}(u^*)$ is a value of E at the optimum point u^* for the set $T^{(k)}$.

This underlines the fact that the diference of normed progressive directions of a neural network presented with the complete training set T and the selected training set $T^{(k)}$ at the step k is proportional to the residual expression (2). Proportionality is give by the scaling factor, $\frac{(1-a)}{\left| \alpha^{(k)} \right|}$. Implying from these theoretical results a general behavior of an arbitrary sample selection technique can be formulated:

For any convergent sequence $\{u^{(k)}\}$ of states of first order optimization technique such that $\{u^{(k)}\} \to u^*$, where u^* is the optimum point, holds: $\left\| s^{(k)}_{T^{(k)}} \right\|_2 \to \left\| s^{(k)} \right\|_2$.

The proof of the statement is detailed in [44]. Freedom of selecting exemplars dynamically at each iteration of training is controlled by the convergence of the optimization procedure. When the optimization procedure is relatively far from the optimum point the freedom for choosing proper exemplars in order to progress is higher. The amount of freedom decreases as the optimization procedure converges to the optimum point. That is, asymptotically l_2 norm of the search direction $s^{(k)}_{T^{(k)}}$ for selected data set $T^{(k)}$ should approach l_2 norm of the search direction $s^{(k)}$ for the complete data set T, as the algorithm reaches the optimum point. It is also important to note that the above statement holds for arbitrary dynamic sample selection mechanism, arbitrary first order optimization procedure, and arbitrary function E.

4.2 Functions having Lipschitz Continuous First Partial Derivatives & Dynamic Sample Selection

In the previous subsection the only assumption on functions to be optimized was the existence of first partial derivatives. The case addressed in this subsection imposes one more assumption on function E, that is, the Lipschitz continuity condition [47]:

For a given initial point $u^0 \in \Re^n$ if $E \in C^1$ on the set $S(u^0) = \{u \,|\, E(u) \le E(u^0)\}$, there exists a Lipschitz constant $K > 0$ such that,

$$\left\| \nabla E(u^{(i)}) - \nabla E(u^{(j)}) \right\|_2 \le K \cdot \left\| u^{(i)} - u^{(j)} \right\|_2$$

for every pair $u^{(i)}, u^{(j)} \in S(u^0)$.

Further focus is on deriving feasible expressions of normed gradient vectors allowing establishment of the dynamic sample selection approach for functions having Lipschitz continuous first partial derivatives. Similarly as in the previous case, the dependence of normed gradient vectors on the selected set of exemplars can be expressed using l_2 norms.

$$\left\| \nabla E(u^{(j)}) - \nabla E_{T^{(j)}}(u^{(j)}) \right\|_2 \ge$$
$$\ge K \cdot \frac{\left| E(u^{(i)}) - E(u^{(j)}) - E_{T^{(j)}}(u^{(i)}) + E_{T^{(j)}}(u^{(j)}) \right|}{\left\| \nabla E(u^{(k)}) \right\|_2^2}$$

Existence of stationary point u^* implies the following.

$$\left| 1 - \left\| \nabla E_{T^{(k)}}(u^{(k)}) \right\|_2 \right| \ge$$
$$\ge K \cdot \frac{\left| E(u^*) - E(u^{(k)}) - E_{T^{(k)}}(u^*) + E_{T^{(k)}}(u^{(k)}) \right|}{\left\| \nabla E(u^{(k)}) \right\|_2^2}$$

The above expressions are generally valid for any two points $u^{(j)}$, $u^{(j)}$, or the stationary point u^*. Their validity is also independent of optimization procedure. Hence they can be used for implementing dynamic sample selection mechanism into first order, second order, heuristic, or any other optimization procedure.

The relationship between the Lipschitz constant K and other parameters is formulated as,

$$\frac{1-a}{\left|\alpha^{(k)}\right|} \leq K.$$

This expression outlines important statement for practical implementation of dynamic sample selection mechanism into first order line search optimization procedures. It helps to simplify expressions for monitoring the dependence of the gradient vectors on selected set of exemplars.

4.3 Implementation and Simulations

A gradient vector ∇E in batch mode of back propagation procedure is formed as the sum of the gradients of the error function E for each exemplar presented to a network. Hence each presented exemplar plays a role in determining the search direction. By eliminating certain exemplars at a given step of training the search direction is modified. Proper modification of the search direction by exemplar selection may then be beneficial for the convergence speed increase of the optimization procedure.

Direct application of the derived theoretical material leads to the dynamic sample selection algorithm based on monitoring the values of the normed gradient vectors.

$$\min_{T^{(k)} \in T} \left[\left\| \nabla E(u^{(k)}) \right\|_2^2 - \left\| \nabla E_{T^{(k)}}(u^{(k)}) \right\|_2^2 \leq PI_1^{(k)} \right], \quad (3)$$

$$PI_1^{(k)} = \frac{(1-a)}{\left|\alpha^{(k)}\right|} \left| E(u^*) - E(u^{(k)}) \right|. \quad (4)$$

Similarly, dynamic sample selection approach can be derived for functions having Lipschitz continuous first partial derivatives.

$$\min_{T^{(k)} \in T} \left[\left\| \nabla E(u^{(k)}) - \nabla E_{T^{(k)}}(u^{(k)}) \right\|_2 \leq PI_2^{(k)} \right] \quad (5)$$

$$PI_2^{(k)} = \frac{K^{(k)}}{\left\| \nabla E(u^{(k)}) \right\|_2} \left| E(u^*) - E(u^{(k)}) \right|, \quad (6)$$

and

$$\min_{T^{(k)} \in T} \left[\left| 1 - \nabla E_{T^{(k)}}(u^{(k)}) \right\|_2 \right| \leq PI_3^{(k)} \Big], \quad (7)$$

$$PI_3^{(k)} = \frac{K^{(k)}}{\left\| \nabla E(u^{(k)}) \right\|_2^2} \cdot \left| E(u^*) - E(u^{(k)}) \right|, \quad (8)$$

where $K^{(k)}$ is a Lipschitz constant at the *k-th* iteration calculated as,

$$K^{(k)} = \frac{\left\| \nabla E(u^{(k)}) - \nabla E(u^{(k-1)}) \right\|_2}{\left\| u^{(k)} - u^{(k-1)} \right\|_2}$$

The expressions (3), (5), and (7) naturally follow from the presented theoretical material. However, they themselves require minimization process in order to find the proper set of selected exemplars $T^{(k)}$. The best solution to this problem would be to test all possible combinations on T. Apparently, this would result in computational excess of a method and impracticality for large data sets. To avoid the impracticality it is necessary to utilize a priori knowledge on importance of exemplars for progress of first order optimization techniques. Importance of a given exemplar is measured in terms of l_1 norm of $\nabla E^{(kp)}$, that is gradient of E for the *p-th* pattern at the k-th iteration.

Practically, in cases when optimization requires higher precision, asymptotic behavior is not always satisfied due to the approximations introduced in terms $PI_1^{(k)}$ (4), $PI_2^{(k)}$ (6), and $PI_3^{(k)}$ (8). Then dynamic sample selection may have divergent behavior and may eliminate large amount samples even if the algorithm has reached the attractor basin.

For this reason, the suppression function is introduced.

$$N_{EP} = (N_P - N_{TP}) \frac{\frac{1}{Iter} \sum_{j=1}^{Iter} PI_i^{(j)}}{PI_i^{(1)}} \quad (9)$$

N_{EP} is the maximum number of eliminated exemplars, N_P is the cardinality of the data set T, N_{TP} is the pre-determined minimum number of exemplars in order to keep the optimization problem well-posed, *Iter* is a given iteration, $PI_i^{(j)}$, i = 1, 2, 3, is one of functions (4), (6), or (8).

The efectiveness of the dynamic sample selection algorithm implemented into the back propagation training algorithm is demonstrated on simulations. Simulation tasks are selected according to the value of E-FP (exemplars & free parameters) ratio. First, performance of dynamic sample selection algorithm is tested on the problem with E-FP ratio 1.66. The second simulation task has high value of E-FP ratio equal to 15.

In the case of E-FP ratio 1.66 the network had configuration 4-3-1. The network was trained on the Lenses data set [48]. The hidden layer of the network contained sigmoidal transfer function units. The network was trained on the Lenses data set. The stopping criterion for network's training was the value of the expected error less than or equal to 0.05. The weights of the network were initialized randomly in the interval $<-0.1, 0.1>$ which corresponded to the steepest part of the sigmoidal nonlinearity in the hidden units. It was observed that the implementation of dynamic sample selection algorithm, based on (3) and (4) into the back propagation training procedure results in 32.06 % overall increase of convergence speed measured in number of training cycles required to reach the given value of the expected error.

In the second case we observed the simulation results for the problem that had E-FP ratio equal to 15, which is extremely high. The network had configuration 4-2-1 with sigmoidal hidden units. The training set was the IRIS data set [49], [50]. Network's free parameters were again initialized as random values in the interval $<-0.1, 0.1>$. Training was terminated when the value of the expected error decreased below 0.013. Overall increase of convergence speed of 4.03 % is indicated for implementation of dynamic sample selection based on (3), (4), and 3.27 % for dynamic sample selection implementation utilizing expressions (5), (6).

5. Parameter Adaptation

This section introduces modification of the first order line search optimization technique with automatically and dynamically adaptable parameters. The focus is on dynamic adjustments of both step length $\alpha^{(k)}$ and momentum term $\beta^{(k)}$. Step length $\alpha^{(k)}$ and momentum term $\beta^{(k)}$ are allowed to take diferent values at each iteration of the optimization procedure [51] - [56].

First observed is the extension of first order line search optimization procedure with adaptable step length $\alpha^{(k)}$ to a procedure that additionally incorporates constant momentum term β. Initially, the following expressions are obtained [56]:

$$|\alpha^{(k)}| \geq \left((1-a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + \right.$$

$$\left. + |\beta| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right), \tag{10}$$

and

$$|\alpha^{(k)}| \leq \left((1+a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + \right.$$

$$\left. + |\beta| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right), \tag{11}$$

where $s^{(k-1)}$ is a search direction at the $(k-1)$-*th* iteration of the optimization procedure.

Assumption of superlinear convergence rates results in the following update formula for step length $\alpha^{(k)}$.

$$|\alpha^{(k)}| = \left(\frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + |\beta| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right) \tag{12}$$

Superlinear convergence rates assumption shrinks the boundaries (10) and (11) for step length $\alpha^{(k)}$ to a single point. This naturally simplifies the line search subproblem to a one step calculation of $\alpha^{(k)}$. In order to attain higher flexibility, it is also recommended to use the median value of (10) and (11) for calculation of the step length $\alpha^{(k)}$,

$$|\alpha^{(k)}| = \left(a \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + |\beta| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right) \tag{13}$$

where a is a parameter determined by user. The dependency of the modifiable momentum term $\beta^{(k)}$ can be obtained from first order line search optimization techniques, and conjugate gradient techniques in particular. Considering the definition of general linear convergence rates and essential formula for parameter updates of conjugate gradient methods [57], the following inequalities are derived [56].

$$|\beta^{(k)}| \geq \frac{1}{\|s^{(k-1)}\|_2} \left(|\alpha^{(k)}| \cdot \|\nabla E(u^{(k)})\|_2 - \right.$$

$$\left. - (1+a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2} \right) \tag{14}$$

$$|\beta^{(k)}| \leq \frac{1}{\|s^{(k-1)}\|_2} \left(|\alpha^{(k)}| \cdot \|\nabla E(u^{(k)})\|_2 - \right.$$

$$\left. - (1-a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2} \right) \tag{15}$$

From the assumption of superlinear convergence rates of conjugate gradient method the following implies.

$$|\beta^{(k)}| = \frac{1}{\|s^{(k-1)}\|_2} \left(|\alpha^{(k)}| \cdot \|\nabla E(u^{(k)})\|_2 - \right.$$

$$\left. - \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2} \right) \tag{16}$$

To determine the dependencies of step length $\alpha^{(k)}$ in the conjugate gradient method it is possible to apply the formerly obtained results. Then for the step length $\alpha^{(k)}$, considering the dynamically adjustable momentum term $\beta^{(k)}$,

$$|\alpha^{(k)}| \geq \left((1-a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + \right.$$

$$\left. + |\beta^{(k)}| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right), \tag{17}$$

and

$$|\alpha^{(k)}| \leq \left((1+a) \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + \right.$$

$$\left. + |\beta^{(k)}| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right), \tag{18}$$

is implied. Accounting for the superlinear convergence rates of the conjugate gradient method,

$$|\alpha^{(k)}| = \left(\frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2} + |\beta^{(k)}| \cdot \frac{\|s^{(k-1)}\|_2}{\|\nabla E(u^{(k)})\|_2} \right) \tag{19}$$

is obtained.

As clearly seen from (14)-(16) and (17)-(19) the expressions for the adjustable momentum term $\beta^{(k)}$ (14)-(16) incorporate step length $\alpha^{(k)}$, likewise the expressions for adjustable step length $\alpha^{(k)}$ (17)-(19) contain momentum term $\beta^{(k)}$. This in practice leads to the dynamic loop. Thus it is impossible to determine which expressions should be calculated first (whether the ones for $\alpha^{(k)}$, or the ones for $\beta^{(k)}$). To overcome this difficulty it is necessary to find another relevant expression for either the adjustable momentum term $\beta^{(k)}$ or the adjustable step length $\alpha^{(k)}$. Theoretical material in [51] already offers possible solution to this problem in the form of the expressions,

$$|\alpha^{(k)}| = a \cdot \frac{|E(u^*) - E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2}, \tag{20}$$

$$|\alpha^{(k)}| = a \cdot \frac{|E(u^{(k)})|}{\|\nabla E(u^{(k)})\|_2^2}. \tag{21}$$

However, it can easily be verified that the substitution of (20) into (16) leads to $\beta^{(k)} = 0$, which on one side supports the

sufficiency of only adjustable step length $\alpha^{(k)}$, but on the other side, it eliminates the momentum term. Hence the only appropriate choice for $\alpha^{(k)}$ is expression (21). Then, substitution of (21) into (16) results in the expression for the adaptable momentum term $\beta^{(k)}$ as follows,

$$\left|\beta^{(k)}\right| = \frac{a}{\left\| s^{(k-1)} \right\|_2 \cdot \left\| \nabla E(u^{(k)}) \right\|_2} \cdot$$

$$\cdot \left(E(u^{(k)}) - \left| E(u^*) - E(u^{(k)}) \right| \right) \qquad (22)$$

Taking into account the absolute value $\left| E(u^*) - E(u^{(k)}) \right|$ in (22) the following two expressions for the modifiable momentum term further imply,

$$\left|\beta^{(k)}\right| = a \cdot \frac{E(u^*)}{\left\| s^{(k-1)} \right\|_2 \cdot \left\| \nabla E(u^{(k)}) \right\|_2}, \qquad (23)$$

and

$$\left|\beta^{(k)}\right| = a \cdot \frac{2E(u^{(k)}) - E(u^*)}{\left\| s^{(k-1)} \right\|_2 \cdot \left\| \nabla E(u^{(k)}) \right\|_2}. \qquad (24)$$

It is important to note that the convergence proof in [56] further eliminates expression (24). Use of expression (24) results in the divergence of the optimization technique. Then the only suitable expression for adaptable momentum term $\beta^{(k)}$ is (23). The constant a stands for the universality of the algorithm. This leads to the following modification of the conjugate gradient optimization technique.

ALGORITHM 1

1. Set the initial parameters: $u^{(0)}$, $E(u^*)$, (β).
2. Calculate the gradient $\nabla E(u^{(k)})$.
3. *Constant momentum:* calculate $\alpha^{(k)}$ according to expression (12) or (13).
   Adaptable momentum: calculate $\alpha^{(k)}$ according to (21) and $\beta^{(k)}$ as (23).
4. Update the system parameters as follows.

$$u^{(k+1)} = u^{(k)} - \alpha^{(k)} \cdot \nabla E(u^{(k)}) + \beta^{(k)} \cdot s^{(k-1)}.$$

ALGORITHM 1 has substantially simplified the line search subproblem (step 3). The proper parameters $\alpha^{(k)}$ and $\beta^{(k)}$, (β), are determined automatically in a single calculation. ALGORITHM 1 has a linear computational complexity $O(N_F)$, where N_F is a number of free parameters. The necessity of keeping the track of the previous search direction $s^{(k-1)}$ in conjugate gradient techniques leads to the linear memory requirements $O(N_F)$ of ALGORITHM 1. Despite the simplicity of the line search subproblem, ALGORITHM 1 is convergent with superlinear convergence rates [56]. The superlinear convergence rates are established under the assumption that second and higher order terms of Taylor expansion of the objective function E around the optimum point are negligible for convergence of the sequence of points $[u^{(k)}]_k$ generated by ALGORITHM 1.

ALGORITHM 1 is demonstrated in Fig. 3. ALGORITHM 1 (charts *b)* and *c)*) clearly converges substantially more smoothly

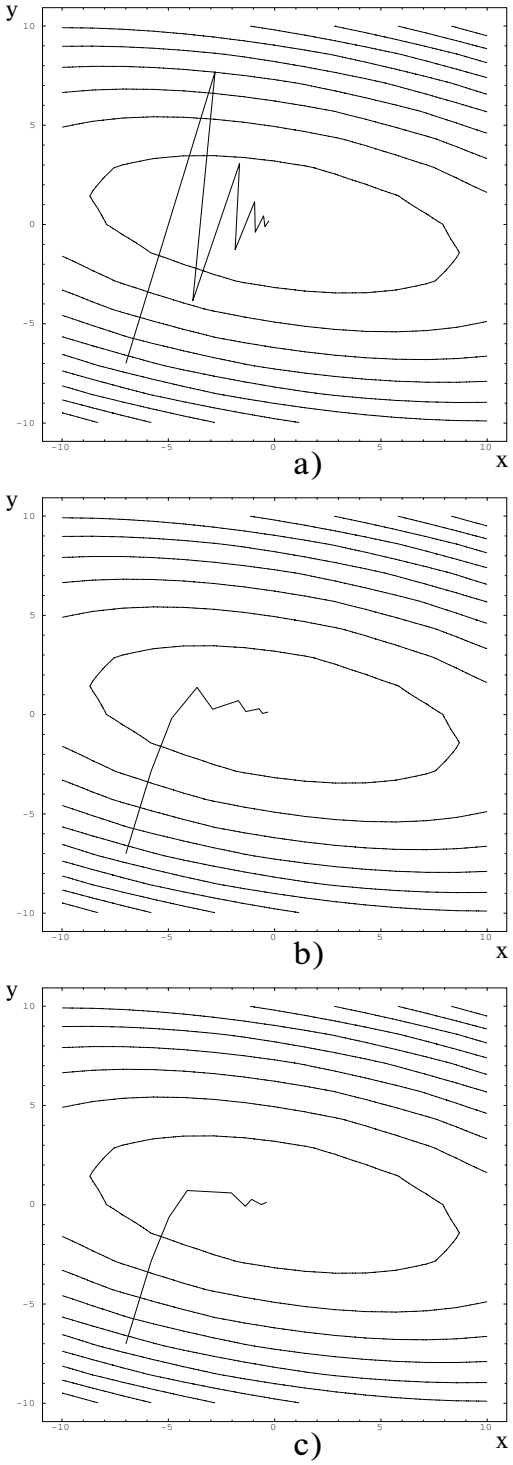


Fig. 3 Comparison of optimization progress between ALGORITHM 1 (chart b) (constant β = 0.1) and c) (adjustable α, β)) and BP with momentum (α = 0.3, β = 0.1) (chart a)) on quadratic function f(x, y) = $= 0.5x^2 + 3y^2 + xy$ from the starting point [-7,-7]. ALGORITHM 1 had setting: E(u) = 0, a = 1. Stopping criterion was the value f(x, y) ≤ 0.1. It is evident that the progress of ALGORITHM 1 is smoother and faster than BP with momentum.*

to the optimum point than the conventional conjugate gradient method (chart *a)*) having a constant step length and momentum.

