A Short Note on Emergence of Computational Robot Consciousness[1]

Jozef Kelemen

Institute of Computer Science, Silesian University, Opava, Czech Republic,
and VSM College of Management, Bratislava, Slovak Republic
kelemen@fpf.slu.cz

Abstract: A way is sketched how to answer the question about the computational power supposed behind the consciousness, esp. the computational robot consciousness. It is illustrated that a formal model of the possible functional architecture of certain type of robot enables to satisfy the test of emergence proposed earlier.

Keywords: Robots, Machine Consciousness, Emergence, Eco-Grammar Systems

1 Introduction

This article in certain extent enlarges the existing spectrum of consciousness-oriented computer science opinions proposing a possible way of how to treat consciousness from the positions of the theory of computation, especially from the point of view of the theory of formal grammars and languages, more specifically, of the theory of so called *eco-grammar systems* as presented in (Csuhaj-Varju et al., 1997).

C*onsciousness will not need to be programmed in. They will emerge*, stated R. Brooks (1999, p. 185), one among the leading specialaits of the present days artificial intelligence and advanced robotics research. Our main goal consists in arguing – from a theoretical computer science point of view – for the *possibility that* consciousness will emerge. First, we will try to show some *views of consciousness* which seem to be relevant for our computationalistic treatment of the topic. Then we will focus to another relevant matter – to the *nature of emergent* phenomena, and to the *phenomenon of emergence*.

[1] The author's research is supported by the Czech Ministry of Education grant MSM 4781305903, and by the Gratex International, Inc., Bratislava, Slovakia. The present article is based on the author's previous contribution (Kelemen, 2006a) delivered at the 15th International Workshop on Robotics in Alpe-Adria-Danube Region, RAAD 2006 (Balatonfüred, Hungary, June 15-17, 2006).

In order to connect the contents of the centents of the just mentioned sections, we will continue with sketching two fundamental *approaches to the architecture of robots*. The sketched architectures will be then analysed from the perspective of their *computational power*. We will recognize at least one fundamental difference between the two basic architectural approaches to robot construction, and we will close the article with expressing the discovered difference in the conceptual framework of advanced robotics.

2 Consciousness

A. Zeman – approaching the study of consciousness from positions of a neurophysiologist with a considerable strong philosophical background – writes on three basic meenings of consciousness, all related with knowledge: *Being awake, our first sense of cons-ciousness*, he writes, *is a pre-condition for acquiring knowledge of all kinds. Once awake, we usually come* – he continues – *by knowledge through ex-perience, the second sense of cons-ciousness. The knowledge we gain is then 'conscious' in the third sense we distinguished*, he completes his analysis (Zeman, 2002, p. 36).

From another position – from the position typical for the fields of artificial intelligence and cognitive science – is the subject of consciousness treated e.g. by P. O. Haikonen. As a crutial for forming the consciousness he recognize the phenomenon of perception of the self. *What we actually see is only the projected image on the retina, what we actually hear is the vibration of the eardrums, so why don't we just perceive these as such, percepts originating at the senses or originating at the related sensory nerve endings? How can we possibly perceive the situation being anything else?* he asks (Haikonen, 2003, p. 71).

The above questions formulated by Haikonen lead to the concept of some kind of "internal" (mental) representation of the "external" (physical) stimuli sensed by machines, and in consequences to the familiar mind/body problem of the philosophy of mind. Haikonen's position is explained by an example (Haikonen, 2003, pp. 248-249) as follows: *The operation of the signals in the cognitive machine can be compared to radio transmission where a carrier signal is modulated to carry the actual radio signal. The carrier wave is what is received, yet what is detected is the modulation, the actual sound signal that is in causal connection to the original physical sound via a microphone. We do not hear the carried signal even through without it there would be no music. Thus it is possible to perceive carried informa-tion without the perception of the material basis of the carrier.*

In (Holland, Goodman, 2003), an analysis is given concerning the role internal symbolic representations of the robots environment and their own capabilities of robots – as the term representation is used in traditional artificial intelligence research – to robots abilities to act in the their outer environments, and the abilities of robots to construct and use their internal symbolic representations are connected with the phenomenon of the robots consciousness.

3 Emergence

The traditional and most vide informal definition of emergence has been formulated e.g. in (Holland, 1998, pp. 121-122): Emergence is, according him „*... a product of coupled, context-dependent interactions. Technically these interactions, and the resulting system, are nonlinear: The behavior of the overall system cannot be obtained by summing the behaviors of its constituent parts... However, we can reduce the behavior of the whole to the lawful behavior of its parts, if we take nonlinear interactions into account*".

In (Searle, 1992) at least two interpretations of the concept of emergence are distinguished: The first one Searle calls *emergence1*. This kind of emergence refers that a higher order feature of a system can be understood by a complete explication of the parts of a system and their interpretation. The more adventurous conception of emergence Searle calls *emergence2*. A feature of a system emerges in this way if it has causal powers that cannot be explained by the parts of a system and their interactions.

The consequence of the conception of *emergence2* for the consciousness is the following: If consciousness is of the *emergence2* type, that it could cause things that could not be explained by the causal properties of the neuronal networks. A serious problem arising from the *emergence2* – called in (Van Gulick, 2001) as *radical kind of emergence* – consists in making the physicalist view of consciousness problematic*: If* [...] *systems could have causal powers that were radically emergent from the powers of their parts in the sense that those system-level powers were not deter-mined by the laws governing the powers of their parts, then that would seen to imply the existence of powers that could override or violate the laws governing the powers of the parts*, states (Van Gulick, 2001, p. 18).

The emergent nature – we hope the radical one – of phenomena appearing in complex systems we may test using the so-called *test of emergence* proposed in (Ronald et al., 1999). The requirements putted onto systems in which the emergence of some phenomenon appears are the following (Roland et al., 1999):

Design. The designer designs the systems by describing *local* in-teractions between components in a language L_1.

Observations. The observer describes global behaviors of the running system using a language L_2.

Surprise. The language of design L_1 and the language of observation L_2 are distinct, and the causal link between the elementary interactions program-med in L_1 and the observations observed in L_2 is non-obvious.

4 Robots

Intuitively we feel that any robot consciousness necessarily requires attentiveness and emotionality. This opinion is expressed clearly in first dreams on robots in artistic works; cf. e.g. (Horakova, Kelemen, 2008), but also in numerous theoretical studies rooted in the computational and engineering approaches, e.g. in an attempt to a formal axiomatic definition of consciousness, an approach by which we will inspired in this section (Aleksander, Dunmall, 2003). According their opinion, being a *conscious agent* means – intuitively and roughly speaking – to have some kind of agent's *private sense* of an outer world, of a self in this world, of self's contemplative planning of when, what and why to act, of self's own inner emotional states Moreover it means also the conscious agents ability to include the self's *private sense* into all of its above mentioned functional capabilities. But how incorporate it into agents?

All the real experimental robots which work with internal symbolic representations of their outer environment have from our perspective one important common feature: At least in certain extent, their behavioral, re-presentational, and decision making capacities are based on the abilities of the present day computers to execute more or less complicated computations. In a theoretical level, these computations might be reduced into the form of the theoretical abstraction of computational processes known as *Turing-computations*, so computations performed by the abstract universal Turing machine. The *Turing machine* – with respect of their computational power in the machines over-simplified environments of symbols on a tape, and a head going one step left or right and rewriting the symbols according simple instructions sequences – differs in an important sense very significantly from the real embodied robots situated in dynamically changing physical environments, and interacting with these environments very massively in many different ways.

However, there exists a largely accepted hypothesis in theoretical computer science – the *Church-Turing hypothesis*; see e.g. (Cleland, 1993) for some discussion concerning it – according which, very roughly speaking, all what is intuitively in certain sense computable (so, transformable from certain inputs into certain outputs according precisely defined and exactly executed sequences of rules – according computer programs) is computable by the Turing machine.

Especially interactions are very appealing for re-consideration of the form of a "computation" performed by agents or robots, and for drawing perhaps new boundaries between what we consider as computable and what as non-computable.

In present day theoretical computer science there are numerous efforts to demonstrate that the notion of computation might be enlarged beyond the traditional boundaries of the Turing-computability. In (Burgin, Klinger, 2004) it is proposed to call algorithms and automata that are more powerful than Turing machines as *super-recursive*, and computations that cannot be realized or simulated by Turing machines as *hyper-computations*.

Turn now our attention toward the provoking notion of the *private sense* related to robots. To have the *private sense* means – metaphorically speaking, for more details see (Kelemen, 2006b) – to have an ability of a given robot to consider itself as another robot identical with it, and to consider this type of "schizophrenia" in the work of other functions which characterize our real robot. This type of recursion is might be extremely complicated for expressing it in the frame of the traditional paradigm of one-processor computation. It requires at least some suitable framework for dealing with behaviors that appear thanks to interrelations between individually autonomous robots.

The appearing situation insinuates the framework of considering a conscious robot as a system consisting in more then one agent, so in a form of a multi-agent systems. The robot's private sense is perhaps an emerging product of interactions of several other robots functional modules. Perhaps the conscious behavior of such a robot might be then described as a phenomenon, which emerges – in the above cited sense proposed in (Holland, 1998) – from interactions of traditionally computable behaviors of simpler constituting parts of it, and has the form of a hyper-computation. Let us try to demonstrate in the next section how it is possible to proceed in this way.

5 Emergent Computation

There are several different approaches to study the emergent computational power of interacting systems. In this Section we will sketch a formal model of robots with functional components producing a rule-governed Turing-computable behaviors each, but producing – as a whole – a behavior which does not be generated traditionally by any Turing-equivalent generative device, so which requires the generative power of hyper-computation. We will consider in this role the so-called eco-grammar systems. First, we introduce in a few words this model, presented originally in (Csuhaj-Varjú et al., 1997).

According (Csuhaj-Varju et al., 1997), an *eco-grammar system* Σ consists, roughly speaking, of

- a finite alphabet V,
- a fixed number (say n) of components evolving according sets of rules P_1, P_2, ..., P_n applied in a parallel way as it is usual in L-systems (Rozenberg, Salomaa, 1980), and of
- an environment of the form of a finite string over V (the states of the environment are described by strings of symbols w_E, the initial one by w_0).
- the functions φ and ψ which define the influence of the environment and the influence of other components, respectively, to the components (these functions will be supposed in the following as playing no roles, and will not be considered in the model of eco-grammar systems as treated in this article).

The rules of components depend, in general, on the state (on the just existing form of the string) of the environment. The particular components act in the commonly shared environment by sets of sequential rewriting rules R_1, R_2, ..., R_n. The environment itself evolves according a set P_E of rewriting rules applied in parallel as in L systems.[2]

The evolution rules of the environment are independent on components' states and of the state of the environment itself. The components' actions have priority over the evolution rules of the environment. In a given time unit, exactly those symbols of the environment that are not affected by the action of any agent are rewritten.

In the EG-systems we assume the existence of the so-called *universal clock* that marks time units, the same for all components and for the environment, and according to which the evolution of the components and of the environment is considered.

In (Wätjen, 2003) a variant of EG-systems without internal states of components is proposed and studied. The fixed number of components of the so-called teams of components in EG systems originally proposed in (Csuhaj-Varju, Kelemenova, 1998) is replaced by a dynamically changing number of components in teams. As the mechanism of reconfiguration, a function, say f, is defined on the set N of integers with values in the set $\{0, 1, 2, ..., n\}$ (where n is the number of components in the corresponding EG-system) in order to define the number of components in teams. For the i-th step of the work of the given EG-system, the function f relates a number $f(i) \in \{0, 1, 2, ... n\}$. The subset of the set of all components of thus EG-system of the cardinality $f(i)$ is then selected for executing the next derivation step of the EG system working with Wätjen-type teams. So, Wätjen, roughly speaking, proved that there exist EG-systems such that if f is (in the traditional sense) non-recursive function, then the corresponding EG-system generates a non-recursive (in fact a super-recursive) language.

[2] So, the triplet (V, P_E, w_E) is (and works as) a Lindenmayer-system.

The *emergent nature* of the behavior (language) generated by the above described EG system is – applying the above mentioned test of emergence – rather clear: The components of a given EG system generate *recursive languages* each. Recursive languages play, in the context of the above cited *emergence test* the role of the *designer language*. The local interactions of the components of the systems are given only. But *surprisingly*, the whole system generates a *non-recursive language* (behavior), which is, as the *language of the observer* of the system behavior, substantially, surprisingly, different, from computational positions, from the *language of the designer*.

Conclusions

We saw that there exist formalized systems set up from decentralized components with higher computational power as Turing machines have. There are no principal reasons to reject the hypothesis that it is possible to construct real robots as certain kind of implementations of these formalized systems. If we include into the functioning of such robots the activation of their functional modules according a non-recursive (in Turing sense) computation, the behavior of the agents might be non-recursive. We suppose that this situation may appear inif some of the functional partsof the robots are swich on or off on the base of the random behavior of the robots environments, for instance. So we exclude the situation when a computer simulation of randomness are included into the functional architecture of robots. Rather, we suppose the randomness appearing in the environment, a randomness which follows from the ontology of robots situated in their environments. More about this can be found in (Kelemen, 2005a) and (Kelemen, 2005b).