5.1 Simulations

In this subsection the effectiveness of the algorithms is practically demonstrated on five tasks represented by the following data sets: Lenses [48], Glass, Monks 1 [58], Monks 2 [58], and Monks 3 [58]. The presented algorithms were applied to training various MLP networks to perform tasks given by five, the above mentioned, data sets. Neural network's performance was optimized according to the mean square error. The stopping criterion was the value of the expected error.

In the case of the Lenses data set [48], a neural network had configuration 4-3-1 with sigmoidal hidden units. Expected error was set to $5 \cdot 10^2$. In the Glass problem, a network was configured as: 9-5-1 (sigmoidal hidden units) and the expected error was equal to 0.35. Finally, for Monks 1, 2, and 3 problems [58] a neural network structure was set as: 6-3-1 (sigmoidal hidden units), and the expected error was equal to 0.103. Network's weights were initialized randomly in the interval $< -0.1, 0.1 >$, which corresponded to the steepest region of the sigmoidal transfer function of the hidden units. The parameter a was equal to 1. In case network's error did not converge to the value less than or equal to the expected error within 20000 cycles, the training process was terminated. It is interesting to note that additional stopping condition of maximum 20000 cycles was practically applied only to the BP employing standard first order techniques. ALGORITHM 1 always converged.

The experiments were performed with the value of the step length (learning rate) for BP corresponding to the best results of BP as reported in [51] (in Monks 1 case $\alpha = 0.8$, and for all other data sets $\alpha = 0.9$). The momentum term ranging from 0.1 to 0.7 in 0.1 increments was then applied. BP with the momentum term and the best value of step length (denoted in further text as BPM) was compared to ALGORITHM 1 with constant momentum term (see results in Table 1), and with automatically adjustable momentum term (see Table 2). Values of the constant momentum term in ALGORITHM 1 were set equal to the values of the momentum term in BPM. For a given setting of learning rate and momentum term the simulations were run 10 times for different randomly initialized weights in the interval $< -0.1, 0.1 >$. Percentual convergence speed increases were calculated in order to simplify the comparison. Hence the values in Table 1, and 2 represent ten-run-averages. Criterion for comparison of the convergence speed was the number of cycles required to decrease the mean square error E of a neural network below the value of the expected error.

It is clear, from Table 1, and 2, that the proposed algorithm (that is, ALGORITHM 1) converged substantially faster than the standard techniques. As previously mentioned, ALGORITHM 1 converged each time, whereas BPM for some initial setting of weights and parameters α, β could not achieve convergence even after 20000 cycles. This accounts for higher stability of ALGORITHM 1.

Table 1: Comparison of ALGORITHM 1 (constant momentum) and BPM with setting of constant learning rate that corresponded to the best obtained results of BP. Momentum term was set from 0.1 to 0.9 in 0.1 increments, and kept constant for both ALGORITHM 1 and BPM. ALGORITHM 1 had the following setting: $E(u^*) = 0$, $a = 1$. Values in the table represent ten-run-averages of percentual convergence speed increase of ALGORITHM 1 over BPM. Convergence speed was compared in terms of cycles required to reach a given value of the expected error.

β	0.1	0.2	0.3	0.4	0.5	0.6	0.7	AVR
Lenses (α = 0.9)	43.99	19.65	7.03	44.49	93.12	94.46	95.14	56.84
Glass (α = 0.9)	54.4	52.05	47.76	44.43	39.02	35.43	29.05	43.18
Monks 1 (α = 0.8)	57.37	51.73	40.42	39.7	49.38	85.07	90.95	59.23
Monks 2 (α = 0.9)	-12.41	-28.49	4.55	58.95	86.87	95.18	96.07	42.96
Monks 3 (α = 0.9)	40.91	34.33	31.97	58	82.07	87.39	89.7	60.62

Table 2: Comparison of ALGORITHM 1 (adaptable momentum) and BPM with learning rate setting corresponding to the best obtained results of BP. Momentum term setting varied from 0.1 to 0.9 in 0.1 increments. ALGORITHM 1 used the setting: $a = 1$. Values in the table stand for ten-run-averages of percentual convergence speed increase of ALGORITHM 1 over BPM. Convergence speed comparison was made in terms of number of cycles necessary to decrease the network's error below the value of the expected error.

β	0.1	0.2	0.3	0.4	0.5	0.6	0.7	AVR
Lenses (α = 0.9)	51.23	20.5	10.14	46.95	93.18	95.15	97.13	59.18
Glass (α = 0.9)	55.35	53.38	48.93	45.97	41.35	37.89	30.25	44.73
Monks 1 (α = 0.8)	59.37	53.25	43.19	41.53	53.68	87.79	93.25	61.72
Monks 2 (α = 0.9)	1.45	-3.47	14.33	60.78	87.96	97.31	97.93	50.89
Monks 3 (α = 0.9)	42.35	40.37	35.96	59.97	83.47	89.93	92.23	63.47

6. Structural Adaptation

The progress of a training algorithm can be seen as a movement in a space with principal coordinates defined by the singular vectors. The error surface increases most rapidly in the direction of a vector corresponding to the maximum singular value and most slowly in a direction of a vector corresponding to the minimum singular value. The singular values give partial information about the determination of the next step of an algorithm. They provide a relative measure on the direction and the proportion of the movement. Since the singular value decomposition is computationally expensive task it is desirable to approximate location of the singular values given by the spectral radius estimate [31].

Spectral Radius Estimate
Let $A \in M_{m,n}$. For the spectral radius of singular values holds,

$$\rho(A) \leq min[\rho'(A), \rho''(A)],$$

where

$$\rho''(A) = min\left\{ \max_i \left(\sum_{j=1}^{m} |a_{ij}| \right), \max_j \left(\sum_{j=1}^{n} |a_{ij}| \right) \right\},$$

$$\rho'(A) = min\left\{ \max_r \left(\sqrt{\sum_{c=1}^{m} \left| \sum_{j=1}^{n} a_{rj} \cdot a_{jc} \right|} \right), \max_c \left(\sqrt{\sum_{r=1}^{m} \left| \sum_{j=1}^{n} a_{rj} \cdot a_{jc} \right|} \right) \right\}.$$

"r" and "c" denote row and column indices of AA^T, respectively.

How the estimates ρ'' (25) and ρ' (25) typically work in practice is shown in Fig. 4. The task depicted in Fig. 4 is training an MLP neural network with the configuration 2-2-1 on the XOR problem. Network's connections were initialized by the exponential series with base 0.9 and 0.5 for hidden-to-output and input-to-hidden weights, respectively. Steepest descent version of BP with the constant learning rate 0.9 was used. Training was terminated when the mean square error decreased below 10^{-2}. The actual spectral radius ρ in Figure 4 was obtained by the singular value

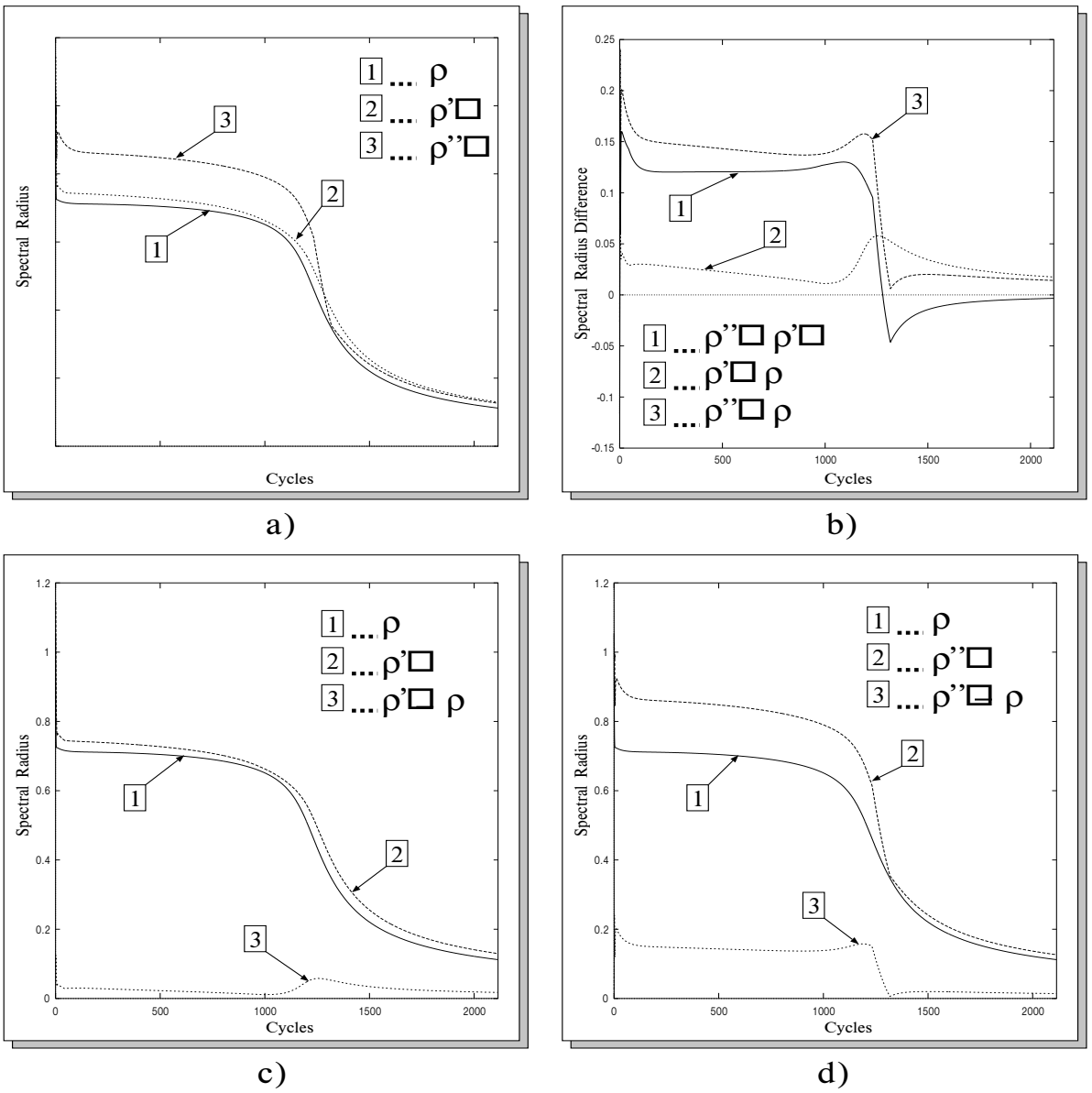


Fig. 4 Typical behavior of the estimates of spectral radiuses ρ' and ρ''. The spectral radius ρ was obtained by singular value decomposition. A training task for MLP neural network with the configuration 2-2-1 was the standard benchmark XOR. Chart a) shows the values of ρ, ρ', and ρ'' at each cycle of training. Chart b) displays the differences $\rho'' - \rho'$, $\rho' - \rho$, and $\rho'' - \rho$. Details of the relationship between ρ and ρ' is shown in chart c). Analogously, the values of the estimate ρ'' in relation to ρ are depicted in chart d).

decomposition. Note that the XOR problem serves as a testbed for essential linear non-separability. The training set contains four patterns $T = \{[(0, 0), 0]; [(0, 1), 1]; [(1, 0), 1]; [(1, 1), 1]\}$. A sufficient three layer network configuration incorporates two input units, two hidden units, and one output unit.

The reason for having two different expressions for spectral radiuses of singular values and taking minima of them follows from the need for more precise location and from the nature of the optimization techniques. When the first order optimization procedure is relatively far from the equilibrium point, it is more likely that ρ'' gives more precise estimate of the spectral radius because the derivatives will be higher in value. However, when the algorithm converges (in an ideal case to 0) then also the first order derivatives should converge, possibly to 0, and thus become smaller. In this case, when the derivatives are lesser than 1, the expression ρ' may locate the singular values within a smaller interval.

As it can be seen from the charts a) and b), the estimate ρ' was more precise when the network was relatively far from the optimum point. Once the attractor basin was reached, the estimate ρ'' showed slightly higher precision than ρ'. The reason for this is the difference of the slopes of the error surface. After the initial progress (approximately 40 cycles) the network was progressing on the flat region of the error surface for almost 1200 cycles. Surface flatness is indicated by small gradient values. Therefore, the estimate ρ' had smaller values than ρ''. As the algorithm reached the attractor basin, the network started to progress on a sharper slope of the error surface. The gradients were higher. Thus the estimate ρ'' had smaller values than ρ'. When the algorithm converged to the optimum point, both ρ' and ρ'' converged to almost the same values (see chart a) and also differences in chart b)). Specifics of each estimate ρ' and ρ'' in relation to ρ are depicted in charts c) and d).

Taking into account the static and dynamic importance of the weight connections in the network the authors propose the measure of use of the network's potentials (to progress) at each step of a training algorithm as,

$$PM = \frac{1}{\rho(J_E)} \frac{1}{N_F} \sum_{u_l \in u} \left| u_l \sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right|, \qquad (25)$$

where E stands for some specific error function, u_l is the *l-th* free parameter of a network, and $\rho(J_E)$ is the estimate of a spectral radius of the error matrix J_E. The expression (25) represents the overall average performance of a network.

The value of the estimate of a spectral radius cannot exceed the minimum of the maxima of row and column sums of the elements in an error matrix for an artificial neural network. That is, the largest singular value σ_{max} lies within the interval specified by the matrix measures (or norms) such as column or row l_1 norms. Regarding search directions, a network's progress in each dimension is limited by the value of the estimate of a spectral radius. The sum of column elements of a specific matrix (depending on error function) represents how far a network will move in a particular dimension. Hence, the network in any dimension cannot progress more than the estimate shows. Considering the static relevance of weight $u_l \in u$, given by its real value ($u_l \in \Re$), multiplied by the dynamic relevance of the connection represented by the sum of the gradients for each pattern sample $\left(\sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right)$ and then scaling the multiplication with respect to the maximum progress formulated by the estimate of a spectral radius ρ, the overall average performance of a network is obtained at each iteration (25). From (25) immediately implies that the individual performance of each connection is given as,

$$I_{pm_{u_l}} = \frac{1}{\rho(J_E)} \left| u_l \sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right|. \qquad (26)$$

Similarly the maximum performance of a network during training can be measured,

$$PM_m = \frac{1}{\rho(J_E)} \max_{u_l \in u} \left| u_l \sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right|. \qquad (27)$$

The relevance and importance of the above measures is illustratively demonstrated on the standard benchmark task XOR.

First, the relevance of measures (25) and (27) is demonstrated. To obtain a closer look at the behavior of a training procedure the auxiliary expressions are evaluated,

$$D_{pm} = \sum_{u_l \in u} \left| u_l \sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right|, \qquad (28)$$

$$D_{pm_m} = \max_{u_l \in u} \left| u_l \sum_{p=1}^{N_p} \frac{\partial E^{(p)}}{\partial u_l} \right|. \qquad (29)$$

Configuration of the network was 2-2-1 and the weight connections were initialized by the exponential series with base 0.9 for the hidden-to-output weights and 0.5 for the input-to hidden weights. A batch mode of BP training with a constant learning rate equal to 0.9 was used. Simulation charts are shown in Figure 5 and Fig. 6. Fig. 5 shows the performance of a network according to the performance measure (25) and Figure 6 depicts the maximum performance chart for a network according to the measure (27). Both figures clearly indicate four distinguishable phases of a training procedure.

The first phase depicts the initial progress of a network and sudden stagnation. The error E decreased in a few starting training cycles and then stabilized. This phase is indicated by a rapid decline of both the average performance PM (25) and the maximum performance PM_m (27), which exactly reflects the fact that the network was not progressing, since it has reached the flat surface of an error landscape. The estimate of a spectral radius $\rho(J_E)$ had lightly fluctuated around 1 and then stabilized. Detail of this phase is given in upper charts of Fig. 5 and Fig. 6. It shows first 40 cycles of a training procedure. Low levels of D_{pm} and D_{pm_m} measures underline the fact that the network reached a flat region of error surface which is, according to the error E, positioned quite high from the minima.

The second phase shows slow progress of a network on a flat region of an error surface. The error E changes very little. The

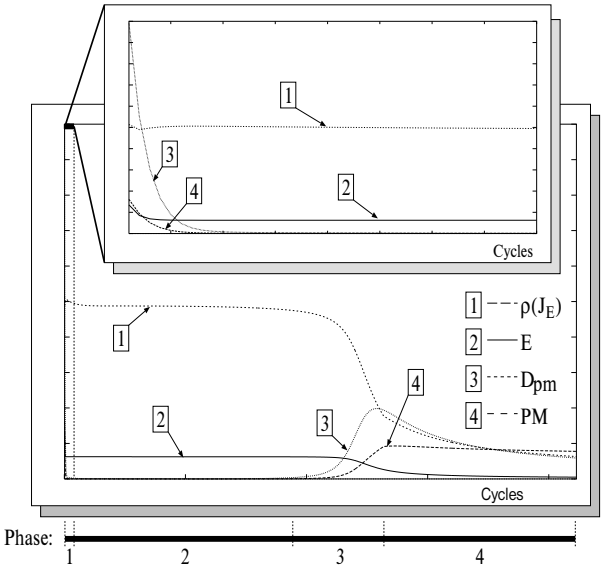


Fig. 5 Training behavior of the network with configuration 2-2-1 for the XOR problem according to the measures PM and D_{pm}. Four training phases could be distinguished. The detail of the first phase is given in the upper chart. The network converged to the point where $E < 10^{-2}$ after 2114 cycles. A batch mode of BP training used constant learning rate equal 0.9.

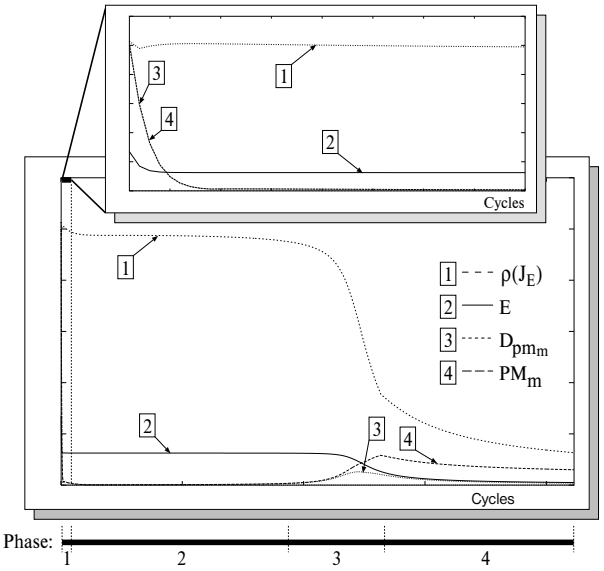


Fig. 6 Behavior of the network with configuration 2-2-1 according to the measures PM_m and Dp_{m_m} during training to perform the XOR task. Similarly as in the previous figure there are four distinguishable phases. The first phase is shown in detail in the upper chart. The error $E < 10^{-2}$ was reached after 2114 cycles for a batch mode of BP training with constant learning rate 0.9.

performance of a network (by means of the measures *PM* and PM_m) is very low. The estimate of a spectral radius $\rho(J_E)$ based on l_1 norms is relatively stable and high. High values of the error E and the estimate of a spectral radius $\rho(J_E)$, and low values of the expressions D_{pm} and D_{pm_m} imply that the flatness of an error surface is caused by sign oscillations of the weight gradients for different patterns.

In the third phase, the network passed over the at region of an error surface and reached the basin of attraction of an attractor point. It started to converge toward the minimum. The error *E* smoothly decreased and according to the performance measures *PM* and PM_m the network indicates rapidly increased progress. Declining estimate of a spectral radius $\rho(J_E)$ and rising values of D_{pm} and D_{pm_m} indicate that the network balanced its computational resources by eliminating the oscillations of gradients for different training samples. A sharp rise of D_{pm} and D_{pm_m} indicates that the network progressed on a sharp slope of an error surface.

The fourth phase shows the smooth convergence of a network toward the minimum point. Error nicely decreases and the network progresses at constant rate, as indicated by the average performance measure *PM*. The maximum performance PM_m smoothly stabilizes. The stable convergence ratio is also nicely indicated by stabilization of the estimate of a spectral radius as well as measures D_{pm} and D_{pm_m}. Low values of maximum gradients (D_{pm_m}) underline the reach of the optimum point.

6.1 Structural Adaptation Utilizing Learning Performance Measures

The relevance and importance of the derived performance measures for structural modifications of a network, particularly pruning, is shown. As previously mentioned, the expression (26) represents a combination of static and dynamic importance of a specific weight connection in a structure of a network. Thus it can be used for detecting less important structural elements suitable for pruning. The use of the measure (26) for pruning is illustratively depicted again on an example of the XOR problem (see Fig. 7).

An overdetermined network structure with configuration 2-3-1 was generated. The initial values of the weights were set in the same way as in the previously mentioned simulation. A batch mode of BP training procedure with the constant learning rate 0.9 was used. It can be seen that the original network structure (2-3-1) was able to converge approximately after 1900 cycles (to be precise 1896 cycles) to the point where the error $E < 10^{-2}$. The evidence that the network converged is shown in upper-left chart of Fig. 7. After approximately 1000 cycles the network started to converge. The performance measure *PM* as well as D_{pm} rose. The estimate of a spectral radius started to decline. And from around cycle 1400 the network was progressing at a constant rate, as indicated by the stabilized curve of the *PM* measure.

When the network reached cycle 111 the maximum performance measure PM_m indicated minimum value ($1.229526 \cdot 10^{-3}$). At this point, since the progress was very slow, the network was

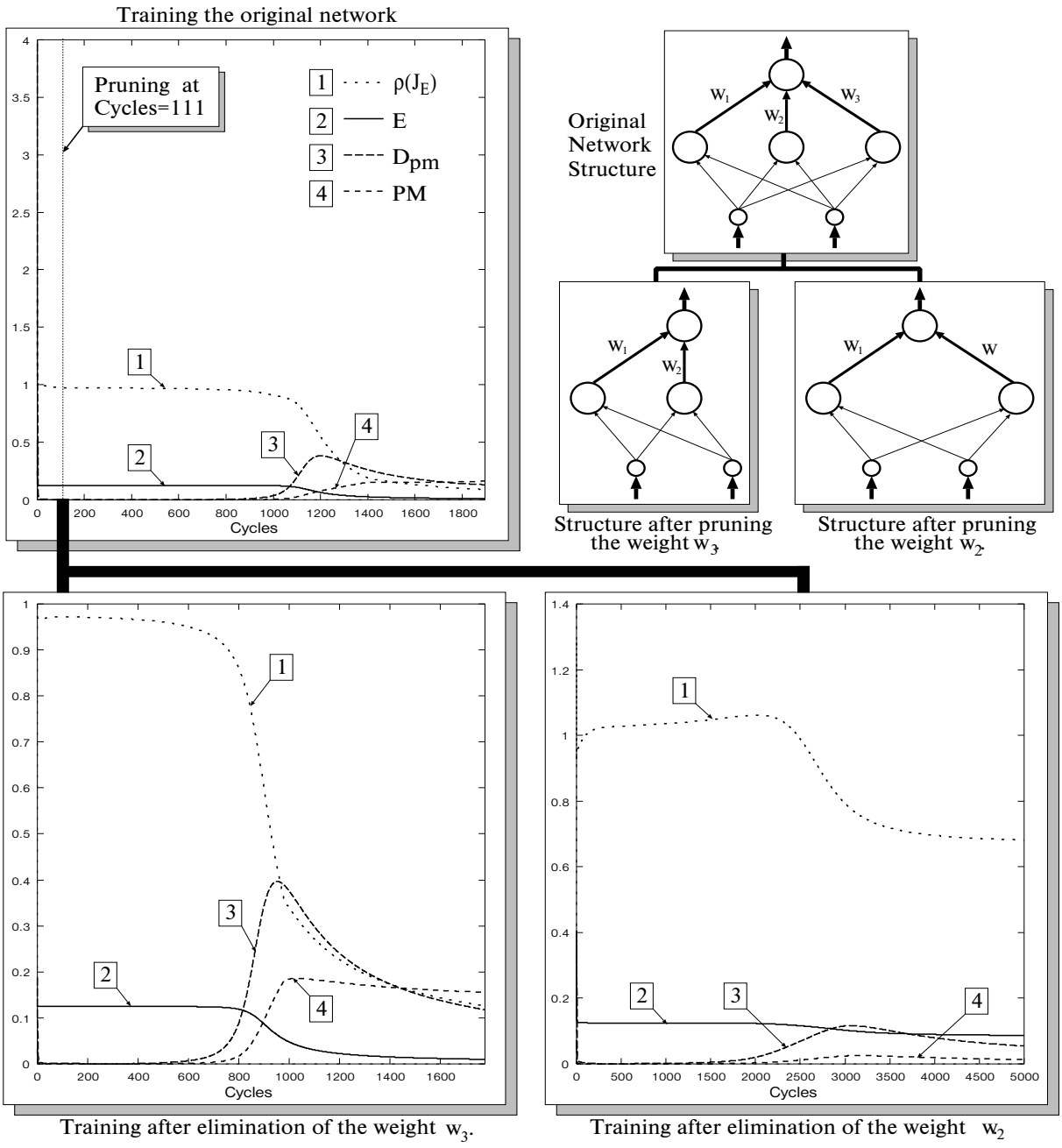


Fig. 7 The effect of pruning the original network with configuration 2-3-1 on training behavior. The pruning criterion was $I_{pm_{u_j}}$. Top-left chart depicts the training behavior of the original network (2-3-1). Top-right part shows the modification of the structure after elimination of the hidden-to-output weight w_3 (left) and after elimination of the hidden-to-output weight w_2 (right). Bottom-left chart shows the training behavior of the network after pruning the hidden-to-output weight w_3 (the weight with low individual performance measure). Bottom-right chart shows the training behavior of the network after pruning the hidden-to-output weight connection w_2 (the weight with high individual performance measure). The pruning point for the original network structure was cycle 111.

suitable for pruning. The individual performance measures (26) of hidden-to-output weights were evaluated. It was found that at the cycle=111 the individual performance measure for the hidden-to-output weight w_3 was extremely low ($1.5993 \cdot 10^{-5}$), which was about ten times less than that for the hidden-to-output weight w_2 ($1.20916 \cdot 10^{-4}$). The individual performance measures for the hidden-to-output weights w_2 and w_3 were at the same level (approximately ten times higher than $I_{pm_{w_3}}$). Hence the suitable candidate for pruning was the weight w_3. The bottom-left chart in Figure 7 shows the training behavior of the network after pruning

the hidden-to-output weight w_3. Six hundred cycles after pruning the weight w_3 the network started to converge (see the charts of measures *PM*, $\rho(J_E)$, and D_{pm}). And 1000 cycles after pruning the network was progressing at almost constant rate (see the curve *PM*). At cycle 1777 after pruning the network reached a state where the error E was less than 10^{-2}.

It is also important, on the other hand, to observe the behavior of a network when the weight with a high individual performance measure (26) was pruned. The bottom-right chart of Figure 7 shows the training behavior of the network after pruning the hidden-to-output weight connection w_2, that is, the connection with high individual performance measure (26). It is clear (from the bottom-right chart in Figure 7) that the network after pruning the hidden-to-output weight w_2 was not able to converge even after 5000 cycles. Approximately 2000 cycles after pruning the performance *PM* of the network slightly rose, however, after reaching the maximum point at cycle=2995 ($2.036367 \cdot 10^{-2}$) the performance *PM* declined. The estimate of a spectral radius $\rho(J_E)$ dropped, but again stabilized at a relatively high level (around 0.7). The measure D_{pm} rose, but then decreased. From the curves of $\rho(J_E)$, *PM*, and D_{pm} it can be concluded that the network reached another relatively flat region of the error surface. Generally, the performance of the network was very low during the whole period of 5000 cycles after pruning the hidden-to-output weight w_2. This underlines the relevance of the individual performance measure (26) as well as the other measures (25) and (27).

It is clear from the above and from (25), (26), and (27) that the essential reference ground for comparison of the combined static and dynamic importance of the network elements is the estimate of a spectral radius $\rho(J_E)$. It is the only suitable reference ground directly implied from the nature of the first order optimization procedures. Furthermore, it has three remarkable properties:

It is dynamic. The estimate of a spectral radius $\rho(J_E)$ dynamically changes according to the state of a training procedure. Thus it represents the most suitable dynamic reference ground at every step of training. This naturally overcomes the problems of choosing the right reference value, which is essential for regularization based approaches (e.g. choosing the right preassigned parameter u_0 in the weight elimination procedure). Most likely there is a very small class of problems which satisfies the optimum condition for constant reference ground.

It is sensitive. The estimate of a spectral radius $\rho(J_E)$ is sensitive enough to reflect the occurrences leading to the progress of a network.

It can be manipulated. Structural modifications and sample selection can influence the values of the estimate of a spectral radius $\rho(J_E)$ so as to maximize the expressions (25), (26), and (27).

The third point, the ability to manipulate the estimate of a spectral radius $\rho(J_E)$, is of primary interest in this study. Although there have been a variety of modifications of first order optimization techniques and their implementation in the field of artificial neural networks, a deterministic theoretical analysis is still lacking. This is true particularly for cases where the training algorithms incorporate dynamic structural changes. Hence, it is the researcher's intention to shed some light onto this specific area of a neural network field.

7. Rule Extraction

The rule extraction problem from neural networks is addressed in its principal form. That is, given an arbitrary three layer artificial neural network trained on particular data extract the rules that reflect the network's data classification as correctly as possible. Note that this is general and essential principle of rule acquisition. The only choice of three layer neural networks is due to their universal approximation abilities. Methodology introduced here is independent of rule types, network types, and learning strategy. Hence, there is no assumption of any a priori knowledge implementation into a neural network.

Given a mapping F of three layer artificial neural network trained on the classification task described by a training set T, the primary interest is to derive the rules that classify the data as correctly as neural network. This is the underlining problem of rule extraction from trained neural networks. Sometimes there must not be the solution to this problem, however, the classification rules describing data with certain accuracy can still be obtained. The availability of solution to rule extraction problem is stated in the following.

Crisp Rule Extraction

Let bu be a set $bu = \{1, 0, 1\}$ (or $bu = \{1, 1\}$). Let $F = F_{HO} \circ F_{IH}$ be a mapping of a nonlinear neural network with structure $N_I - N_H - N_O$ such that input-to-hidden weights v_{ij} and hidden-to-output weights w_{jk} are real valued parameters, $v_{ij} \wedge w_{jk} \in \Re$, and $N_H \geq [\log_{|bu - \{0\}|}(N_O)]$. If there exists a set of vectors $[cr^{(1)}, \ldots, cr^{(N_P)}]$ such that,

$$\max_{k=1,\ldots,N_O} [w_k^T \cdot F_{IH}^{(p)}] = \max_{k=1,\ldots,N_O} [\mathrm{b}w_k^T \cdot (F_{IH}'^{(p)} + cr^{(p)})],$$

$$F_{IH}'^{(p)} = f(bv, x^{(p)});\ cr^{(p)} \in [cr^{(1)}, \ldots, cr^{(N_P)}],$$

where $bw_k = (bw_{1k}, \ldots, bw_{N_H k})$, $bw_{jk} \in bu$, and $bv = (bv_{11}, \ldots, bv_{N_I N_H})$, $bv_{ij} \in bu$, there exists the representation of the network's mapping in form of the rules,

$$IF \left(\bigwedge_{i=1,\ldots,N_I} LOx_i^{(k)} \mathcal{E} < x_{iS}^{(k)}, x_{iE}^{(k)} > \right) THEN\ CLS_k, \qquad (30)$$

where $k = 1, \ldots, N_O$, LO is either void or logical operator of negation $\neg$, and $\mathcal{E}$ is either $\in$ or $\notin$. The rules (30) classify the patterns $[x^{(p)}, y^{(p)}] \in T$, $p = 1, \ldots, N_P$, as correctly as neural network.

General proof of the above statement is provided in [38]. Practical implications of the constructive proof lead to the methodology for deriving the rules from arbitrary three layer neural network. The methodology is applicable even in the cases

when the theoretical conditions are not strictly satisffed. It can be described in the following 5 steps.

1) Code the weight vectors w_k, v_j into trinary vectors bw_k, bv_j according to the shortest Euclidean distance ($min\ d(w_k, bw_k)$, and $min\ d(v_j, bv_j)$).
2) Based on the transferred weight vectors bw_k generate the initial rules,

$$IF\ (\bigwedge_{i=1,\ldots,N_H} LO\ F_{IHj}^{(k)})\ THEN\ CLS_k, \tag{31}$$

such that LO is void if $bw_{jk} = 1$ and LO is $\neg$ if $bw_{jk} = -1$. Note that if $bw_{jk} = 0$, the corresponding $F_{IH_j}^{(k)}$ is eliminated.

3) Expand each $F_{IH_j}^{(k)}$ in (31) according to the trinary vectors bv_j into the form,

$$F_{IH_j}^{(k)} \rightsquigarrow \bigwedge_{i=1,\ldots,N_I} x_i^{(k)} \mathcal{E} < x_{iS}^{(k)}, x_{iE}^{(k)} >, \tag{32}$$

such that $\mathcal{E}$ is $\in$ if $bv_{ij} = 1$ and $\mathcal{E}$ is $\notin$ if $bv_{ij} = -1$. Again, if $bv_{ij} = 0$ the corresponding $x_i^{(k)}$ is eliminated.

4) Simplify the expanded rules by merging the same statements and eliminating contradictions. Then present the rules of the form,

$$IF\ (\bigwedge_{i=1,\ldots,N_I} LO x_i^{(k)} \mathcal{E} < x_{iS}^{(k)}, x_{iE}^{(k)} >)\ THEN\ CLS_k. \tag{33}$$

5) Find appropriate intervals $< x_{iS}^{(k)}, x_{iE}^{(k)} >$ and substitute them into (33). Finally, generate the resulting rules.

As it can be seen in step 4) the above described methodology for rule extraction contains mechanism of optimizing the rules. This allows obtaining relatively simple rules even if the network's structure was overdetermined for a given classification task. In other words, the structural redundancies of a neural network can partially be eliminated when finalizing the rules. This feature has an important practical value since the network's structures may not always be optimal for a given classification task. The simplification mechanism will practically be demonstrated in the next section.

Important part is also finding the appropriate intervals $< x_{iS}^{(k)}, x_{iE}^{(k)} >$ in the step 5). If the conditions of the theorem are satisfied and rule includes the term $x_i^{(k)} \in < x_{iS}^{(k)}, x_{iE}^{(k)} >$, then the intervals can be obtained directly by tracing the min/max values of input coordinates for given class CLS_k. If the rule contains the term $x_i^{(k)} \notin < x_{iS}^{(k)}, x_{iE}^{(k)} >$ and the conditions of the theorem are satisfied, again the intervals are obtained directly by eliminating the intervals of input coordinates for classes CLS_j, $j \neq k$. However, the functionality of the methodology extends even to the cases where the conditions of the theorem do not strictly hold. Then the step 5) can be slightly more complicated. The approaches for obtaining the intervals (or membership functions) can vary [59], [60]. From the practical experience we can recommend the method of dichotomizing the boundaries for conflict patterns.

7.1 Simulations

The effectiveness of the methodology described in the previous section is demonstrated on the IRIS data set [49]. First, the case of rule extraction for optimized structure of a network is presented and in the second case the rule simplification mechanism is demonstrated on the overdetermined network structure.