The ontological randomness might be caused by different reasons – by inprecise work of sensors and actuators of robots, by erroneous behavior of their hardwired or software parts, by non-determinism of the behavior of the environment, etc. All these influences may be reflectedin the specific behavior of the robots and we cannot reject the hypothesis that just these kind of irregularities cause also the phenomenon called robot consciousness.

So, going back to the Brooks' opinion from the beginning of this contribution: Consciousness will not need to be programmed in. They will emerge. The goal of this contribution was not to prove that *it will emerge* sometime durong the course of time, but to prove that *it may emerge* form the functional-computational structure and from properties of robots, and their massive interactions with their complicated unpredictable behaving environments.

References

[1] Aleksander, I., Dunmall, B.: Axioms and Tests for the Presence of Minimal Consciousness in Agents. In: *Machine Consciousness* (O. Holland, Ed.). Imprint Academic, Exeter, 2003, pp. 7-18

[2] Brooks, R. A.: *Cambrian Intelligence*. The MIT Press, Cambridge, Mass., 1999

[3] Burgin, M., Klinger, A.: Three Aspects of Super-Recursive Algorithms and Hyper-Computation or Finding Black Swans. Theoretical Computer Science 317 (2004) 1-11

[4] Cleland, C. E.: Is the Church-Turing Thesis True? *Minds and Machines* **3** (1993) 283-312

[5] Csuhaj-Varju, E., Kelemen, J., Kelemenova, A., Paun, Gh.: Eco-Grammar Systems – a Grammatical Framework for Lifelike In-Teractions. *Artificial Life* 3 (1997) 1-28

[6] Csuhaj-Varju, E., Kelemenova, A.: Team Behaviour in Eco- Grammar Systems. *Theoretical Computer Science* 209 (1998) 213-224

[7] Haikonen, P. O.: *The Cognitive Approach to Conscious Machines*. Imprint Academic, Exeter, 2003

[8] Holland, J. Emergence. Holland, J. H.: *Emergence*. Addison-Wesley, Reading, Mass., 1998

[9] Holland, O., Goodman, R.: Robots with Internal Models – a Route to Machine Consciousness? In: *Machine Consciousness* (O. Holland, Ed.) Imprint Academic, Exeter, 2003, pp. 77-109

[10] Horakova, J., Kelemen, J.: The Robot Story – why Robots were Born and How They Grew Up. In: *The Mechanical Mind in History* (Ph. Husbands, O. Holland, M. Wheeler, eds.). The MIT Pess, Cambridge, Mass., 2008, pp. 283-306

[11] Kelemen, J.: May Embodiment Cause Hyper-Computation? In: *Advances in Artificial Life, Proc. ECAL 05* (M. S. Capcarrére et al., eds.). Springer-Verlag, Berlin, 2005a, pp. 31-36

[12] Kelemen, J.: On the Computational Power of Herds. In: *Proc. IEEE 3rd International Conf. on Computational Cybernetics* (I. J. Rudas, Ed.). Budapest Polytechnic, Budapest, 2005b, pp. 269-273

[13] Kelemen, J.: Computational Robot Consciousness – a Pipe Dream or a (Future) Reality? In: Proc. 15th International Workshop on Robotics in Alpe-Adria-Danube Region, RAAD 2006 (I. Rudas, Ed.). Budapest Tech, Budapest, 2006a, pp. 234-240

[14] Kelemen, J.: Agents from Functional-Computational Perspective. *Acta Polytechnica Hungarica* **3** (2006b) pp. 37-54

[15] Ronald, E. M. A., Sipper, M., Capcarrére, M. S.: Design, Observation, Surprise! A Test of Emergence. *Artificial Life* **5** (1999) 225-239

[16] Rozenberg, G., Salomaa, A.: *The Mathematical Theory of L Systems*. Academic Press, New York, 1980

[17] Searle, J.: *The Rediscovery of the Mind*. The MIT Press, Cambridge, Mass., 1992

[18] Van Gulick, R.: Reduction, Emergence and Other Recent Options on the Mind/Body Problem. In: *The Emergence of Consciousness* (A. Freeman, Ed.) Imprint Academic, Thorverton, 2001, pp. 1-34

[19] Wätjen, D.: Function-Dependent Teams in Eco-Grammar Systems. *Theoretical Computer Science* **306** (2003) 39-53

[20] Zeman, A.: *Consciousness – A User's Guide*. Yale University Press, New Haven, 2002

Graphical Addenda in the Technological Area of the Nodular Iron Cast Rolls Production

Imre Kiss, Stefan Maksay

Faculty of Engineering – Hunedoara, University "Politehnica" Timişoara
Revolutiei, 5, RO-331128 Hunedoara, Romania
ikiss@fih.upt.ro

Abstract: The technical conditions, which are imposed to the cast iron rolls in the exploitation period, are very different and often contradictory. The obtaining of various physical and mechanical properties in the different points of the same foundry product meets difficult technological problems in the industrial condition. One of the parameters, which determine the structure of the irons destined for rolls casting, is the chemical composition. If we not respect this composition, which guaranties the exploitation properties of the each roll in the stand of rolling mill, it will lead to rejection. Alloying elements have in principle the same influence on structure and properties. This paper suggest a mathematical interpretation of the influence of the main alloy elements over the mechanical characteristics (the hardness on the crust and on the necks of the rolls) of this nodular irons, resulting the average values and average square aberration of the variables HB, and the main alloying elements (Cr, Ni, Mo), the equations of the hyper surface in the four dimensional space. For the statistical and mathematical analysis, there were used 23 industrial cases. The resulted surfaces, belonging to the three-dimensional space, can be represented and, therefore, interpreted by technologists. Knowing these level curves allows the correlation of the values of the twos independent variables so that the hardness can be obtained in between the requested limits. The paper presents the results of some researches regarding the chemical composition of the nodular irons destined for casting half-hard rolls. It is presented, in graphical form, used the Matlab area, the influence of the main alloying elements upon the hardness, and measured on the necks and the core of the rolls, respectively on the working surface (body) of these very important rolling mill components The realization of an optimal chemical composition can constitute a technical efficient mode to assure the exploitation properties, the material from which the rolling mills cylinders are manufactured having an important role in this sense. From this point of view is applied the mathematical molding, which is achieved starting from the differentiation on rolls component parts, taking into consideration the industrial data obtained from the hardness mensuration on rolls, which recommends the hardness, for different chemical compositions.