Structure modifying training techniques play an important role in rule extraction approaches. Essentially, they lead to simpler initial rules (31) after the first three steps in the proposed methodology. In the first experimental task a structure modifying training technique based on performance measures was used [31]. Back propagation learning was terminated when the expected mean square error was less than 0.057. Resulting network, after transforming the original weight vectors into trinary vectors, had 3-2-4 structure. The transformation of weight vectors was as follows.

$$\begin{aligned} w_1 &= [0,\ 2.72] \rightarrow [0,\ 1] \\ w_2 &= [0.51,\ -0.94] \rightarrow [1,\ -1] \\ w_3 &= [0.65,\ -1.74] \rightarrow [1,\ -1] \\ v_1 &= [0,\ 0,\ 2.15,\ 0] \rightarrow [0,\ 0,\ 1,\ 0] \\ v_2 &= [0,\ 0,\ 0,\ -1.7] \rightarrow [0,\ 0,\ 0,\ -1] \end{aligned}$$

According to the proposed rule extraction methodology, from the network's structure the following rules immediately imply.

$$IF\ (x_4^{(1)} \notin < x_{4S}^{(1)}, x_{4E}^{(1)} >)\ THEN\ CLS_1 \tag{34}$$

$$IF\ (x_3^{(2)} \in < x_{3S}^{(2)}, x_{3E}^{(2)} > \wedge\ x_4^{(2)} \in < x_{4S}^{(2)}, x_{4E}^{(2)} >)\ THEN\ CLS_2 \tag{35}$$

$$IF\ (x_3^{(3)} \in < x_{3S}^{(3)}, x_{3E}^{(3)} > \wedge\ x_4^{(3)} \in < x_{4S}^{(3)}, x_{4E}^{(3)} >)\ THEN\ CLS_3 \tag{36}$$

Since the above rules cannot be further simplified the next step is only to determine the appropriate intervals. By detecting the min/max values of input coordinates for different classes the following intervals are obtained.

$$x_4^{(1)} \notin <1.0,\ 2.5>;\ x_3^{(2)} \in <3.0,\ 5.1>;\ x_4^{(2)} \in <1.0,\ 1.8>;$$

$$x_3^{(3)} \in <4.5,\ 6.9>;\ x_4^{(3)} \in <1.4,\ 2.5>$$

Inserting these intervals into the rules (34), (35), (36) leads to 94.6 % correct classification (142 correctly classified patterns out of 150). Dichotomizing the boundary of the input coordinate x_4 for conflict samples results in modification of $x_4^{(2)}$ and $x_4^{(3)}$ intervals: $x_4^{(2)} \in <1.0,\ 1.7>$, $x_4^{(3)} \in <1.35,\ 2.5>$. Substituting these boundaries into (35) and (36) gives 97.3 % correct classification (146 correctly classified exemplars out of 150).

The second simulation example demonstrates the rules simplification mechanism. An overdetermined network with structure 4-3-3 was trained using back propagation with constant learning rate 0.7 until the expected mean square error was less

than 0.057. The following weight vectors and their codified trinary vectors were obtained.

$$w_1 = [0.02, -0.007, 1.003] \rightarrow [0, 0, 1]$$
$$w_2 = [1.1, 0.71, -1.28] \rightarrow [1, 1, -1]$$
$$w_3 = [-1.31, 1.03, 1.123] \rightarrow [-1, 1, 1]$$
$$v_1 = [1.69, 2.0, -2.68, -3.29] \rightarrow [1, 1, -1, -1]$$
$$v_2 = [-0.29, -0.46, 0.94, 0.81] \rightarrow [0, 0, 1, 1]$$
$$v_3 = [0.73, 1.68, -3.36, -1.85] \rightarrow [1, 1, -1, -1]$$

It can clearly be seen that the connectivity of the output units O_2 and O_3 to the hidden units H_1 and H_3 leads to logical contradictions in derived rules. Then by principle of eliminating the contradictions and redundant 0-connections the simplified structure is obtained. The simplified structure leads to the same rules for classes CLS_2 and CLS_3 as (35) and (36). The only rule which is different and slightly more complicated is the rule for class CLS_1.

$$IF\ (\bigwedge_{i=1,\,2} x_i^{(1)} \in < x_{iS}^{(1)}, x_{iE}^{(1)} > \wedge \bigwedge_{i=3,\,4} x_i^{(1)} \notin < x_{iS}^{(1)}, x_{iE}^{(1)} >)\ THEN\ CLS_1 \qquad (37)$$

By substituting the same intervals as in the previous case for the rules classifying the classes CLS_2, CLS_3, and the intervals: $x_1^{(1)} \in <4.3, 5.8>$, $x_2^{(1)} \in <2.3, 4.4>$, $x_3^{(1)} \notin <3.0, 6.0>$, $x_4^{(1)} \notin <1.0, 2.5>$, into the rule (37), it is again obtained 97.3 % correct classification.

Note that in this case the network's structure was highly redundant. The rule simplification mechanism was able to eliminate more than 57 % of unnecessary rule terms (or connections) which substantially clarified the final rules.

8. Conclusions

The article introduced a neural network based IAS that incorporates all the essential features of adaptability required by wide range of communication technologies. Novel and theoretically consistent approach allowed effective operation and dynamic interlink of modules that have been formerly treated as separate subdomains. The presented IAS is able to appropriately select learning instances, adapt its parameters and structure. Moreover, the system includes a rule extraction module that could serve as a logical interface between IAS and expert systems. IAS utilizes first order optimization techniques with superlinear convergence rates. First order optimization is sufficiently fast and computationally inexpensive - with linear computational complexities. Thanks to the speed and computational inexpensiveness of adaptable techniques the system can adapt fast even on large number of training data. Additional advantage of the presented concept is that adaptability at parametric, structural, and interface levels is simultaneous and dynamic. The kernel of the system, that is neural network, can automatically at each iteration of adaptation select different number of training exemplars, adjust the learning parameters such as learning rate and/or momentum term, and also appropriately alter the structure in order to reach the optimum learning performance. After completing the adaptation, the system is able to interpret the learned task by logical formalism such as crisp rules. This feature may have indispensable value in expert system applications.

This research was partially funded by the Hori Foundation for Promotion of Information Sciences. The authors would like to thank Dr. Naohiro Toda of Aichi Prefectural University for his valuable comments. Requests for reprints should be addressed to Professor Shiro USUI, Department of Information and Computer Sciences, Toyohashi University of Technology, Hibarigaoka, Toyohashi 441-8580, Japan.

Reviewed by: V. Olej, R. Jarina

Nomenclature

N_I	number of input units
N_0	number of output units
N_H	number of hidden units
N_F	number of free parameters of network
N_P	cardinality of training set
T	training set
x	input vector
y	output vector
E	objective (or error) function for a neural network
F	mapping of MLP network
F_{HO}	hidden-to-output submapping
F_{IH}	input-to-hidden mapping
f_j	nonlinear sigmoidal transfer function of the j-th hidden unit
$\Re$	real space
$\Re^{N_I}$	N_I dimensional real space (input space)
$\Re^{N_H}$	N_H dimensional real space (hidden space)
$\Re^{N_O}$	N_O dimensional real space (output space)
w_{kl}	hidden-to-output weight connection, from the k-th hidden unit to the l-th output unit, or input-to-output weight connection, from the k-th input unit to the l-th output unit
w	vector of hidden/input-to-output weights
bw	binary/trinary hidden/input-to-output weight vector
v_{ij}	input-to-hidden weight connection, from the i-th input unit to the j-th hidden unit
v	weight vector of input-to-hidden weights
bv	binary/trinary input-to-hidden weight vector
Θ_h	threshold weight vector of hidden units
Θ_O	threshold weight vector of output units
u	set of free parameters of neural network
$u^{(0)}$	set of free initial parameters of neural network
u_l	the l-th free parameter of a neural network
$\alpha^{(n)}$	value of the learning rate at the n-th iteration
$\beta^{(n)}$	value of the momentum term at the n-th iteration
ρ	spectral radius estimate

Literatúra - References

[1] SHANNON, C. E., WEAVER, W.: The Mathematical Theory of Communications. University of Illinois Press, University of Illinois, 1949.
[2] ARBIB, M. A. (Editor): The Handbook of Brain Theory and Neural Networks. MIT Press, Cambridge, Massachusetts, 1995.
[3] BAUM, E., HAUSSLER, D.: What size of network gives valid generalization. Neural Computation, 1(1):151–160, 1989.
[4] HWANG, J., CHOI, J. J., SEHO, O., MARKS II, R. J.: Query learning based on boundary search and gradient computation of trained multilayer perceptrons. In Proceedings of IJCNN'90, pp. 57–62, San Diego, 1990.
[5] BAUM, E. B.: Neural net algorithm that learn in polynomial time for examples and queries. IEEE Trans. on Neural Networks, 2(1):5–19, 1991.
[6] R. Battiti. Using mutual information for selecting features in supervised neural net learning. IEEE Trans. on Neural Networks, 5(4):537–550, 1994.
[7] CACHIN, C.: Pedagogical pattern selection strategies. Neural Networks, 7(1):175–181, 1994.
[8] MUNRO, P. W.: Repeat until bored: A pattern selection strategy. In J. E. Moody, S. J. Hanson, and R. P. Lippman, editors, Advances in Neural Information Processing Systems 4 (Denver), pp. 1001–1008, San Mateo, 1992. Morgan Kaufmann.
[9] FLETCHER, K.: Practical Methods of Optimization. John Wiley & Sons, Essex, 1987.
[10] WOLFE, P. Convergent conditions for ascent methods. SIAM Review, 11:226–235, 1969.
[11] POWELL, M. J. D.: A view of unconstrained optimization. In L. C. W. Dixon, editor, Optimization in Action, London, 1976. Academic Press.
[12] AL-BAALI, M., FLETCHER, R.: An ecient line search for nonlinear least squares. Journal of Optimization Theory and Application, 48(3):359–377, 1986.
[13] JACOBS, R. A.: Increasing rates of convergence through learning rate adaptation. Neural Networks, 1:295–307, 1988.
[14] T. P. Vogl, J. K. Manglis, A. K. Rigler, T. W. Zink, and D. L. Alkon. Accelerating the convergence of the back-propagation method. Biological Cybernetics, 59:257–263, 1988.
[15] PFLUG, Ch. G.: Non-asymptotic confidence bounds for stochastic approximation algorithms. Mathematic, 110:297–314, 1990.
[16] TOLLENAERE, T., SuperSAB: Fast adaptive back propagation with good scaling properties. Neural Networks, 3:561–573, 1990.
[17] DARKEN, C., MOODY, J.: Towards faster stochastic gradient search. In J. E. Moody, S. J. Hason, and R. P. Lipmann, editors, Proceedings of the Neural Information Processing Systems 4 (Denver), pp. 1009–1016, San Mateo, 1992. Morgan Kaufmann.
[18] OCHIAI, K., TODA, N., USUI, S.: Kick-Out learning algorithm to reduce the oscillation of weights. Neural Networks, 7(5):797–807, 1994.
[19] PERANTONIS, S. J., KARRAS, D. A.: An efficient constrained learning algorithm with momentum acceleration. Neural Networks, 8(2):237–249, 1995.
[20] BECKER, S., LEE CUN, Y.: Improving the convergence of back-propagation learning with second order methods. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, Proceedings of The 1988 Connectionist Models Summer School (Pittsburgh), pp. 62–72, N.Y., 1989. Wiley.
[21] BISHOP, C.: Exact calculation of the Hessian matrix for the multilayer perceptron. Neural Computation, 4(4):494–501, 1992.
[22] YU, X., LOH, N. K., MILLER, W. C.: A new acceleration technique for the backpropagation algorithm. In Proceedings of The IEEE International Conference on Neural Networks, pp. 1157–1161, San Francisco, 1993.
[23] YU, X., CHEN, G., CHENG, S.: Dynamic learning rate optimization of the backpropagation algorithm. IEEE Transactions on Neural Networks, 6(3):669–677, 1995.
[24] HINTON, G. E.: Connectionist learning procedures. Technical Report CMU-CS-87-115, Carnegie-Mellon University, 1987.
[25] WEIGEND, S. A., RUMELHART, D. E., and HUBERMANA, B. A.: Generalization by weight elimination with application to forecasting. In R. P. Lippman, J. E. Moody, and D. S. Touretzky, editors, Advances in Neural Information Processing Systems 3, pp. 875–882, San Mateo, 1991. Morgan Kaufmann.
[26] LECUN, Y., DENKER, J. S., SOLLA, S. A.: Optimal brain damage. In D. S. Touretzky, editor, Advances in Neural Information Processing Systems 2, pp. 598–605, San Mateo, 1990. Morgan Kaufmann.
[27] HASSIBI, B., STORK, D. G., WOLF, G. J.: Optimal brain surgeon and general network pruning. In IEEE International Conference on Neural Networks, pp. 293–299, San Francisco, 1993.
[28] CIBAS, T., SOULIÉ, F. F., GALLINARI, P., RANDYS, S.: Variable selection with neural networks. Neurocomputing, 12:223–248, 1996.
[29] GÉCZY, P., USUI, S.: Learning performance measures for MLP networks. In Proceedings of ICNN'97, pp. 1845–1850, Houston, 1997.
[30] GÉCZY, P., USUI, S.: Effects of structural adjustments on the estimate of spectral radius of error matrices. In Proceedings of ICNN'97, pp. 1862–1867, Houston, 1997.
[31] GÉCZY, P., USUI, S.: Effects of structural modifications of a network on Jacobean and error matrices. Submitted to Neural Networks, March 1997.
[32] TOWELL, G., SHAVLIK, J. W.: Extracting refined rules from knowledge-based neural networks. Machine Learning, 13:71–101, 1993.

[33] ANDREWS, R., DIEDERICH, J., TICKLE, A. B.: A survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems, 8:373–389, 1995.
[34] KASABOV, N.: Learning fuzzy rules and approximate reasoning in fuzzy neural networks and hybrid systems. Fuzzy Sets and Systems, 2:135–149, 1996.
[35] FU, L.: Rule generation from neural networks. IEEE Transactions on SMC, 24:1114–1124, 1994.
[36] GÉCZY, P., USUI, S.: Rule extraction from trained artificial neural networks. In Proceedings of ICONIP'97, pp.835–838, Dunedin, 1997.
[37] GÉCZY, P., USUI, S.: Fuzzy rule acquisition from trained artificial neural networks. Journal of Advanced Computational Intelligence (accepted), March 1998.
[38] GÉCZY, P., USUI, S.: Rule extraction from trained artificial neural networks. BEHAVIORMETRIKA, 26(1):89–106, 1999.
[39] GÉCZY, P., USUI, S.: Knowledge acquisition from networks of abstrast bio-neurons. In Proceedings of ICONIP'99, pp. 610–615, Perth, 1999.
[40] HORNIK, K.: Multilayer feedforward networks are universal approximators. Neural Networks, 2:359–366, 1989.
[41] MHASKAR, H. N.: Neural networks for optimal approximation of smooth and analytic functions. Neural Computation, 8:164–177, 1995.
[42] GÉCZY, P., USUI, S.: A novel dynamic sample selection algorithm for accelerated learning. Technical Report NC97-03, IEICE, pp. 189–196, March 1997.
[43] GÉCZY, P., USUI, S.: Sample selection algorithm utilizing Lipschitz continuity condition. In Proceedings of JNNS'97, pp.190–191, Kanazawa, 1997.
[44] GÉCZY, P., USUI, S.: Dynamic sample selection: Theory. IEICE Transactions on Fundamentals, E81-A(9):1931–1939, 1998.
[45] GÉCZY, P., USUI, S.: Dynamic sample selection: Implementation. IEICE Transactions on Fundamentals, E81-A(9):1940–1947, 1998.
[46] GÉCZY, P., USUI, S.: Deterministic approach to dynamic sample selection. In Proceedings of ICONIP'98, pp. 1612–1615, Kitakyushu, 1998.
[47] ARMIJO, L.: Minimization of functions having Lipschitz continuous first partial derivatives. Pacific Journal of Mathematics, 16(1):1–3, 1966.
[48] CENDROWSKA, J., Prism: An algorithm for inducing modular rules. International Journal of Man-Machine Studies, 27:349–370, 1987.
[49] FISHER, R. A.: The use of multiple measurements in taxonomic problems. Annual Eugenics,7(II):179–188, 1936.
[50] DASARATHY, B. V.: Nosing around the neighborhood: A new system structure and classification rule for recognition in partially exposed environments. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2(1):67–71, 1980.
[51] GÉCZY, P., USUI, S.: Fast back-propagation with automatically adjustable learning rate. Technical Report NC97-61, IEICE, pp. 47–54, December 1997.
[52] GÉCZY, P., USUI, S.: Superlinear and automatically adaptable conjugate gradient training algorithm. Technical Report NC97-149, IEICE, pp. 71–78, March 1998.
[53] GÉCZY, P., USUI, S.: On design of superlinear first order automatic machine learning techniques. In Proceedings of WCCI'98, pp.51–56, Anchorage, 1998.
[54] GÉCZY, P., USUI, S.: Novel first order optimization classification framework. IEICE Transactions on Fundamentals, Submitted(June), 1999.
[55] GÉCZY, P., USUI, S.: Superlinear conjugate gradient method with adaptable step length and constant momentum term. IEICE Transactions on Fundamentals, Submitted(June), 1999.
[56] GÉCZY, P., USUI, S.: Universal superlinear learning algorithm design. IEEE Transactions on Neural Networks, Submitted(February), 1999.
[57] FLETCHER, R., REEVES, C. M.: Function minimization by conjugate gradients. Comput. Journal, 7:149–154, 1964.
[58] WNEK, J., MICHALSKI, R. S.: Comparing symbolic and subsymbolic learning: Three studies. In R. S. Michalski and G. Tecuci, editors, Machine Learning: A Multistrategy Approach, volume 4, San Mateo, 1993. Morgan Kaufmann.
[59] ALEFELD, G., HERZBERGER, J.: Introduction to Interval Computations. Academic Press, New York, 1983.
[60] DUDA, R. O., HART, P E.: Pattern Classification and Scene Analysis. John Wiley & Sons, 1973.

https://doi.org/10.26552/com.C.2000.2.56-60

komunikácie
COMMUNICATIONS

Oldřich Kovář - Juraj Miček *

ČÍSLICOVÁ KONCEPCIA SIGMA- DELTA Č/A PREVODNÍKA IMPLEMENTOVANÁ DO FPGA

THE DIGITAL CONCEPTION OF THE SIGMA - DELTA CONVERTER IMPLEMENTED INTO FPGA

Č/A prevodníky sú technické zariadenia, ktoré umožňujú vzájomnú komunikáciu medzi diskrétnymi a spojitými systémami. Ich základná funkcia je výstižne obsiahnutá vo výroku „Č/A prevodníky tvoria most medzi číslicovým a analógovým svetom".

Č/A prevodníky transformujú diskrétny signál, postupnosť binárnych čísiel na analógový ekvivalent. Len veľmi ťažko si dnes vieme predstaviť realizáciu číslicových riadiacich systémov a niektoré aplikácie číslicového spracovania signálov bez ich využitia.

Autori príspevku predkladajú číslicovú koncepciu sigma- delta Č/A prevodníka, ktorý je kompletne implementovaný do programovateľného logického obvodu typu FPGA XC 4005E. Predkladaná koncepcia sigma-delta Č/A prevodníka vyžaduje minimálne prídavné obvody, dolnopriepustný filter, pozostávajúci iba z jedného rezistora a kapacitora.

Kľúčové slová: *SD Č/A prevodník, FPGA, dolnopriepustný filter, tvarovací obvod 0-tého rádu.*

D/A converters are technical means which allow mutual communication between discrete and analog systems. Then base function is accurately comprehended in the statement "D/A converters create bridge between digital and analog world".

D/A converter transforms the discrete signal, progression of the binary numbers to the analog equivalent.

Nowadays it is difficult to imagine realization of the digital control systems and some applications of the digital signal processing without using them.

The authors of the article describe the digital conception of the Sigma - Delta D/A converter which is completely implemented into a programmable logical device FPGA, XC 4005E. The described conception of the Sigma -Delta D/A converter requires minimum of the external components, only one resistor and one capacitor as a lowpass filter.

Keywords: *SD D/A Converter, FPGA, Lowpass Filter, Zero Order Hold Circuit-S/H.*

Úvod

Proces Č/A prevodu je možné opísať vzťahom medzi postupnosťou vstupných vzoriek diskrétneho signálu $x(n)$ a zodpovedajúcim spojitým signálom $x(t)$. Ak predpokladáme, že požadovaný spojitý signál je frekvenčne ohraničený frekvenciou f_{max}, potom je ho možné získať z postupnosti vzoriek $x(nT_S)$ na základe známej Shannonovej interpolačnej formuly:

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_S) \frac{\sin\left(\frac{\pi}{T_S}(t - nT_S)\right)}{\frac{\pi}{T_S}(t - nT_S)} \tag{1}$$

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_S)\, g(t - nT_S), \tag{2}$$

pričom interpolačná funkcia $g(t)$ je v tvare:

$$g(t) = \frac{\sin\left(\frac{\pi}{T_S}t\right)}{\frac{\pi}{T_S}t} = \frac{\sin(2\pi f_{max}t)}{2\pi f_{max}t}, \tag{3}$$

kde $f_{max} = f_S/2 = 1/2T_S$

Introduction

The Process of D/A conversion can be described by the relation between the progression input samples $x(n)$ and corresponding analog signal $x(t)$. If we suppose that the required analog signal is limited by the frequency f_{max}, then we take it from the progression of the samples $x(nT_S)$ on the base of the well known the Shannon interpolation formula:

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_S) \frac{\sin\left(\frac{\pi}{T_S}(t - nT_S)\right)}{\frac{\pi}{T_S}(t - nT_S)} \tag{1}$$

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_S)\, g(t - nT_S), \tag{2}$$

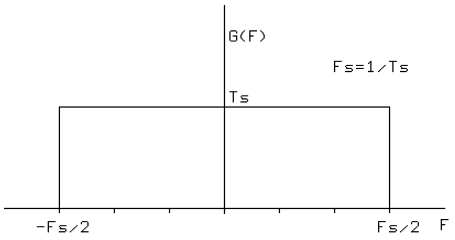
whereby interpolation function $g(t)$ is in the form:

$$g(t) = \frac{\sin\left(\frac{\pi}{T_S}t\right)}{\frac{\pi}{T_S}t} = \frac{\sin(2\pi f_{max}t)}{2\pi f_{max}t}, \tag{3}$$

where $f_{max} = f_S/2 = 1/2T_S$

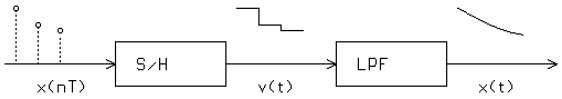
* **Ing. Oldřich Kovář, PhD., Doc. Ing. Juraj Miček, PhD.**
Department of Technical Cybernetics, Faculty of Management Sciences and Informatics, University of Žilina, Veľký diel, SK-01026 Žilina, Slovak Republic, Tel: +421-89-5254042, E-mail: kovar@frtk.utc.sk, micek@frtk.utc.sk

Proces ideálneho Č/A prevodu je možné chápať ako proces filtrácie, pri ktorom sa snažíme potlačiť všetky frekvenčné zložky diskrétneho signálu ležiace mimo frekvenčného rozsahu $-f_{max}$ až f_{max}. Interpolačná funkcia $g(t)$ potom prestavuje impulznú odozvu dolnopriepustného filtra s frekvenčnou charakteristikou uvedenou na obr. 1.



Obr. 1 Frekvenčná charakteristika ideálneho dolnopriepustného filtra
Fig. 1 Frequency characteristic of the ideal low pass filter

Poznamenajme, že v uvedenom prípade sa stretávame s nekauzálnym IIR filtrom, ktorého priama realizácia nie je možná. V praktických aplikáciách sa preto pri realizácii Č/A prevodníkov stretávame s jednoduchšími prístupmi. Najčastejšie sa používajú Č/A prevodníky s tvarovačom 0-tého rádu. Výstup tvarovacieho obvodu (S/H) je ďalej upravený dolnopriepustným filtrom. Štruktúra Č/A prevodníka je uvedená na obr. 2.



Obr. 2 Štruktúra Č/A prevodníka
Fig.2 Structure of the D/A converter

V uvedenom prípade je možné impulznú odozvu tvarovacieho obvodu vyjadriť nasledovne:

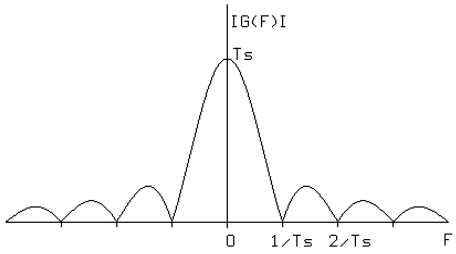
$$g(t) = \begin{matrix} 1 & 0 \le t < T \\ 0 & inak \end{matrix} \qquad (4)$$

Frekvenčnú odozvu tvarovacieho člena je možné určiť z impulznej odozvy aplikovaním Fourierovej transformácie:

$$G(F) = \int_{-\infty}^{\infty} g(t) e^{-j\pi Ft}\, dt = T_S \frac{\sin \pi F T_S}{\pi F T_S} e^{-j\pi FTS} \qquad (5)$$

Amplitúdová frekvenčná charakteristika je znázornená na obr. 3

Z obrázka je zrejmé, že rozdiel medzi frekvenčnou charakteristikou ideálneho Č/A prevodníka - obr. 1 a Č/A prevodníka realizovaného na báze tvarovacieho obvodu 0-tého rádu je pomerne značný. Prítomnosť vyšších nežiaducich frekvenčných zložiek v spojitom signále je daná zvlnením frekvenčnej charakteristiky v nepriepustnom pásme ($F > 1/2T_S$). Tieto nežiaduce frekvenčné zložky je možné potlačiť pomocou analógového dolnopriepustného filtra.



Obr. 3 Amplitúdová frekvenčná charakteristika tvarovača 0-tého rádu
Fig. 3 Amplitude frequency characteristic of the S/H.

Pri návrhu a realizácii Č/A prevodníkov na báze tvarovacích členov sa využívajú i tvarovače vyšších rádov, ktoré však nepriná-šajú podstatné zlepšenie vlastností Č/A prevodníkov. Jedným zo zaujímavých prístupov k realizácii Č/A prevodníkov je prevodník, ktorý využíva princíp sigma-delta modulácie.

Process of an ideal conversion can be understood as a process of the filtering in which we try to supress all frequency components out of the frequency range $-f_{max}$ to f_{max}.

Interpolation function $g(t)$ then represents the impulse response of the lowpass filter with frequency characteristic depicted in Fig. 1.

In this case we use a non causal IIR filter whose direct realization is impossible. In the design and realization of the D/A converters we have more simple approaches to the practical application. D/A converters with a zero order hold circuit (S/H) are frequently used. Output of the S/H is further formed by the lowpass filter. The structure of the D/A converter is depicted in Fig.2.

It is possible to express the impulse response of the S/H by the following:

$$g(t) = \begin{matrix} 1 & when\ 0 \le t < T \\ 0 & other \end{matrix} \qquad (4)$$

It is possible to determine the frequency response of the S/H from the impulse response by the application of Fourier transformation:

$$G(F) = \int_{-\infty}^{\infty} g(t) e^{-j\pi Ft}\, dt = T_S \frac{\sin \pi F T_S}{\pi F T_S} e^{-j\pi FTS} \qquad (5)$$

The amplitude frequency characteristic is depicted in Fig. 3.

The difference between the frequency characteristic of an ideal D/A converter (Fig. 1) and a D/A converter realized on the base of the S/H is quite significant.

The presence of the higher frequency components contained in the analog signal is given by the ripple of the frequency characteristic above the frequency $F > 1/2T_S$ (stop band). These undesirable higher frequency components can be eliminated by an analog lowpass filter.

In the design and realization of the D/A converters on the base of the S/H , higher order hold circuits are used, but without a remarkable improvement of the qualities of the D/A converters.

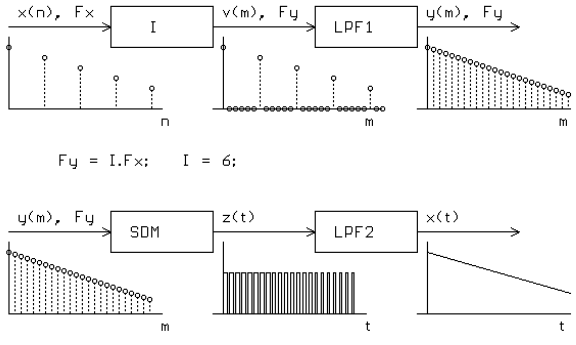
One interesting approach to the design and realization of the D/A converters is a converter which uses the principle of the sigma - delta modulation.

Č/A prevodník na princípe Sigma-Delta Modulátora

Sigma-delta modulátor transformuje postupnosť vzoriek $\{y(m)\}$ na dvojhodnotový signál $z(t)$. Dvojhodnotový signál $z(t)$ je privedený na analógový dolnopriepustný filter, ktorý potláča vplyv nežiaducich frekvenčných zložiek. Princíp činnosti je znázornený na obr. 4.

SD Č/A prevodník obsahuje:
- blok interpolácie I
- číslicový dolnopriepustný filter LPF1
- sigma-delta modulátor SDM
- analógový dolnopriepustný filter LPF2.

Blok interpolácie realizuje I-násobné zvýšenie vzorkovacej frekvencie. Transformuje vstupnú postupnosť $\{x(n)\}$ na postupnosť $\{v(m)\}$ tak, že medzi dve po sebe idúce vzorky $x(n)$ a $x(n+1)$ doplní $I-1$ nulových hodnôt podľa vzťahu:



Obr. 4 Štruktúra Č/A SD prevodníka
Fig. 4 Structure of the SD D/A converter

$$v(m) = \begin{matrix} x(n/I) & m = 0, \pm I, \pm 2I, \ldots \\ 0 & inak \end{matrix} \qquad (6)$$

Postupnosť $\{v(m)\}$ vstupuje do číslicového dolnopriepustného filtra LPF1, ktorý potlačí frekvenčné zložky nad frekvenciou π/I. Výstup z dolnopriepustného filtra, postupnosť $\{y(m)\}$, je spracovaná v sigma-delta modulátore na dvojhodnotový časovo spojitý signál $z(t)$. Signál $z(t)$ je upravený prostredníctvom analógového dolnopriepustného filtra LPF2 na spojitý ekvivalent diskrétneho signálu $x(n)$. Poznamenajme, že číslicový dolnopriepustný filter pracuje so vzorkovacou frekvenciou I-násobne vyššou než je vzorkovacia frekvencia vstupného diskrétneho signálu $x(n)$. Z dôvodov technickej realizovateľnosti sa preto snažíme o čo najjednoduchšiu implementáciu uvedeného filtra. V nasledujúcej časti je uvedené riešenie Č/A SD prevodníka, v ktorom interpolačný filter je realizovaný ako tvarovač 0-tého rádu.

Architektúra číslicového sigma-delta Č/A prevodníka

Prezentovaný SD Č/A prevodník je navrhnutý ako číslicový systém implementovaný do reprogramovateľného obvodu FPGA XC 4005E. Bloková schéma je uvedená na obr. 5.

Vstupné a výstupné signály:
DAC_{OUT} - výstupný signál, sled impulzov, ktorý je spracovávaný dolnopriepustným filtrom
DAC_{IN} - vstupná binárna zbernica.
CLK - vstupný taktovací signál. Jeho nábežnou hranou je riadený pracovný cyklus prevodníka
Reset - inicializačný signál

D/A Converter on the base of the S-D modulatro

Sigma - delta modulator transforms progression of the samples $\{y(m)\}$ into bi-level signal $z(t)$. The bi-level signal $z(t)$ goes through an analog lowpass filter which eliminates the influence of undesirable frequency components. The principle of the function is depicted in Fig. 4.

SD D/A converter contains:
- block of interpolation I
- digital lowpass filter LPF1
- sigma-delta modulator SDM
- analog lowpass filter LPF2

Block of the interpolation realizes I-times raising of the sampling frequency. It transforms input progression $\{x(n)\}$ to output progression $\{v(m)\}$ by loading $I-1$ zeroes between two successive samples $x(n)$ a $x(n+1)$ according to the formula:

$$v(m) = \begin{matrix} x(n/I) & m = 0, \pm I, \pm 2I, \ldots \\ 0 & other \end{matrix} \qquad (6)$$

Progression $\{v(m)\}$ inputs into digital lowpass filter LPF1 which eliminates frequency components above the frequency π/I. The output signal of the lowpass filter, progression $\{y(m)\}$, is modulated by the sigma-delta modulator to the bi-level time continuous signal $z(t)$. The signal $z(t)$ is being modified by the analog lowpass filter LPF2 to the form of the continuous equivalent of the discrete signal $x(n)$. We notice that a digital lowpass filter works with I-times higher the sampling frequency than is sampling frequency of the discrete input signal $x(n)$. We endeavour to find the most simple implementation of the filter because of having in mind the technical realization.

The solving of the SD D/A converter is described in the following part of the article. The SD D/A converter is designed and implemented as a pure digital system with a zero order hold circuit.

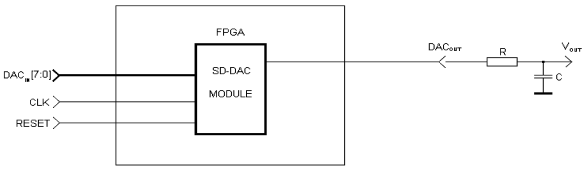
Architecture of the digital sigma - delta D/A converter

The herein presented SD D/A converter is designed as a digital system implemented in a reprogrammable logic device, Field Programmable Gate Array (FPGA) XC 4005E. The block scheme of the implemented system with interface signals is depicted in Fig. 5.

where:
DAC_{OUT} -output signal, pulse string, that drives the external lowpass filter
DAC_{IN} - digital input bus
CLK - system input clocks
Reset - initializes system

Číslicová koncepcia SD Č/A prevodníka podstatne eliminuje teplotné vplyvy na jeho funkciu.

Bloková štruktúra SD Č/A prevodníka je uvedená na obr. 6.

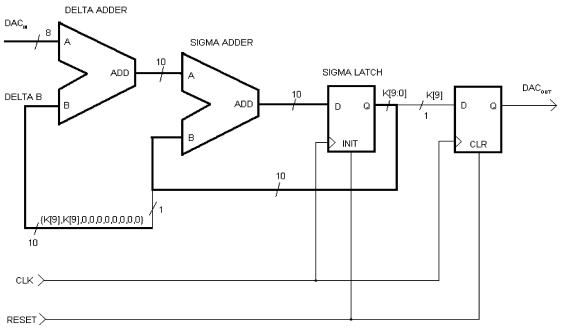


Obr. 5 Bloková schéma SD Č/A prevodníka
Fig. 5 Block scheme of the SD D/A converter

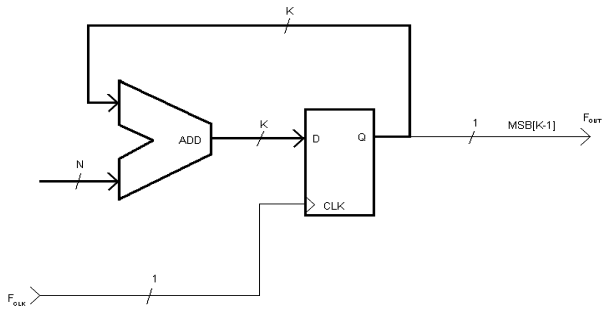
Jadro SD Č/A prevodníka vytvárajú sumátor Sigma Adder a register Sigma Latch, ktoré v uvedenom funkčnom spojení predstavujú systém priamej číslicovej frekvenčnej syntézy *DDFS (Direct Digital Frequency Synthesis)*. Jej všeobecná bloková schéma je uvedená na obr. 7

The digital conception of the SD A/D converter significantly eliminates the influence of temperature on its function. The structure of the SD A/D converter is depicted in Fig. 6.

The core of the SD D/A converter consists of the Sigma Adder and Sigma Latch register, which, in the described function connection works as a *Direct Digital Frequency Synthesis (DDFS)*. Its common block diagram is depicted in Fig. 7.



Obr. 6 Bloková štruktúra SD Č/A prevodníka
Fig. 6 Structure of te Sigma- Delta D/A converter



Obr. 7 Bloková schéma priamej číslicovej frekvenčnej syntézy
Fig. 7 The block diagram of the DDFS

Ako je zrejmé z obr. 7, systém priamej číslicovej frekvenčnej syntézy je realizovaný akumulátorom.

Frekvencia výstupného signálu (bitu MSB registra) systému priamej číslicovej syntézy je určovaná binárnou hodnotou konštanty N, ktorá je privedená na jeden zo vstupov sumátora podľa nasledovného vzťahu (7):

$$F_{OUT} = F_{CLK} \cdot \frac{N}{2^k} \tag{7}$$

kde k je počet bitov spätnovezbovej zbernice
N je riadiaca binárna konštanta na vstupe sumátora
F_{CLK} je frekvencia taktovacích hodín systému

Ako je zo vzťahu (7), zrejmé frekvenciu výstupného signálu je možné meniť s krokom 2^{-k}.

Frekvenčné obmedzenie výstupného signálu je určené vzťahom (8):

$$F_{OUT} < \frac{F_{CLK}}{2} \tag{8}$$

Funkčné parametre číslicového SD Č/A prevodníka

Opisovaný SD Č/A prevodník je komplexne navrhnutý ako číslicový systém a implementovaný do reprogramovateľného obvodu typu FPGA bez použitia prídavných externých diskrétnych súčiastok. Tým je zabezpečená značná tepelná stabilita SD Č/A prevodníka a odpadajú problémy s presnosťou diskrétnych súčiastok a stálosťou ich parametrov.

As we can see in Fig. 7 the system DDFS is, in fact, an accumulator. The frequency of the output signal (bit MSB of the Sigma Latch register) of the system DDFS is controlled by the value of the binary constant N placed on one of the inputs of the summator. The frequency FOUT is given by the following equation:

$$F_{OUT} = F_{CLK} \cdot \frac{N}{2^k} \tag{7}$$

where k is a number of bits of the feedback bus
N is a control binary constant in the input of the summator
F_{CLK} is frequency of the systems clock

The frequency of the output signal can be changed with a resolution step 2^{-k}.