Keywords: iron rolls, alloying elements, hardness, modelling, graphical addenda

1 Introduction

The nodular graphite cast iron is considered as one of the most versatile roll materials nowadays. This type of material may be used to produce large scale rolls in double pouring process, the barrel of rolls has high hardness while the neck has high toughness, so this type of rolls exhibits the properties of high thermal stability and resistance to wear. The rolls must present high hardness at the crust of rolls and lower hardness in the core and on the necks, adequate with the mechanical resistance and in the high work temperatures. If in the crust the hardness is assured by the quantities of cementite from the structure of the irons, the core of the rolls must contain graphite to assure these properties. One of the parameters, which are determined the structure of the irons destined for rolls casting, is the chemical composition. If we do not respect this composition, which are guarantied the exploitation properties of the each roll in the stand of rolling mill, leads to rejection of this.

All FNS type rolls are alloyed especially with chrome, nickel and molybdenum, in different percentages. The irons destined to these cast rolls belong to the class of low-alloyed irons, with reduced content of these elements. The technological instructions firmly state the elements required to rise the quality of rolls. In this case, the contents of these elements stand between large limits. Also, the contents of these alloying elements can be reduced due to the strong effect of the magnesium from the nodulising agent, upon the graphite's structure and form.

This study analyses iron rolls cast in the simplex procedure, in combined forms (iron chill, for the crust and moulding sand, for the necks of the rolls). The research included rolls from the half-hard class, with hardness, between 33…59 Shore units (219…347 Brinell units) for the 0 and 1 hardness class, measured on the crust, respectively 59…75 Shore units (347…550 Brinell units), for the class 2 of hardness.

This study is required because of the numerous defects, which cause rejection, since the phase of elaboration of these irons, destined to cast rolls. According to the previous presentation, it results that one of the most important reject categories is due to the inadequate hardness of the rolls. The research includes half-hard cast rolls, from nodular graphite irons (type FNS), hardness class 1 and 2, with the half-hard crust of 40…150 mm depth. All these types of rolls have high strength, excellent thermal properties and resistance.

The statistical methods of the analysis do not solve a whole series of appearances regarding to the decisions model to establish the management of the process. For this reason, in parallel with the statistical methods, was developed the methods of optimization. As part as the basic experiment, through the regression analysis, it was aimed the determination of the mathematical functions form which connect the dependent variables u of the technological process with the free variables (the technological parameters) $x, y, z,\ldots$, meaning $u = f(x, y, z,\ldots)$, on the strength of some experimental determinations, this after it accomplished a dispersion analysis of these correlation data. The determination of what real coefficients enter into the expression $u = f (x, y, z,\ldots)$ is done, in the vast majority of the cases, through the method of the smallest squares. Depending on the number of free variables (the technological parameters) that we consider, it was chosen the analysis of multiple regressions studying the influence of free variables *x, y, z,…* upon the dependent variable *u*. In this sense, it was aimed to establish calculus methodologies of values for the technological parameters in the manufacturing process of the semihard rolling mill cylinders, obtained through the simplex classical cast of the iron with nodular graphite, for which the mechanical features of rolling mill cylinders have the required values.

Table 1

Recommended hardness of half-hard cast iron rolls

Roll Types	Class of Hardness	Recommended hardness for these rolls			
		on the surface of rolls		on core and necks of rolls	
		[Shore]	[Brinell]	[Shore]	[Brinell]
FNS	0	33…42	218…286	30…40	195…271
FNS	1	43…59	294…347	30…40	195…271
FS	2	59…68	420…491	35…45	218…309
FNS	2	69…75	499…550	35…45	218…309

Table 2

Recommended chemical compositionof the half-hard cast iron rolls

Type	The basic chemical composition [%]				
	C	Si	Mn	P	S
FS	2.9...3.6	0.3...1.2	max 0.6	max 0.15	max 0.1
FNS	3.0...3.5	1.2...2.5	0.1...0.7	max 0.15	max 0.02

Type	The alloying elements and the nodulising agent [%]			
	Ni	Cr	Mo	Mg
FS	max 0.6	max 0.5	0.3...0.5	-
FNS	1.5...2.5	max 0.8	0.3...0.5	0.02...0.04

2 Interpretation and Simulation

All FNS type rolls are alloyed especially with chrome, nickel and molybdenum, in different percentages. The irons destined to these cast rolls belong to the class of low alloyed irons, with reduced content of these elements. The technological instructions firmly state the elements required to rise the quality of rolls.

In this case, the contents of these elements stand between large limits. Also, the contents of these alloying elements can be reduced due to the strong effect of the magnesium from the nodulising agent, upon the graphite's structure and form.

The two-dimensional representations can not present only that than of tendencies of influences, through his diminution or the enlargement of the feature characteristics, and the polynomial functions just appreciate, on ensemble, the influence of the chemical elements upon hardness in different in points of rolls, indicating just the limits of variation. For this reason is enforced an analysis of the cumulated influences of elements upon the hardness in different in parts of rolls, which study will be the study of the further researches.

Therefore, we suggest a mathematical interpretation of the influence of the main alloy elements over the mechanical characteristics (the hardness on the crust of the rolls) of this nodular irons, resulting the average values and average square aberration of the variables HB, and the main alloying elements (Cr, Ni, Mo), the equations of the hyper surface in the four dimensional space. For the statistical and mathematical analysis, there were used 23 industrial cases.