There is a limitation of the frequency of the output signal according to the following equation (8):

$$F_{OUT} < \frac{F_{CLK}}{2} \tag{8}$$

Functional parameters of the digital SD D/A converter

The described SD D/A converter is completely designed as a digital system and is implemented in FPGA XC 4005E. The advantages of using FPGA are: no dependence on temperature, voltage or aging and accuracy of the external analog components.

Výstupné napätie SD Č/A prevodníka

Pre implementovanú číslicovú štruktúru SD Č/A prevodníka podľa obr. 6 je možné výstupné napätie prevodníka vyjadriť ako funkciu hodnoty vstupnej binárnej vzorky prevodníka (DAC_{IN}) podľa nasledujúceho vzťahu

$$V_{OUT} = \frac{\langle DAC_{IN} \rangle}{2^{MSBI+1}} \cdot V_{cco} \quad [\mathrm{V}] \tag{9}$$

kde V_{OUT} je výstupné napätie za dolnopriepustným filtrom, analógový ekvivalent binárnej hodnoty vstupnej vzorky prevodníka
$\langle DAC_{IN} \rangle$ binárna hodnota vstupnej vzorky prevodníka
$MSBI$ najvyšší váhový bit vstupnej vzorky

Interval prevodu

Interval prevodu SD Č/A prevodníka je v tomto prípade určený vzťahom

$$T_P = 2^{MSBI+1} \cdot T_{CLK} \tag{10}$$

kde T_{CLK} perióda taktovacích hodín systému

V prípade 8-bitovej vstupnej vzorky a taktovacej frekvencie 50 MHz je v našom prípade čas prevodu 5,12 μs.

Uvedený číslicový SD Č/A prevodník má univerzálne použitie v oblastiach, v ktorých vyhovuje s hľadiska rýchlosti prevodu. Môžu to byť napríklad nasledovné aplikácie:

- programovateľný generátor napätia
- generátor periodických priebehov
- generátor zvuku
- RGB generátor

Záver

Popísaný Sigma Delta Č/A prevodník je jedným z príkladov efektívneho využitia programovateľných logických obvodov typu FPGA v oblastiach, ktoré boli donedávna doménou analógových obvodov. Hustota integrácie a rýchlosť dnešných FPGA obvodov dovoľuje implementáciu veľmi rozsiahlych číslicových štruktúr pracujúcich frekvenciou až 350 MHz. Nezanedbateľným aspektom uvedenej koncepcie SD Č/A prevodníka je minimalizácia počtu stavebných komponentov s pozitívnymi dôsledkami na zvýšenie spoľahlivosti systému.

Recenzenti: J. Mintal, M. Hrianka

The output voltage of the SD D/A converter

For the implementation in Fig. 6, the output voltage V_{OUT} as a function of the SD D/A converter input may be expressed as follows:

$$V_{OUT} = \frac{\langle DAC_{IN} \rangle}{2^{MSBI+1}} \cdot V_{cco} \quad [\mathrm{V}] \tag{9}$$

where: V_{OUT} is an output voltage on the lowpass filter output, analog equivalent of the binary value of the input sample
$\langle DAC_{IN} \rangle$ contents of the input bus
$MSBI$ the most significant bit of the input sample

The time of the conversion

The time of the conversion of the SD D/A converter is in this case expressed as follows:

$$T_P = 2^{MSBI+1} \cdot T_{CLK} \tag{10}$$

Where T_{CLK} is a period of the system clock.

In the case of an 8-bit input sample and CLK frequency 50 MHz is a time of the conversion 5.12 μs.

The presented SD D/A converter has universal applications in the areas where it satisfies the criteria of the conversion speed. The following applications can be introduced:

- Programmable Voltage Generator
- Waveform Generator
- Sound Generator
- RGB Color Generator

Conclusion

Digital Sigma -Delta D/A converter is one of the examples how the reprogrammable logic devices FPGAs can be effectively used in applications, where analog circuits dominated until recently. The density and speed of today's FPGAs circuits make them ideal for implementation of a wide range of very vast digital systems working on the frequency up to 350MHz. The unneglectible aspect of the introduced conception of the SD D/A converter is a reduction of the system components with direct consequences on the raising of the system reliability.

Reviewed by: J. Mintal, M. Hrianka

Literatúra - References

[1] PROAKIS, J.G., MANLOAKIS, D.G.: Digital signal processing, MPC, New York 1992
[2] The programmable logic Data Book, Žilina, 1999
[3] XCELL Journal, Issue 31, 1999
[4] Analog Devices Data Converter Reference Manual, Volume I, 1992

https://doi.org/10.26552/com.C.2000.2.61-63

Peter Farkaš - Sergio Herrera-Garcia *

DVA NOVÉ NAJLEPŠIE KÓDY [27,10,9]

TWO NEW BEST [27,10,9] CODES

V tomto príspevku sú zverejnené dva nové lineárne binárne blokové kódy [27,10,9], ktoré dosahujú hornú hranicu pre kódovú vzdialenosť najlepších kódov [1]. Nové kódy sa vyznačujú rozdielnymi váhovými spektrami v porovnaní s doteraz zverejnenými kódmi [27,10,9] skonštruovanými Piretom [2], Farkašom a Julingom [5].

Kľúčové slová: *Blokový kód, Hammingova vzdialenosť, váhové spektrum, generujúca matica.*

1. Úvod

Otázka existencie dobrých samoopravných kódov patrí k najdôležitejším otázkam *Teórie komunikácie.* Označme *n* dĺžku kódového slova, *k* počet informačných symbolov v kódovom slove a *d* minimálnu Hammingovu vzdialenosť medzi dvoma kódovými slovami v kóde - *kódovú vzdialenosť.* Uvedené základné hodnoty sa štandardne zapisujú vo forme [*n*, *k*, *d*]. Ak majú dva kódy rovnaké hodnoty *n* a *k*, tak za lepší kód sa považuje ten, ktorý má väčšiu hodnotu *d*. Je to preto, že takýto kód dokáže pri tej istej nadbytočnosti a dĺžke kódového slova ochrániť lepšie prenášanú alebo uchovávanú informáciu. Dolná a horná hranica pre kódovú vzdialenosť ako v závislosti od *n* a *k* - *d*(*n*,*k*), sa dá nájsť pre niektoré hodnoty v [1].

Interaktívny prístup k dolným a horným hraniciam *d*(*n*,*k*) pre binárne a nebinárne kódy je možný prostredníctvom: *http://www.win.tue.nl/win/math/dw/voorlicod.html*

V nasledujúcej časti uvedieme generujúce matice a váhové spektrá známeho Piretovho kódu a dvoch nových [27,10,9] kódov, ktoré dosahujú hornú hranicu pre d(n,k) v [1]. V závere sú uvedené niektoré poznámky.

2. Nové kódy

P. Piret v [2] skonštruoval prvý [27,10,9] kód. Nám sa podarilo nájsť iné [27,10,9] kódy. Naša metóda hľadania bola založená na jednej modifikácii známeho postupu z [3], ktorá je opísaná v [4]. Pre vyhľadávanie bol využitý superpočítač Fujitsu VP 2600/20.

Na dôkaz, že kódy, ktoré sa vyznačujú totožnými parametrami *n*, *k*, *d* nie sú zhodné je potrebné určiť ich váhové spektrá. Váhové spektrum je možné opísať ako množinu konštánt a_i, ktoré udávajú počet slov s Hammingovou váhou *i*. Hammingova váha

This paper presents two new [27,10,9] binary linear block codes which reach the upper bound in [1] for best known codes. They have different spectra as the known [27,10,9] codes constructed by Piret [2], Farkaš and Juling [5].

Keywords: *Block codes, Hamming distance, weight spectrum, generator matrix.*

1. Introduction

The question concerning the existence of good error control codes belongs to the most important in Communication theory. Let us denote *n* the codeword length, *k* the number of information symbols and *d* the Hamming-distance of a code. If two codes have the same value n and identical value k the code with greater value d is classified as better, because this code has greater error control capability by the same redundancy.

The lower and upper bounds on *d*(*n*,*k*), the maximum possible Hamming-distance of some linear binary block codes can be found in table in [1].

An interactive interface to the lower and upper bounds on *d*(*n*,*k*) of some linear binary, ternary and quadruple block codes, can be found on: *http://www.win.tue.nl/win/math/dw/voorlicod.html*

In the following section we present generator matrices and weight spectra of the known and two new [27,10,9] codes, which reach the upper bound for d(n,k) in [1]. In conclusion we give some remarks.

2. New Codes

Piret [2] has constructed the first [27,10,9] Code [2]. We found other [27,10,9] codes. Our searching method was based on one variation of the known algorithm [3], described in [4]. The search was realized on the super computer Fujitsu VP 2600/20.

To show that codes with the same parameters *n*, *k*, *d* aren't equivalent, it is necessary to find their weight spectra.

The code weight spectrum can be given by a set of constants a_i which represent the number of code words with the Hamming-

* **Peter Farkaš[1], Sergio Herrera-Garcia[2]**
[1]Department of Telecommunications, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, SK-812 19 Bratislava, Slovak Republic, E-mail: p.farkas@ieee.org
[2]CITEDI-IPN Research Center and CONACyT 2498 Roll Dr. # 757 Otay Mesa, San Diego, CA 92154

kódového slova je definovaná ako počet nenulových súradníc v tomto kódovom slove. Kódy [*n*,*k*,*d*] s rôznymi váhovými spektrami sú rozdielne.

Piretov kód [27,10,9] z [2] má nasledujúcu generujúcu maticu a ako váhové spektrum:

weight *i*. The Hamming weight of a code word from a linear binary code is equal to the number of ones in this code word. The [*n*,*k*,*d*] codes with different weight spectra are not equivalent.

The [27,10,9] Piret code [2] has the following generator matrix and weight spectrum:

$$\mathbf{G} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

Váhové spektrum kódu:

$a_0 = 1$ $a_9 = 56$ $a_{10} = 99$ $a_{11} = 90$ $a_{12} = 129$
$a_{13} = 144$ $a_{14} = 126$ $a_{15} = 132$ $a_{16} = 117$ $a_{17} = 72$
$a_{18} = 31$ $a_{19} = 18$ $a_{20} = 9$

Weight spectrum of the code:

$a_0 = 1$ $a_9 = 56$ $a_{10} = 99$ $a_{11} = 90$ $a_{12} = 129$
$a_{13} = 144$ $a_{14} = 126$ $a_{15} = 132$ $a_{16} = 117$ $a_{17} = 72$
$a_{18} = 31$ $a_{19} = 18$ $a_{20} = 9$

Nové kódy [27,10,9] majú nasledujúce generujúce matice a spektrá:

The new [27,10,9] codes have the following generator matrices

1.

1.

$$\mathbf{G} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0
\end{bmatrix}$$

váhové spektrum kódu:

$a_0 = 1$ $a_9 = 54$ $a_{10} = 99$ $a_{11} = 99$ $a_{12} = 128$
$a_{13} = 129$ $a_{14} = 131$ $a_{15} = 142$ $a_{16} = 107$ $a_{17} = 72$
$a_{18} = 41$ $a_{19} = 15$ $a_{20} = 4$ $a_{21} = 1$ $a_{22} = 1$

and weight spectra:

$a_0 = 1$ $a_9 = 54$ $a_{10} = 99$ $a_{11} = 99$ $a_{12} = 128$
$a_{13} = 129$ $a_{14} = 131$ $a_{15} = 142$ $a_{16} = 107$ $a_{17} = 72$
$a_{18} = 41$ $a_{19} = 15$ $a_{20} = 4$ $a_{21} = 1$ $a_{22} = 1$

2.

$$\mathbf{G} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0
\end{bmatrix}$$

váhové spektru kódu:

$a_0 = 1$ $a_9 = 53$ $a_{10} = 100$ $a_{11} = 104$ $a_{12} = 124$
$a_{13} = 119$ $a_{14} = 136$ $a_{15} = 152$ $a_{16} = 107$ $a_{17} = 67$
$a_{18} = 36$ $a_{19} = 16$ $a_{20} = 8$ $a_{21} = 1$

3. Záver

Je známe, že z kódov, ktoré sme našli a zverejnili v tomto článku, je možné skonštruovať dva ďalšie optimálne kódy [27,10,10] pridaním jedného paritného bitu ku každému kódovému slovu. Doteraz bolo publikovaných deväť kódov, ktoré dosahujú hranicu d (27,10) = 9. Otázka, či existujú ďalšie optimálne kódy s rovnkými parametrami, ostáva otvorená.

Poďakovanie

Autori chcú vyjadriť vďaku CITEDI-IPN Research Center and CONACyT Mexico, ďalej prof. Dieterovi Hauptovi z RWTH Aachen a DAAD za podporu tejto práce a Dr. Klausovi Bruehlovi z Výpočtového strediska RWTH Aachen za napísanie programov potrebných k vyhľadávaniu kódov.

Recenzenti: P. Krajčí, M. Brezňan

2.

and weight spectra:

$a_0 = 1$ $a_9 = 53$ $a_{10} = 100$ $a_{11} = 104$ $a_{12} = 124$
$a_{13} = 119$ $a_{14} = 136$ $a_{15} = 152$ $a_{16} = 107$ $a_{17} = 67$
$a_{18} = 36$ $a_{19} = 16$ $a_{20} = 8$ $a_{21} = 1$

3. Conclusion

It is known that from the codes we found two other new [27,10,10] codes could be constructed by adding overall parity check to each code word. This paper shows, that minimum nine linear binary codes exist, which reach the lower bound d (27,10) = 9. We cannot guarantee, that we found all linear binary [27,10,9] codes and therefore the question how many such codes exist remains open.

Acknowledgement

The authors want to express their thanks to CITEDI-IPN Research Center and CONACyT Mexico and as well to Prof. Dieter Haupt RWTH Aachen and the DAAD for supporting this work and Dr. Klaus Bruehl from the computing center of RWTH Aachen for writing the software needed for the search.

Reviewed by: P. Krajčí, M. Brezňan

Literatúra - References

[1] BROUWER, A. E., VERHOEFF, T.: An Updated Table of Minimum - Distance Bounds for Binary Linear Code. In: IEEE Trans. Inform. Theory, IT-39, 1993, no. 2, pp. 662-677.
[2] PIRET, P.: Good linear codes of lengths 27 and 28. In: IEEE Trans. on Inform. Theory, IT-26, 1980, pp. 227.
[3] FARKAŠ, P., SMIRNOV, A. S., SOTSKOV, J. V.: Lineárne kódy opravujúce mnohonásobné chyby. In: Informačné systémy, vol. 14, 1986, no. 5, pp. 533-542.
[4 FARKAŠ, P., BRUEHL, K.: Three Best Binary Linear Block Codes of Minimum Distance Fifteen. In: IEEE Trans. on Inform. Theory, IT-40, 1994, no. 3, pp. 949-951.
[5] FARKAŠ, P, JULING W.: Five New Best [27,10,9] Codes. In: International Journal of Electronics and Communications, (AEU), vol. 48, 1994, no. 2, pp. 115-116.

https://doi.org/10.26552/com.C.2000.2.64-70

Csaba Stupák - Rastislav Lukáč - Stanislav Marchevský *

PROCESOR PRE POTLÁČANIE IMPULZOVÉHO ŠUMU V TELEKOMUNIKAČNÝCH KANÁLOCH

PROCESSOR FOR IMPULSIVE NOISE SUPPRESSION IN TELECOMMUNICATION CHANNELS

V ostatnom čase bolo mnoho lineárnych filtrov nahradených nelineárnymi filtrami, obzvlášť v prípade poškodenia signálu impulzovým šumom. Trieda kompozičných filtrov, ktorá patrí medzi nelineárne filtre, je zaujímavá kvôli ľahkej hardvérovej realizácii a vlastnosti nakladania. Rozvoj digitálnej HDTV podporil technologický vývoj v tejto oblasti. V mnohých aplikáciách je vhodné použitie veľmi účinných poriadkovo-štatistických filtrov, napr. mediánového alebo vyhladzovacieho LUM filtra. Tieto filtre môžu byť realizované ako kompozičné filtre, ktorými sa zaoberáme v tomto krátkom prehľade.

In recent years, many linear filters have been replaced by non-linear filters, especially in case of impulse noise distorting. Stack filters, the class of non-linear filters are very interesting for easy hardware realisation and threshold decomposition property. The development of digital HDTV gives technological push in this area. Various of very efficient order-statistics filters such as median or LUM smoother are useful in many applications. In addition, these filters can be realised as stack filters. This paper gives a short review of the stack filter class.

1. Úvod

Väčšina súčasných komunikačných systémov, systémov riadenia a systémov pre spracovanie signálov, je vyvíjaných so zreteľom na odolnosť voči Gaussovmu šumu [2]. Avšak mnohé prostredia sú vhodnejšie modelované ako impulzové, nie gaussové distribúcie [16-18]. V praxi impulzový šum pochádza z atmosférického rušenia, napr. atmosférické výboje, vyžarovanie v rádiokomunikáciách a šum spínacích relé v telefónnych kanáloch. Okrem týchto prirodzených negausovských zdrojov, existuje aj veľké množstvo umelých zdrojov ako automatické vzplanutie, neónové svetlá a iné elektronické zariadenia.

Impulzový šum je vysoko závislý od fyzikálneho prostredia a môže sa relatívne zriedkavo vyskytovať vzhľadom na nestacionaritu a nevhodný štatistický popis.

Nie Gaussova podstata predurčuje na potlačenie impulzového šumu nelineárne filtre s vlastnosťou robustnosti. Pre tieto vlastnosti sa široko používajú kompozičné filtre.

Zvyšok tohto článku je organizovaný nasledovne. V prvej časti sú popísané modely impulzového šumu. Základné vlastnosti procesora sú načrtnuté v druhej časti. V ďalších častiach je opísaná dvojrozmerná mediánová filtrácia, vyhladzovacie LUM filtre, podstata kompozičných filtrov a neurónové kompozičné filtre. V závere sú zhrnuté základné myšlienky.

1. Introduction

The majority of the present systems in communication, control and signal processing are developed under the assumption that the interfering noise is Gaussian [2]. However, many physical environments are more accurately modelled as impulsive non-Gaussian distributions [16-18]. In practice, sources of impulse noise include atmospheric noise, such as lightning spikes and spurious radio emission in radio communication, and relay switching noise in telephone channels. In addition to these natural non-Gaussian noise sources, there is a great variety of man-made sources such as automatic ignition, neon lights, and other electronic devices.

Impulse noise is highly dependent on the physical environment and may be relatively infrequently occurring and non-stationary, which often renders it impossible to obtain an accurate statistical description. The non-Gaussian nature of impulse noise dictates that the suppression filter should be non-linear, and due to the presence of impulses, it must be robust. For these properties, the stack filters are widely used.

The paper is organized as follows. In the first section, impulsive noise types are described. The second section outlines the basic principle of the processor. The following sections describe the two-dimensional median filtering, LUM smoothers, fundamentals of stack filters and neural stack filters. Conclusions are drawn in the last section.

* **Csaba Stupák, Rastislav Lukáč, Stanislav Marchevský**
Technical University of Košice, Faculty of Electronics and Informatics, Department of Electronics and Multimedial Communications, Park Komenského 13, SK-04120 Košice, Slovak Republic, Tel.: +42-95-6333458, 6024108, Fax.: +42-95-633 0115, E-mail: stupak@tuke.sk, lukacr@tuke.sk, marchs@tuke.sk

2. Šumové modely

Rozoznávame dva druhy impulzového šumu. V prípade 8 bitového kvantovania impulzový šum s premenlivou hodnotou, jednoducho nazvaný impulzový šum, znehodnotí vzorku signálu náhodnou hodnotou s rovnomerným rozložením z intervalu ⟨0, 255⟩. Tento proces je možné vyjadriť v tvare

$$y_i = \begin{cases} x_i & 1-p_{rnd} \\ rnd & p_{rnd} \end{cases} \quad (1)$$

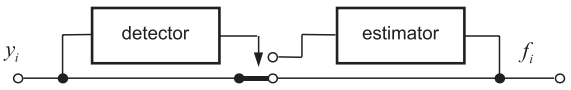
kde y je znehodnotený signál, x je pôvodný nezašumený signál, rnd náhodná hodnota, p_{rnd} je pravdepodobnosť výskytu impulzu a i je pozícia vzorky signálu.

$$y_i = \begin{cases} x_i & 1-(p_0+p_{255}) \\ 255 & p_{255} \\ 0 & p_0 \end{cases} \quad (2)$$

Druhým šumovým modelom je impulzový šum s pevnou hodnotou, tzv. čiernobiely šum, ktorý znehodnotí signál mimoležiacimi prvkami. Pravdepodobnosti p_0 a p_{255} sú pravdepodobnosti výskytu impulzov s jasovými úrovňami „0“ a „255“.

3. Procesor

Z modelu impulzového šumu vyplýva, že len niektoré časti signálu budú znehodnotené (zvyčajne do 30 %). Z tohto dôvodu by mali byť spracovávané len znehodnotené prvky, zatiaľ čo zvyšné nezašumené vzorky by mali byť ponechané bez zmeny. Avšak podstatná časť súčasných filtračných postupov spracováva každý signálový prvok a to bez ohľadu, či je tento prvok znehodnotený alebo nie. Spomínaný procesor pozostáva z dvoch častí (obr. 1).



Obr. 1 Architektúra procesora
Fig. 1 Architecture of the processor

Samotný princíp je jednoduchý [7,8,10-12,14]. V prvej časti detektora sú vyhodnotené vstupné dáta y_i. V prípade detekcie neznehodnoteného prvku, výstup filtra je rovný hodnote spracovávaného prvku. Výkon detektora ovplyvňuje množstvo falošných detekcií. V prípade detekcie znehodnoteného signálového prvku sa tento privádza na vstup estimátora, v ktorom sa koriguje odchýlka od pôvodnej vzorky.

Tento článok je zameraný na nelineárne estimátory. V nasledujúcich častiach budú popísané typy estimátorov patriacich do triedy kompozičných filtrov.

4. Mediánový filter

Mediánový filter je najjednoduchším filtrom patriacim do triedy kompozičných filtrov [1]. Tento filter je často používaný kvôli svojim vlastnostiam ako výborné potláčanie impulzového šumu so súčasným zachovaním hrán. Navyše, tento filter je vysoko robustný.

2. Noise model

Two impulse noise models are usually used. The first one is the variable valued impulse noise, simply called impulse noise, where some of the signal elements are replaced by random value with uniform distribution from interval ⟨0, 255⟩ in case of 8bit-quantized signal. It is defined as follows:

$$y_i = \begin{cases} x_i & 1-p_{rnd} \\ rnd & p_{rnd} \end{cases} \quad (1)$$

where y is the distorted signal, x is the original, noise-free signal, rnd random value, p_{rnd} probability of impulse occurrence and i is position of the signal elements.

$$y_i = \begin{cases} x_i & 1-(p_0+p_{255}) \\ 255 & p_{255} \\ 0 & p_0 \end{cases} \quad (2)$$

The second noise model (2) is the fixed value impulse noise, or so-called salt and pepper noise, which distorts the signal by outlier. The p_0 and p_{255} are the probabilities of “0“ and “255“ impulses, respectively.

3. The proposed processor

From the impulse noise model is clear that only few elements of the signal, usually up to 30%, are distorted. Therefore, only distorted signal elements should be filtered and the rest, noise-free elements should be left without filtering. However, the majority of present filters process every elements of the signal even the not distorted signal elements, too. The proposed processor consists of two parts (Fig. 1).

Its principle is simple [7,8,10-12,14]. The first part of the processor, the detector, investigates the input data y_i. In the case of noise-free detection, the processed sample is sent to the output without change. The amount of false detection influences the detector performance. In the case of impulse detection, the distorted signal element is sent to the estimator that tries to correct this element.

This paper is focused on non-linear estimators. The following sections describe several type of estimators belong to class of stack filters.

4. Median filter

The simplest filter belonging to the class of stack filter is the median filter [1]. This filter is preferred in application because of its performance. The MF well suppresses impulse noise and maintains image edges at same time. Moreover, it is generally the most robust filter.

Princíp filtrácie je nasledovný. Operačné okno (OW) sa pohybuje po signále. Rozmer OW je zvyčajne volený v rozmedzí od troch do sedem. Výstup filtra je určený aplikovaním operátora mediána (4) na dáta z OW. Mediánový operátor zoradí dáta v zostupnom poradí podľa veľkosti. Nech x_1, x_2, ..., x_3 sú dáta z operačného okna, potom zoradené vzorky sú

$$x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(n)}. \tag{3}$$

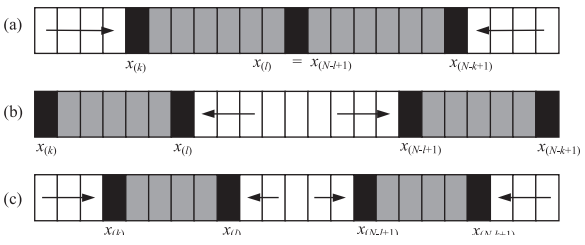
Značením $x_{(i)}$ sa označuje i-tá poriadková štatistika, a maximálne a minimálne prvky z OW sú určené $x_{(n)}$ and $x_{(0)}$. V prípade spracovania obrazov je veľkosť okna nepárna.

$$med(x) = \begin{cases} \dfrac{x_{\left(\frac{N}{2}\right)} + x_{\left(\frac{N+1}{2}\right)}}{2} & \textit{rozmer je nepárny} \\ x_{\left(\frac{N+1}{2}\right)} & \textit{rozmer je párny} \end{cases} \tag{4}$$

Za hlavný nedostatok mediánového filtra je považované rozmazávanie hrán a odstránenie jemných signálových detailov. Z tohto dôvodu bolo vytvorených niekoľko ďalších filtrov založených na poriadkových štatistikách. Tieto filtre eliminujú túto nevýhodu.

5. Vyhladzovací LUM filter

MF však vo veľkom počte aplikácií vnáša priveľa vyhladzovania, čo v praxi znamená rozmazanie hrán a detailov. Rozmazanie obrazu tak môže predstavovať väčšiu odchýlku od originálneho obrazu, než zašumený obraz. Rovnováhou medzi potlačením šumu a ochranou signálových detailov sa vyznačujú vyhladzovacie LUM filtre (Lower Upper Middle) [13,15]. Vyhladzovací LUM filter sa vyznačuje jednoduchou štruktúrou, pričom úroveň vyhladenia je riadená ladiacim parametrom k na vyhladzovanie. Zmenou ladiaceho parametra sa môžu získať rôzne úrovne vyhladenia od identického filtra (pre $k=1$, kde $y=x$) až po maximálne množstvo vyhladenia vykonané MF ($k = (N + 1)/2$).



Obr. 2 LUM filter ako: a) vyhladzovací b) ostriaci c) hybridný filter
Fig. 2 LUM filter: a) smoother b) sharpener c) hybrid filter

Vyhladzovacia funkcia LUM filtra je tvorená porovnaním spracovávanej vzorky x_4 s vyššou a nižšou poriadkovou štatistikou. Ak x_4 patrí do rozsahu vymedzenom týmito vzorkami, nie je menené. V opačnom prípade je nahradené vzorkou ležiacou bližšie k mediánu.

Výstup vyhladzovacieho LUM filtra je definovaný

$$y = med\left\{x_{(k)}, x_4, x_{(N-k+1)}\right\} \tag{5}$$

kde $1 \leq k \leq [N + 1]/2$, $x_{(k)}$ a $x_{(N-k+1)}$ vyššia poriadková štatistika zoradenej množiny.

Its principle is following. An operation window (OW) is moved over the signal. Usually, the OW size varies from three up to seven. The data from OW are sent to the median operator (4) that calculates the filter output. The median operator arranges the data of the OW in ascending order of magnitude. Let x_1, x_2, ..., x_3 are the data of the OW, then the arranged data are

$$x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(n)}. \tag{3}$$

The $x_{(i)}$ is called the i-th order statistic, and the maximum and minimum of the OW are denoted by $x_{(n)}$ and $x_{(0)}$, respectively. Usually, in the case of image processing the size of the OW is odd.

$$med(x) = \begin{cases} \dfrac{x_{\left(\frac{N}{2}\right)} + x_{\left(\frac{N+1}{2}\right)}}{2} & \textit{OW size is even} \\ x_{\left(\frac{N+1}{2}\right)} & \textit{OW size is odd} \end{cases} \tag{4}$$

The main drawback of the MFis that it blurs signal edges and removes fine details of the signal. On that account another filters based on order statistic trying to eliminate those disadvantages were developed.

5. LUM smoother

In many applications, the median introduces too much smoothing that in the practice means the blurring of edges and details. The LUM smoother [13,15], a subclass of a lower-upper-middle (LUM) filters, achieves the best balance between noise smoothing and signal-detail preservation. A structure of LUM smoother is based on tuning parameter k for the smoothing. Varying this parameter changes the level of the smoothing from no smoothing (i.e. identity filter for $k=1$, where $y=x$) to the maximum amount of smoothing (i.e. median, $k = (N + 1)/2$).

Thus, the smoothing function is created by a simply comparing of the processed sample to the lower- and upper-order statistics. If x_4 lies in a range formed by these order statistics, it is not modified. If x_4 lies outside this range it is replaced by a sample that lies closer to the median.

The output of LUM smoother is given by

$$y = med\left\{x_{(k)}, x_4, x_{(N-k+1)}\right\} \tag{5}$$

where $1 \leq k \leq [N + 1]/2$, $x_{(k)}$ a $x_{(N-k+1)}$ are lower and upper order statistics of the ordered set.

Trieda vyhladzovacích LUM filtrov je ekvivalentná so stredne váženými mediánmi (CWM - Center-Weighted Median), ktorých výstup je mediánom modifikovanej množiny mnohonásobnú spracovávanú vzorku. Implementácia CWM podľa (5) požaduje menej operácií, než (6), pretože menej prvkov musí byť triedených.

$$y = med\left\{W \cup \underbrace{\{x_4, x_4, \ldots, x_4\}}_{w-1}\right\} \quad (6)$$

V (6) w je váha strednej vzorky, pričom sa uvažuje nepárne kladné celé číslo. Vzájomný vzťah medzi parametrom pri CWM a parametrom k pri LUM filtri je daný

$$w = N - 2k + 2 \quad (7)$$

Proces zoraďovania dát je výpočtovo veľmi náročný. V ďalšej časti je popísaný filter ktorý sa vyznačuje rýchlou VLSI implementáciou.

6. Kompozičný filter

Mediánová filtrácia binárnych signálov je relatívne jednoduchá a celkom dobre spracovaná. Výpočet mediána sa dá zjednodušiť na počítanie jednotiek vo vnútri pracovného okna. Ak ich počet je väčší alebo rovný ako $(N + 1)/2$, výstup mediánového operátora je jedna, v opačnom prípade je nula. Filtrácia binárnych signálov je atraktívna tak z praktického ako aj z teoretického hľadiska.

Binárny signál sa získa pomocou operátora prahovej dekompozície. Nech x je M úrovňový signál, v prípade 8 bitového signálu $M = 256$: $0 \le x \le M - 1$ a počet kvantizačných úrovní je $M - 1$: $1 \le j \le M - 1$. Signál x môže byť rozložený do $M - 1$ kvantizačných úrovní $x^{(j)}$, pomocou funkcie prahovej dekompozície $I_j(x)$ nasledovne:

$$x^{(j)} = T_j(x) = \begin{cases} 1 & if\ x \ge j \\ 0 & if\ x < j \end{cases} \quad (8)$$

Takto rozložené binárne signály môžu byť filtrovaný nezávisle aj pomocou mediánového filtra.

$$y^{(j)} = med(x_1^{(j)}, x_2^{(j)}, \ldots, x_n^{(j)}) \quad (9)$$

Rekonštrukcia výstupu filtra spočíva v sčítaní výstupov mediánových filtrov pre jednotlivé úrovne.

$$y = \sum_{j=1}^{M-1} y^{(j)} \quad (10)$$

Architektúra kompozičného filtra je znázornená na obr. 3.

Prahová dekompozícia má teoretický aj praktický význam v oblasti mediánovej filtrácie. Používa sa na popísanie štatistických rozdelení výstupu a na nájdenie koreňov mediánového

This definition is equivalent to the centre-weighted median that is given by the median over a modified set of observations containing multiple processed samples. However, the implementation of the LUM smoother as shown in (5) requires fewer operations than that of (6), since fewer elements must be sorted.

$$y = med\left\{W \cup \underbrace{\{x_4, x_4, \ldots, x_4\}}_{w-1}\right\} \quad (6)$$

In (6) w is the weight of the central sample and is assumed to be an odd positive integer. The relationship between the parameter in the centre-weighted median and the parameter k in the LUM smoother is

$$w = N - 2k + 2 \quad (7)$$

The process of arranging the data needs high computational demand. The next section describes a filter whose VLSI implementation is fast.

6. Stack filter

Median filtering of binary signals is relatively easy and fairly well understood. Its computation can be reduced to counting the 1's inside the OW. If their number is greater than or equal to $(N + 1)/2$, the output of the median is 1, otherwise is 0. The filtering of the binary signal is attractive from both practical and theoretical point of view.

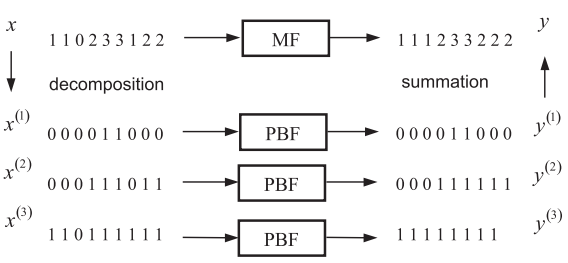
The binary signal can be obtained by the operator of *threshold decomposition.* Let x be an M - valued signal, in case of 8bit quantized signal $M = 256$: $0 \le x \le M - 1$ and consider the $M - 1$ thresholds: $1 \le j \le M - 1$. The signal x can be decomposed to $M - 1$ binary valued signals $x^{(j)}$, by using a threshold decomposition function $I_j(x)$ as follows:

$$x^{(j)} = T_j(x) = \begin{cases} 1 & if\ x \ge j \\ 0 & if\ x < j \end{cases} \quad (8)$$

These $M - 1$ binary valued signals can be filtered independently for example by MF.

$$y^{(j)} = med(x_1^{(j)}, x_2^{(j)}, \ldots, x_n^{(j)}) \quad (9)$$

The output of the whole filter can be reconstructed by summing the binary output signals.



Obr. 3 Kompozičný filter
Fig. 3 The stack filter

$$y = \sum_{j=1}^{M-1} y^{(j)} \quad (10)$$

The architecture of stack filter is shown in Fig. 3.

The threshold decomposition has both theoretical and practical significance for median filtering. It has been used to find the statistical distribution of the output of the median filter and of the roots of the

filtra. Prahová dekompozícia je taktiež použitá pri vývoji architektúr na rýchlu mediánovú filtráciu.

Binárny mediánový filter môže byť nahradený Boolovou funkciou na každej kvantizačnej úrovni. Výstupy týchto binárnych filtrov majú vlastnosť nakladania, ak výstup $y^{(j)}\ 1 \le j \le k-1$ v čase i sa skladá zo stĺpca jednotiek, na ktorých sa nachádzajú len nuly. Filtre, ktoré majú vlastnosť nakladania, sa nazývajú kompozičné filtre [1,5]. Výstup kompozičných filtrov je daný:

$$y = S_j(x) = \sum_{j=1}^{M-1} y^{(j)} = S_j\left(\sum_{j=1}^{M-1} T_j(x)\right) \qquad (11)$$

Do triedy kompozičných filtrov platia aj také nelineárne filtre ako mediánový filter alebo LUM filter. VLSI implementácia mediánových filtrov založených na prahovej dekompozícii sa veľmi zjednoduší náhradou mediánového filtra Boolovými funkciami.

Trieda kompozičných filtrov je veľmi široká. Ak veľkosť pracovného okna je n, počet kompozičných filtrov je viac ako $2^{2^{n/2}}$. Problém nájdenia vhodných kompozičných filtrov sa redukuje na problém nájdenia Boolových funkcií, ktoré spĺňajú vlastnosť nakladania. Také funkcie sú tzv. pozitívne Boolové funkcie. Boolová funkcia je vtedy a len vtedy pozitívna, ak na vstupné hodnoty nebol aplikovaný operátor inverzie.

Hlavnou úlohou návrhárov je nájsť vhodnú pozitívnu Boolovú funkciu, ktorá dobre potláča impulzový šum a súčasne zachováva hrany a jemné detaily signálu. Počet pozitívnych Boolových funkcií je veľmi vysoký. V prípade, ak veľkosť pracovného okna je päť, potom počet funkcií je 7581, ak veľkosť okna je sedem, počet funkcií narastá na neskutočne veľkú hodnotu $> 2.4\ 10^{12}$! Z toho dôvodu je potrené nájsť metódu, ktorá za krátky čas nájde vhodnú funkciu z tej veľkej množiny funkcií.

7. Neurónové kompozičné filtre

Neurónové siete podstatne zjednodušili návrh kompozičných filtrov. V tomto prípade binárne mediánové filtre na každej kvantizačnej úrovni sú nahradené neurónovou sieťou. Podobne ako v predchádzajúcom prípade kompozičný filter, môže byť homogénny, ak na všetkých kvantizačných úrovniach sú umiestnené filtre s rovnakými parametrami, alebo nehomogénny, ak na jednotlivých úrovniach parametre filtrov môžu byť rôzne. V skutočnosti však homogénny filter už dosahuje uspokojujúce výsledky.