Table 3

The experimental mesured data, the theoretical determined data and the error

No.	Alloying elements [%]			The hardness on the body			The hardness on the necks		
	Ni	Cr	Mo	*	**	error	*	**	error
1.	1.99	0.62	0.22	342	333.71	8.30	270	262.39	7.60
2.	2.21	0.64	0.26	341	319.33	21.66	269	263.75	5.25
3.	2.16	0.57	0.25	326	326.31	- 0.29	266	265.57	4.32
4.	2.16	0.53	0.24	328	324.08	3.91	256	262.61	- 6.60
5.	1.59	0.38	0.22	292	289.46	2.53	249	244.5	4.49
6.	2.24	0.51	0.28	306	300.86	5.13	269	258.32	10.68
7.	1.82	0.36	0.26	291	288.48	2.51	246	245.84	0.16
8.	1.97	0.52	0.27	282	309.53	- 27.52	242	257.95	- 15.95
9.	2.22	0.66	0.21	322	331.03	- 9.03	266	265.49	0.50
10.	1.49	0.63	0.22	312	312.35	- 0.35	219	223.71	- 4.70
11.	1.63	0.51	0.27	284	294.99	- 10.98	238	239.9	- 1.90
12.	1.55	0.39	0.25	299	295.61	3.39	251	246.71	4.29
13.	1.52	0.47	0.21	286	307.05	- 21.05	241	246.96	- 5.95
14.	1.63	0.45	0.23	296	311.37	- 15.37	244	252.09	-8.08
15.	1.65	0.43	0.18	280	265.79	14.21	239	235.33	3.67
16.	2.11	0.68	0.27	289	300.02	- 11.02	247	249.77	- 2.76
17.	2.19	0.67	0.24	321	330.17	- 9.16	253	265.16	- 12.16
18.	1.62	0.44	0.21	297	301.49	- 4.48	242	248.65	- 6.65
19.	1.63	0.49	0.25	352	311.79	40.21	276	249.19	26.81
20.	1.66	0.42	0.27	296	293.05	2.95	238	247.5	- 9.49
21.	2.08	0.72	0.23	328	327.38	0.62	258	255.56	2.44
22.	1.94	0.64	0.24	339	327.52	11.48	264	255.99	8.00
23.	1.92	0.4	0.22	282	289.66	- 7.66	242	242.08	- 0.07

* experimental measured data

** theoretical determined data

The variables variation limits are: Ni = 1.49…2.24; Cr = 0.36…0.72; Mo = 0.18…0.28, and the hardness variation limits are $HB_{(necks)}$ = 219…276 and $HB_{(body)}$ = 282…352. Therefore, the graphical representation limits, for this modelling case, are presented in *Table 4*.

Table 4

lim Ni_{inf}	lim Ni_{sup}	lim Cr_{inf}	lim Cr_{sup}	lim Mo_{inf}	lim Mo_{sup}
1.61	2.11	0.40	0.67	0.19	0.27

The middle values for the three variables (Ni, Cr, Mo) and the hardness (HB), necessary for the calculation of the optimal form of modelling are:

$$Ni_{med} = 1.86;\ Cr_{med} = 0.52;\ Mo_{med} = 0.23;$$

and

$$HB_{(necks)med} = 251.52;\ HB_{(body)med} = 308.32;$$

Next, there are shown the results of the multidimensional processing of experimental data. For that purpose, we searched for a method of modelling the dependent variables u depending on the independent variables x, y, z:

$$u = c_1\cdot x^2+c_2\cdot y^2+c_3\cdot z^2+c_4\cdot x\cdot y+c_5\cdot y\cdot z+c_6\cdot z\cdot x+c_7\cdot x+c_8\cdot y+c_9\cdot z+c_{10} \quad (1)$$

The optimal modelling's form is given by the equations:

$$HB_{(body)} = -69.2668\ Ni^2 - 843.9321\ Cr^2 - 13082.6971\ Mo^2 + 258.4342\ Ni\cdot Cr - 3258.4415\ Cr\cdot Mo + 757.2487\ Mo\cdot Ni - 45.2572\ Ni + 1278.2053\ Cr + 6349.4428\ Mo - 739.6223 \quad (2)$$

$$HB_{(necks)} = -77.1259\ Ni^2 - 678.1307\ Cr^2 - 4915.8057\ Mo^2 + 384.4321\ Ni\cdot Cr - 1990.8226\ Cr\cdot Mo + 646.2006\ Mo\cdot Ni - 39.5771\ Ni + 471.3705\ Cr + 2131.6892\ Mo - 101.7176 \quad (3)$$

where the correlation coefficients are:

$$rf_{HB(body) = f(Ni, Cr, Mo)} = 0.77;\ \text{and}\ rf_{HB(necks) = f(Ni, Cr, Mo)} = 0.76$$

and the aberrations from the regression surface are:

$$sf_{HB(body) = f(Ni, Cr, Mo)} = 13.96;\ \text{and}\ sf_{HB(necks) = f(Ni, Cr, Mo)} = 8.73$$

3 Presentation the Obtained Results

In the technological field, the behaviour of these hyper surfaces in the vicinity of the saddle point, or of the point where three independent variables take their average value, can be studied only tabular, which means that the independent variables are attributed values on spheres concentric to the studied point. Because these surfaces cannot be represented in the three-dimensional space, the independent variables were successively replaced with their average values. This is how the following equations were obtained.

$$HB_{(body)}Ni_{med} = -843.9321\,Cr^2 - 13082.6971\,Mo^2 - 3258.4415\,Cr\cdot Mo + 1761.1402\,Cr + 7764.5101\,Mo - 1066.0756 \quad (4)$$

$$HB_{(body)}Cr_{med} = -13082.6971\,Mo^2 - 69.2668\,Ni^2 + 757.2487\,Mo\cdot Ni + 4630.9691\,Mo + 91.0387\,Ni - 300.2406 \quad (5)$$

$$HB_{(body)}Mo_{med} = -69.2668\,Ni^2 - 843.9321\,Cr^2 + 258.4342\,Ni\cdot Cr + 135.8241\,Ni + 499.0128\,Cr + 30.6111 \quad (6)$$

$$HB_{(necks)}Ni_{med} = -678.1307\,Cr^2 - 4915.8057\,Mo^2 - 1990.8226\,Cr\cdot Mo + 1189.7571\,Cr + 3339.2414\,Mo - 445.0005 \quad (7)$$

$$HB_{(necks)}Cr_{med} = -4915.8057\,Mo^2 - 77.1259\,Ni^2 + 646.2006\,Mo\cdot Ni + 1081.7467\,Mo + 163.1691\,Ni - 41.7373 \quad (8)$$

$$HB_{(necks)}Mo_{med} = -77.1259\,Ni^2 - 678.1307\,Cr^2 + 384.4321\,Ni\cdot Cr + 114.9492\,Ni - 4.6957\,Cr + 126.9318 \quad (9)$$

4 Presentation the Graphical Addenda

These surfaces, belonging to the three-dimensional space, can be represented and, therefore, interpreted by technologists. Knowing these level curves allows the correlation of the values of the twos independent variables so that the hardness can be obtained in between the requested limits.

These diagrams are built for the average values of the parameters (Cr_{med}, Ni_{med}, Mo_{med}), only that through the representation of the diagrams for parameters values contained in the variations limits we can obtain adjusting diagrams, with which we can completely controlled the process.