Štruktúra kompozičného filtra je obzvlášť vhodná pre neurónové siete, nakoľko vstupné údaje sú {0,1}, a preto nie je potrebné normalizovať vstupné hodnoty do neurónovej siete. Výstup neurónovej siete mal by byť podobne ako vstup 0 alebo 1. Takýto výstup je možné dosiahnuť len zavedením prahu na výstup siete, alebo pomocou prahovej aktivačnej funkcie.

$$y = f_{act}(x) = \begin{cases} 1 & if\ x \ge 0.5 \\ 0 & if\ x < 0.5 \end{cases} \qquad (12)$$

kde x je vstup neurónovej siete a y je výstup siete.

Taká aktivačná funkcia by mala byť na výstupe neurónovej siete. Neurónová sieť sa môže nachádzať v dvoch stavoch. Prvý je stav učenia, keď parametre siete sú nastavované, a druhý stav, keď

MF. It has also been used to develop architectures for fast median filtering.

The binary median filters at each level can be replaced by Boolean functions. The outputs of these binary filters possess what is referred to as the stacking property if the binary output signals $y^{(j)}\ 1 \le j \le k-1$, are piled and the pile of y at time i consists of a column of 1's having a column of 0's on top. The filters that support the stacking property are called stack filters [1,5]. The output of a stack filter is given by:

$$y = S_j(x) = \sum_{j=1}^{M-1} y^{(j)} = S_j\left(\sum_{j=1}^{M-1} T_j(x)\right) \qquad (11)$$

Stack filters constitute a broad class of non-linear filters having median filters and LUM filters as special cases. The replacement of MF by Boolean function greatly facilitates the VLSI implementation of the MF based on threshold decomposition.

The class of stack filters is very large. Their number, when the filter window is n, grows faster than $2^{2^{n/2}}$. The problem of finding them reduces to the problem of finding stackable Boolean functions, or so-called positive Boolean functions (PBF). It has been established that a Boolean function is stackable if and only if it contains no complements of the input variables.

The main task of designers is to find a suitable PBF functions that well suppress the impulsive noise and concurrently well preserves signal edges and fine details. However, the amount of PBF functions is very high. In case of OW of size five, there are 7581 functions and in case of size seven, the number of PBF is tremendously high $> 2.4\ 10^{12}$! Therefore, it is necessary to find methods that find the optimal PBF from such great set.

7. Neural stack filters

Neural networks (NN) [9] greatly simplified the design of the stack filters. In this case, the binary median filters at each level are replaced by neural networks. Similarly as in previous case the stack filter can be homogenous, at each level are identical filters with identical parameters, or non-homogenous, at each level the parameters of the filters can vary. Usually, it a sufficient result can be obtained by the homogenous representation.

The stack filter structure is useful in case of NN, because the input data are {0,1} so it is not necessary to normalize the input for the NN. The output of the NN should be also 0 or 1. Such output can be obtained only by a hard limiter, or also called threshold activation function.

$$y = f_{act}(x) = \begin{cases} 1 & if\ x \ge 0.5 \\ 0 & if\ x < 0.5 \end{cases} \qquad (12)$$

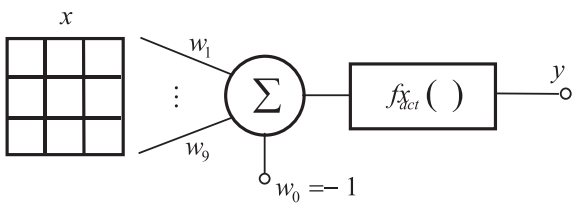
where x is the input of the neuron and y is the neuron output.

Such activation function should be used at output layer of the NN. There are two modes of the NN. The first one is the training mode, where the weights of the NN are set up and the second one,

neurónová sieť spracováva vstupné dáta. V tomto stave váhy siete sú konštantné. Avšak v procese učenia nemôže byť použitá prahová aktivačná funkcia. Z toho dôvodu pri učení sa používa tzv. sigmoidálna aktivačná funkcia.

$$y = f_{act}(x) = \frac{1}{1 + wxp(-x)} \qquad (13)$$

Bolo ukázané, že neurónová sieť s jedným neurónom dáva dobré výsledky [3,4]. Architektúra siete je na obr. 4. Výstup siete je získaný váhovaným súčtom vstupov, pričom výsledok je privedený do vhodnej aktivačnej funkcie. Takáto jednoduchá štruktúra môže výrazne zlepšiť vlastnosti kompozičného filtra. Navyše neurónová sieť môže byť učená pre rôzne úlohy, filtrácia impulzov, zvýraznenie hrán atď. Jediná nevýhoda je VLSI realizácia siete, ktorá je zložitejšia ako realizácia Boolových funkcií.



Obr. 4 Neurónová sieť s jedným neurónom
Fig. 4 Neural network with one neuron

8. Záver

V tomto článku bol ukázaný krátky prehľad triedy kompozičných filtrov. Jednoduchosť hardvérovej realizácie a vlastnosť potláčania impulzového šumu je dôvodom rozšírenia tejto triedy filtrov. Dobre známy mediánový filter a jeho vylepšenie vyhladzovací LUM filter môže byť tiež realizovaný kompozičnými filtrami. Návrh kompozičných filtrov s optimálnymi Boolovými funkciami je časovo náročná úloha. Iný spôsob návrhu ponúkajú neurónové kompozičné filtre. Vlastnosti neurónových kompozičných filtrov sú veľmi zaujímavé najmä kvôli jednoduchej architektúre siete. Na druhej strane však VLSI implementácia je zložitejšia ako v prípade Boolových funkcií. Z tohto dôvodu výskum by sa mal zamerať na zrýchlenie hľadania vhodných Boolových funkcií. Permutačná teória a genetické algoritmy sú jedným z možných riešení ako zrýchliť vyhľadávanie [6].

9. Poďakovanie

Práca prezentovaná v tomto príspevku bola podporovaná grantom ministerstva školstva a Slovenskou akadémiou vied VEGA pod číslom 1/5241/98.

Recenzenti: V. Moucha, L. Schwartz

where the NN processes the input data. In this case the NN weights are fixed. However, the activation function of the neurons cannot be hard limiter in training mode. Therefore, it must be changed to sigmoidal function, also called the soft limiter.

$$y = f_{act}(x) = \frac{1}{1 + wxp(-x)} \qquad (13)$$

It was shown [3,4] that the NN with only one neuron gives sufficient results. The architecture of the NN is shown in Fig. 4. The output of the NN is obtained by weighted summing the input data and applying the appropriate activation function on result. Such a simple structure can greatly improve the performance of the stack filter. Moreover, the neural filter can be trained for an arbitrary task, impulse noise suppression, edge enhancement, etc. However, the VLSI realization of the NN is more difficult than the realization of the Boolean functions.

8. Conclusion

This paper gave a short review of the class of stack filters. The simplicity of hardware realisation and the performance of impulse noise suppression make this class widely useful in many applications. The well-known median filter and its improvement the LUM smoother can be realised as the stack filters. The design of stack filters with optimal Boolean function is very computation demanding. An alternative way of design, the neural stack filters were shown. The performance of the neural stack filters is very interesting because of simple architecture of neural network. However the VLSI implementation is more complicated than the implementation of the PBF. Therefore, the future research should be oriented to accelerate the search of suitable PBF. Permutation theory gives one way of solving the problem and another way, the genetics algorithm [6].

9. Acknowledgements

The work presented in this paper was supported by the Grant Agency of the Ministry of Education and Academy of Science of the Slovak Republic VEGA under Grant No. 1/5241/98.

Reviewd by: V. Moucha, L. Schwartz

10. Literatúra - References

[1] PITAS, I., VENETSANOPOULOS, A.N.: Order statistics in digital image processing. Proceedings of the IEEE, Vol.80, No.12, December 1992, pp.1893-1916

[2] KIM, S.R., EFRON, A.: Adaptive robust impulse noise filtering. IEEE Transactions on signal processing, Vol.43, No.8, August 1995, pp.1855-1866

[3] DRUTAROVSKÝ, M.: Neural weighted order statistics filters based on threshold decomposition. Thesis. Košice, June 1995, (in Slovak)
[4] DRUTAROVSKÝ, M., MARCHEVSKÝ, S.: The Methods of Design and Implementation of Stack Filters for Image Processing. Radioengineering, Vol.4, No.1, April 1995, pp.13-17
[5] GABBOUJ, M., COYE, E.J., GALLAGHER, N.C.: An Overview of Median and Stack Filtering. Circuit Systems Signal Process, Vol.11, No.1, 1992, pp.7-45
[6] HARVEY, N.R., MARSHALL, S.: Optimum Genetic Algorithms for the Design of Stack-Filters. IEEE Transactions on Signal Processing. Vol.42, No.4, April 1994, pp.832-835
[7] MARCHEVSKÝ, S., DRUTAROVSKÝ, M., CHOMAT, O.: Iterative Filtering of Noisy Images by Adaptive Neural Network Filter. New Trends in Signal Processing I, Liptovský Mikuláš, May 1996, pp.118-121
[8] STUPÁK, CS.: Digital Image Filtration Based on Local Statistics. 3rd International Scientific Conference Elektro '99, Žilina, May 25-26 1999, pp.106-111
[9] STUPÁK, CS., MARCHEVSKÝ, S., DRUTAROVSKÝ, M.: Searching the Optimal Training Set for Neural Network Training. Journal of Electrical Engineering, No.5-6, Vol.50, 1999, pp.143-147
[10] STUPÁK, CS., LUKÁČ, R.: Impulse Detection in Grayscale Images. Digital Signal Processing '99, Herľany, September 29-30, 1999, pp.96-99
[11] LUKÁČ, R., STUPÁK, CS.: A Class of Impulse Detectors Controlled by a Threshold. Proceedings of the 3rd International Scientific Conference INFORMATICS AND ALGORITHMS '99, Faculty of Production Technologies at the Technical University of Košice with residence in Prešov, Slovakia, Sept. 9-10 1999, pp.178-181
[12] LUKÁČ, R., MARCHEVSKÝ, S.: Threshold Impulse Detector Based on LUM Smoother (LUMsm detector). Journal of Electrical Engineering, submitted
[13] LUKÁČ, R.: An Adaptive Control of LUM Smoother. Radioengineering, Vol.9, No.1, April 2000, pp.9-12.
[14] LUKÁČ, R.: Impulse Detection by Entropy Detector (H - Detector). Journal of Electrical Engineering, No.9-10 Vol.50, 1999, pp.310-312.
[15] LUKÁČ, R., MARCHEVSKÝ, S.: A Neural LUM Smoother. Radioengineering, submitted.
[16] STUCK, B., KLEINER, B.: A statistical analysis of telephone noise. Bell Syst. Tech. J., Vol.53, Sept. 1974, pp.1263-1320
[17] SHINDE, M., GUPTA, S.: A model of hf impulsive atmospheric noise. IEEE Trans. Electromag. Compat., Vol. EMC-16, May 1974, pp.71-75
[18] MIDDLETON, D.: Man-made noise in urban environments and transportation systems. IEEE Trans. Commun., Vol. COM-21, Nov. 1973, pp.1232-1241

Oceľobetónové konštrukčné prvky

Monografiu vydala Žilinská univerzita r. 1999, 100 strán, 56 obrázkov, 27 tabuliek, ISBN 80-7100-627-0.

*Monografia autorov **Jána Bujňáka** a **Kazimierza Furtaka** podáva súbor informácií o navrhovaní a výsledkoch výskumu oceľových dutých prierezov, vyplnených betónom ako aj obetónovanými tyčovými prvkami, pri kombinovanom namáhaní tlakom a ohybom. Je jednou z prvých prác monotematicky zameraných na oceľobetónové konštrukčné prvky, atraktívne nielen v konštrukciách pozemných stavieb, ale aj v inžinierskom staviteľstve, čo podrobnejšie dokumentuje úvodná kapitola.*

Materiálové vlastnosti betónu, konštrukčnej ocele a výstuže predstavujú dôležité vstupné údaje. Venuje sa im preto 2. kapitola publikácie. Poslaním 3. kapitoly je detailnejší rozbor pôsobenia kruhového stĺpa pri zaťažení axiálnym tlakom. Dôraz sa pritom kladie na teoretické riešenie napätosti jednotlivých častí prierezu oddelene a následne v superpozícii. Tento postup má objasniť prácu stĺpov pod zaťažením, čo by malo pomôcť pri praktickom návrhu správne aplikovať normové postupy.

V rozsiahlejšej 4. kapitole sú podklady pre praktický návrh oceľobetónových prvkov nielen tlačených, ale prenášajúcich tiež ohybové účinky. Okrem rúr, vyplnených betónom, táto časť obsahuje návod na výpočet aj obetónovaných oceľových stĺpov. Podrobnejší popis alternatívnych postupov umožní praktický návrh tiež neštandardných alebo neobvyklých konštrukčných oceľobetónových častí. Publikácia tak môže poslúžiť projektantom, ale najmä vysokoškolským študentom a niektoré náročnejšie časti aj ako podklad pre ďalší výskum. Je výsledkom spolupráce medzi Žilinskou univerzitou a Polytechnikou v poľskom Krakove. Na Slovensku je k dispozícii v Predajni Žilinskej univerzity, Vysokoškolákov 24, 010 11 Žilina.

Doc. Ing. Josef Vičan, CSc.

POKYNY PRE AUTOROV PRÍSPEVKOV DO ČASOPISU KOMUNIKÁCIE - vedecké listy Žilinskej univerzity

1. Redakcia prijíma iba príspevky doteraz nepublikované alebo inde nezaslané na uverejnenie.
2. Rukopis musí byť v jazyku slovenskom a anglickom (týka sa autorov zo Slovenska). K článku dodá autor resumé v rozsahu maximálne 10 riadkov v slovenskom a anglickom jazyku).
3. Príspevok prosíme poslať: e-mailom na adresu **holesa@nic.utc.sk** alebo **polednak@fsi.utc.sk** alebo **vrablova@nic.utc.sk** alebo doručiť **na diskete 3,5″** v programe Microsoft WORD, 1 kópia článku na adresu: ŽU - rektorát, Ing. Vráblová, odd. vedy a výskumu, Moyzesova 20, 010 26 Žilina.
4. Skratky, ktoré nie sú bežné, je nutné pri ich prvom použití rozpísať v plnom znení.
5. Obrázky, grafy a schémy, pokiaľ nie sú spracované v Microsoft WORD, je potrebné priložiť na diskete (ako .GIF, .TIF, .CDR, .BMP, .WMF, .PCX, .JPG súbory), prípadne nakresliť kontrastne na bielom papieri a predložiť v jednom exemplári. Pri požiadavke na uverejnenie fotografie priložiť ako podklad kontrastnú fotografiu alebo diapozitív. Pre obidve mutácie spracovať jeden obrázok s popisom v slovenskom a anglickom, resp. len v anglickom jazyku.
6. Odvolania na literatúru sa označujú v texte alebo v poznámkach pod čiarou príslušným poradovým číslom v hranatej zátvorke. Zoznam použitej literatúry je uvedený za príspevkom. Citovanie literatúry musí byť podľa záväznej STN 01 0197 „Bibliografické citácie“.
7. K rukopisu treba pripojiť presnú adresu autora, tituly, plné meno a priezvisko, dátum narodenia, rodné číslo, adresu inštitúcie v ktorej pracuje, e-mail adresu, funkciu, ktorú zastáva, číslo telefónu alebo faxu.
8. Príspevok posúdi redakčná rada na svojom najbližšom zasadnutí a v prípade jeho zaradenia do niektorého z budúcich čísiel podrobí rukopis recenziám a jazykovej korektúre. O výsledku bude redakcia informovať autora ústne alebo písomne. Po korektúrach a zabudovaní pripomienok recenzentov bude posledný obťah príspevku (pred tlačou) poslaný autorovi na definitívnu kontrolu a odsúhlasenie.
9. Termín na dodanie článkov jednotlivých čísiel je 30. 11., 28. 2., 31. 5. a 31. 8.
10. V číslach 3/2000 až 4/2000 budú tieto nosné témy jednotlivých čísiel: Materiály a hodnotenie medzných stavov, technológie výroby kovových materiálov, Dopravná infraštruktúra.

COMMUNICATIONS - Scientific Letters of the University of Žilina Writer's Guidelines

1. Submissions for publication must be unpublished and not be a multiple submission.
2. Manuscripts written in English language must include abstract also written in English. The abstract should not exceed 10 lines.
3. Submissions should be sent: by e-mail to one of the following addresses: **holesa@nic.utc.sk** or **polednak@fsi.utc.sk** or **vrablova@nic.utc.sk** or on a **3,5″ diskette** in Microsoft WORD, a hard copy to the following address: University ofŽilina, OVaV, Moyzesova 20, SK-010 26 Žilina, Slovakia.
4. Abbreviations which are not common must be used in full when mentioned for the first time.
5. Figures, graphs and diagrams, if not processed by Microsoft WORD, must be sent on a diskette (as .GIF, .TIF,.CDR, .JPG, .PCX, .VMF, .BMP files) or drawn in contrast on white paper, one copy enclosed. Photographs for publication must be either contrastive or on a slide.
6. References are to be marked either in the text or as footnotes numbered respectively. Numbers must be in square brackets. The list of references should follow the paper.
7. The authoręs exact mailing address, titles, full names, e-mail address (telephone or fax number), the address of the organisation where he works and contact information must be enclosed.
8. The submission will be assessed by the editorial board in its forthcoming session. In the case that the article will be the acctepted for future volumes, the board submits the manuscript to the editors for review and language correction. After reviewing and incorporating the editors' remarks, the final draft (before printing) will be sent to authors for final review and adjustment.
9. The deadlines for submissions are as follows: November 30, February 28, May 31 and August 31.
10. In the numbers 3/2000 till 4/2000 the basic topics of the separate editions will be: Materials and marginal stage valuation, technology of metallic substance production, Transport infrastructure

kamUNikácie
COMMUNICATIONS

VEDECKÉ LISTY ŽILINSKEJ UNIVERZITY
SCIENTIFIC LETTERS OF THE UNIVERSITY OF ŽILINA

Adresa redakcie:
Address of the editorial office:
Žilinská univerzita
Oddelenie pre vedu a výskum
Office for Science and Research
Moyzesova 20, Slovakia
SK 010 26 Žilina
Tel.: +421/89/5620 392
Fax: +421/89/7247 702
E-mail: polednak@fsi.utc.sk, holesa@nic.utc.sk

Vydáva Žilinská univerzita
v EDIS - vydavateľstve ŽU
J. M. Hurbana 15, 010 26 Žilina
pod registračným číslom 1989/98
ISSN 1335-4205

It is published by the University of Žilina in
EDIS - Publishing Institution of Žilina University
Registered No: 1989/98
ISSN 1335-4205

Objednávky na predplatné prijíma redakcia
Vychádza štvrťročne
Ročné predplatné na rok 2000 je 500,- Sk

Order forms should be returned to the editorial office
Published quarterly
The subscription rate for year 2000 is 500 SKK.

http://www.utc.sk/komunikacie