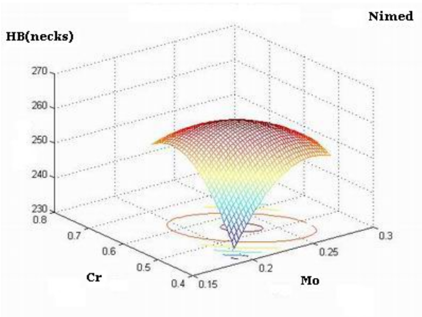


Figure 1
The regression surface
HB $_{(necks)}$ for Ni = Ni_{med}

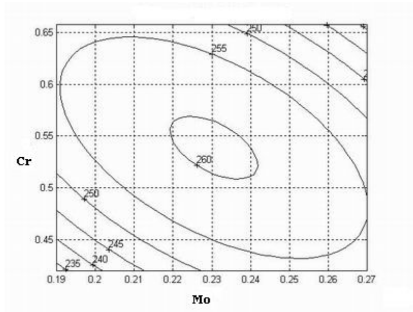


Figure 2
Level curves
HB $_{(necks)}$ = f(Ni_{med}, Cr, Mo)

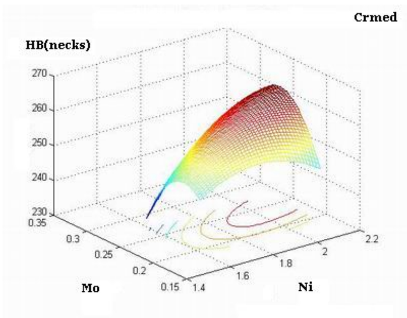


Figure 3
The regression surface
HB $_{(necks)}$ for Cr = Cr_{med}

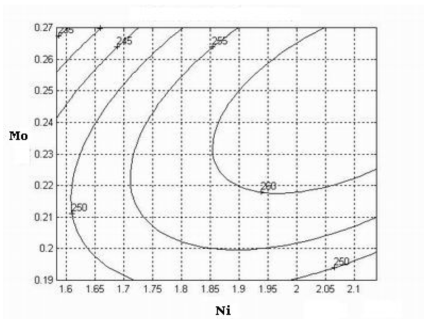


Figure 4
Level curves
HB $_{(necks)}$ = f(Ni, Cr_{med}, Mo)

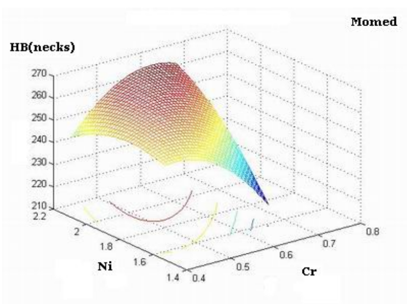


Figure 5
The regression surface
HB $_{(necks)}$ for Mo = Mo_{med}

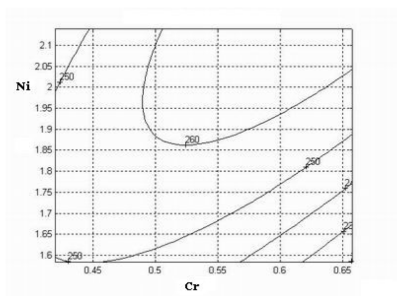


Figure 6
Level curves
HB $_{(necks)}$ = f(Ni, Cr, Mo_{med})

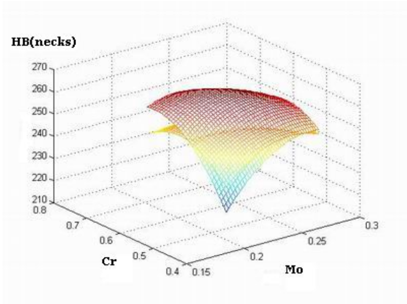


Figure 7
The volume variation of the regression surface HB $_{(necks)}$ for $Ni = Ni_{med}$

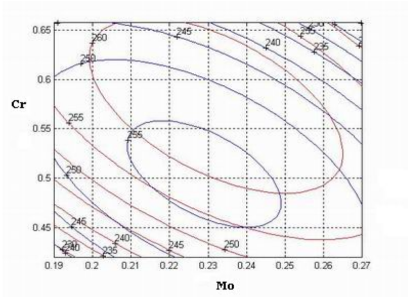


Figure 8
Level curves for volume variation of the regression surface HB $_{(necks)}$ for $Ni= Ni_{med}$

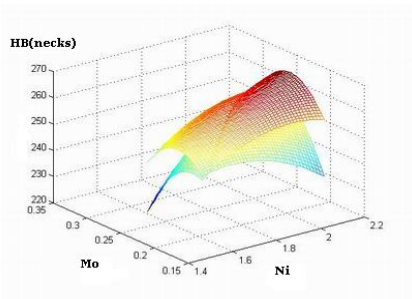


Figure 9
The volume variation of the regression surface HB $_{(necks)}$ for $Cr = Cr_{med}$

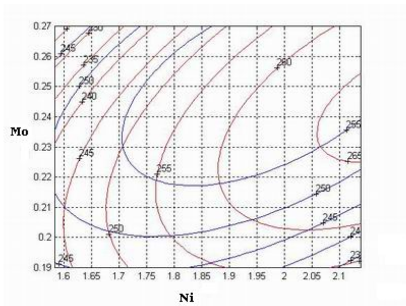


Figure 10
Level surves for volume variation of the regression surface HB $_{(necks)}$ for $Cr = Cr_{med}$

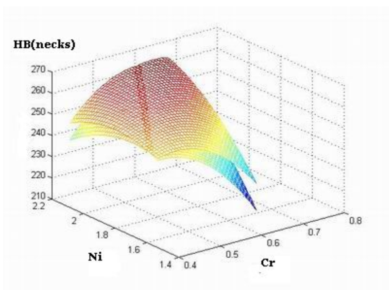


Figure 11
The volume variation of the regression surface HB $_{(necks)}$ for $Mo = Mo_{med}$

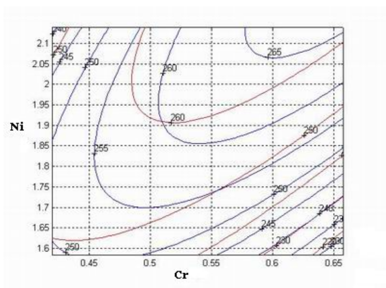


Figure 12
Level curves for volume variation of the regression surface HB $_{(necks)}$ for $Mo= Mo_{med}$

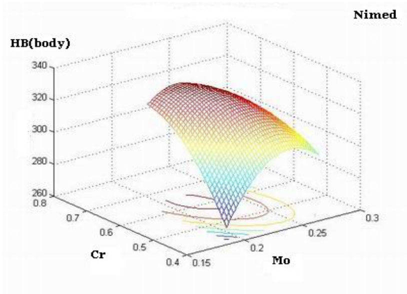


Figure 13
The regression surface
HB $_{(body)}$ for Ni = Ni_{med}

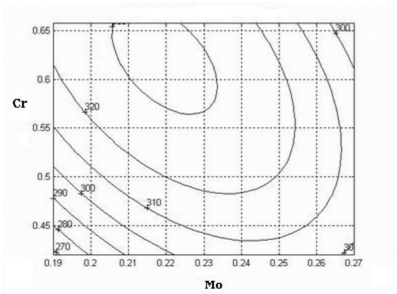


Figure 14
Level curves
HB $_{(body)}$ = f(Ni_{med}, Cr, Mo)

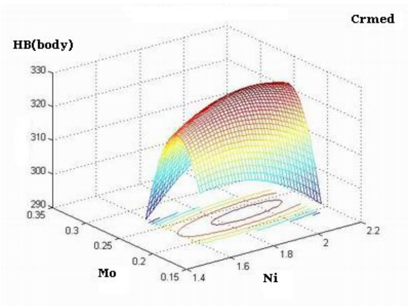


Figure 15
The regression surface
HB $_{(body)}$ for Cr = Cr_{med}

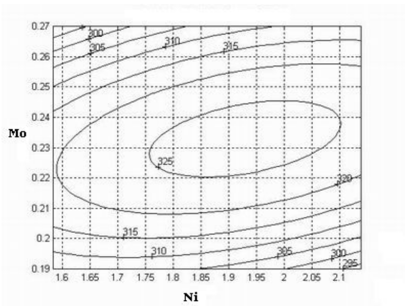


Figure 16
Level curves
HB $_{(body)}$ = f(Ni, Cr_{med}, Mo)

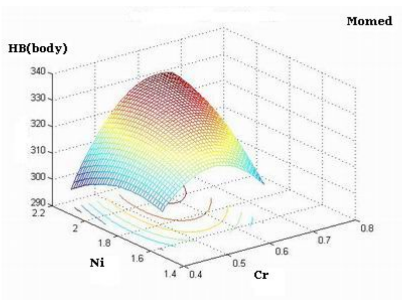


Figure 17
The regression surface
HB $_{(body)}$ for Mo = Mo_{med}

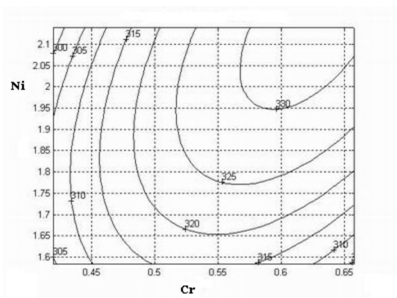


Figure 18
Level curves
HB $_{(body)}$ = f(Ni, Cr, Mo_{med})

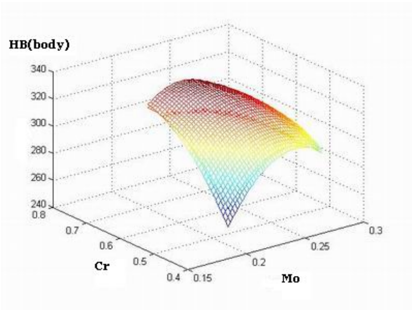


Figure 19
The volume variation of the regression surface HB $_{(body)}$ for Ni = Ni_{med}

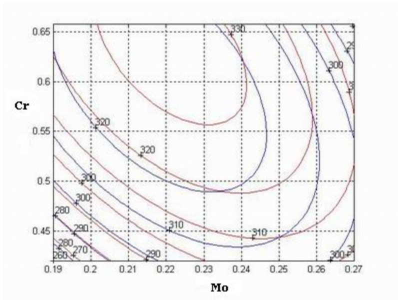


Figure 20
Level curves for volume variation of the regression surface HB $_{(body)}$ for Ni= Ni_{med}

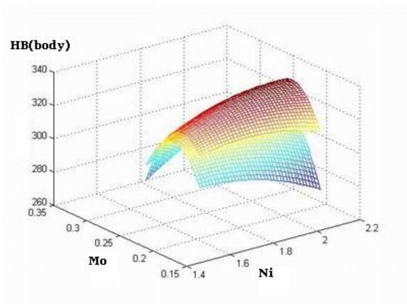


Figure 21
The volume variation of the regression surface HB $_{(body)}$ for Cr = Cr_{med}

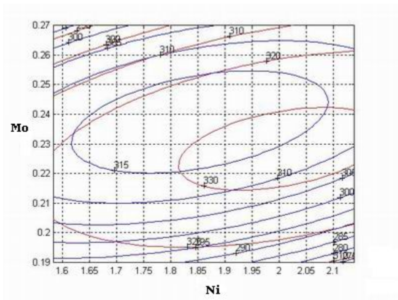


Figure 22
Level surves for volume variation of the regression surface HB $_{(body)}$ for Cr = Cr_{med}

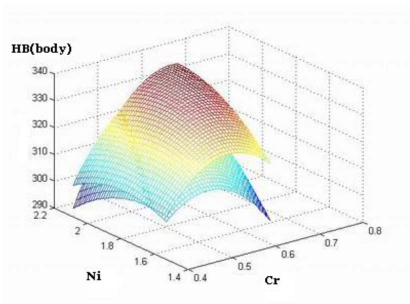


Figure 23
The volume variation of the regression surface HB $_{(body)}$ for Mo = Mo_{med}

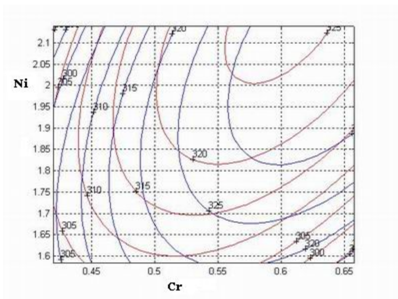


Figure 24
Level curves for volume variation of the regression surface HB $_{(body)}$ for Mo= Mo_{med}

Conclusions

The performed research had in view to obtain correlations between the hardness of the cast iron rolls (on the necks and on the body) and the representative alloying elements (Ni, Cr, Mo). The values processing were made using Matlab calculation program. Using this area we determinate some mathematical correlation, correlation coefficient and the deviation from the regression surface. This surface in the four-dimensional space (described by the general equation 1, and particular equations 2 and 3) admits a saddle point to which the corresponding value of hardness is an optimal alloying elements. Therefore, some conclusions could be presented:

- the existence of a saddle point inside the technological domain has a particular importance as it ensures stability to the process in the vicinity of this point, stability which can be either preferable of avoidable.
- the behaviour of this hyper surface in the vicinity of the stationary point (when this point belongs to the technological domain) or in the vicinity of the point where the three independent variables have their respective mean value, or in a point where the dependent function reaches its extreme value in the technological domain (but not being a saddle point) can be rendered only as a table, namely, assigning values to the independent variables on spheres which are concentrically to the point under study.
- as these surfaces cannot be represented in the three-dimensional space, we resorted to replacing successively one independent variable by its mean value. These surfaces (described by the equation 4…9), belonging to the three-dimensional space can be reproduced and therefore interpreted by technological engineers (Figures 1, 3, 5, respectively Figures 13, 15, 17). Knowing these level curves (Figures 2, 4, 6, respectively Figures 14, 16, 18) allows the correlation of the values of the two independent variables so that we can obtain the hardness within the required limits.
- the Figures 7, 9, 11, respectively Figures 19, 21, 23 presented the volume variation of the regression surfaces $HB_{(necks)}$ and $HB_{(body)}$ for one of the middle value of the variables Ni, Cr, Mo.
- in the Figures 8, 10, 12, respectively Figures 20, 22, 24, the level curves for the volume variation of the regression surfaces $HB_{(necks)}$ and $HB_{(body)}$, for the Ni_{med}, Cr_{med} and Mo_{med}, are presented in the graphical addenda.
- the usage of the Matlab area, can also be extended to the study of influences other chemical components (C, Si, Mn, S, P, Mg), and this influences upon the necks and the body of the rolling mills;

Engineering is concerned with the design of a solution to a practical problem. A scientist may ask *why* a problem arises, and proceed to research the answer to the question or actually solve the problem in his first try, perhaps creating a mathematical model of his observations. By contrast, engineers want to know *how* to solve a problem, and *how* to implement that solution. In other words, scientists attempt to *explain* phenomena, whereas engineers use any available knowledge, including that produced by science, to *construct* solutions to problems.

Often when engineers analyze a system to be controlled or optimized, they use a mathematical model. In analysis, engineers can build a descriptive model of the system as a hypothesis of how the system could work, or try to estimate how an unforeseeable event could affect the processes. As with all modern scientific and technological endeavors, computers and software play an increasingly important role. Numerical methods and simulations can help predict design performance more accurately than previous approximations. A mathematical model usually describes the processes by a set of variables and a set of equations that establish relationships between the variables. In this sense, the following research had in view to obtain correlations between the hardness of cast iron rolls and its chemical composition, defined by the representative alloying elements.

References

[1] Maksay, St,: Special Mathematics, Vols. I & II, Politehnica Publishing House, Timisoara, 2001

[2] Taloi, F., Bratu, C., Florian, E., Berceanu, E.: The Optimization of the Metallurgical Process, Didactical and Pedagogical Publishing House, Bucuresti, 1983

[3] Bonollo, F., Gramegna, N., Odorizzi, S.: Modellizzazioni dei processi di fonderia, in: Fonderia, 11/12, 1997, pp. 50-54

[4] Carless, P.: Present and Future Hot Strip Mill Finishing Train Work Rolls, Ironmaking and Steelmaking, Vol. 27, No. 1, 2000

[5] Frolish, M. F., Beynon, J. H.: Design Criteria for Rolling Contact Fatigue Resistance in Back-up Rolls, at: International Symposium “Rolls 2003” - International Convention Centre, Birmingham, United Kingdom, 2003

[6] Kerr, E. J., Webber, R.: Roll Performance - Present Overview and Look to the Future, at: International Symposium “Rolls 2003” - International Convention Centre, Birmingham, United Kingdom, 2003

[7] Kiss, I., Heput, T.: Mechanical Properties of the Cast Iron Rolls, assured by the chemical composition, in: Scientific Bulletin of the University Politehnica of Timisoara, Transactions on Mechanics, Tome 47, Fascicule 2, pp. 175-181, 2002

[8] Gorvany, V., Mamuzic, I., Radsinsky, V.: Influence of Cooling on the Resistance of Chrom-Nickel Iron Rolls, in: Metalurgija, 42, 2003, pp. 245-247

[9] Sofroni, L., Ripoşan, I., Cernat, C., Chira, I., Gradinaru, C: The Complex Alloying of Cast Iron used for Hard Crust Millrolls Casting in View to

Minimise the Nickel Content, la: 42nd International Foundry Congress, Lisabona, Portugalia – 1975, Giesserei – Praxis, 21, 1976, pp. 50-58

[10] Kiss, I., Maksay, St.: Model de optimizare a compozitiei chimice a fontelor cu grafit nodular (Cr, Ni, Mo) destinate turnarii cilindrilor de laminar, Analele Facultatii de Inginerie din Hunedoara, Tome IV, Fascicule 4, 2002, pp. 159-168

[11] Budaghiant, N. A., Karsski, V. E.: Cast Iron Rolls, Editura Tehnica, Bucuresti, 1984

[12] Kiss, I.: The Quality of Rolling Mills Cylinders Cast by Iron with Nodular Graphite, Mirton, Timisoara, 2005

[13] Kiss, I.: Researches Concerning the Improvement Quality of Rolling Mill Cylinders Cast from Cast-Iron with Nodular Graphite, Doctoral Thesis, Bucuresti, 2005

[14] Kiss, I., Hepuț, T.: The Hardness of the Cast Iron Rolls, Assured by the Alloy Elements and the Modification Treatment of the Graphite Nodularity, in: Scientific Bulletin of Politehnica University of Timişoara, Transactions on Mechanics, Tome 47, Fascicule 2, pp. 169-174, 2002

[15] Kiss, I.: Researches Regarding the Quality Assurance of the Rolling Mills Cast-Iron Cylinders through Mathematical Molding of the Manufacturing Process and the Experimental Study of Durability in Exploitation, National contract No. 5889/2005, Bucuresti

[16] Lipinski, D. M., Schaefer, W., Flender, E.: Numerical Modelling of the Solidification of Castings, Atti del Convegno "Modelling of Casting, Welding and Advanced Solidification Processes VI", 1993, pp. 389-398

[17] McCann, J.: Overview of Work Rolls for Cold Rolling, Ironmaking and Steelmaking, Vol. 27, No. 1, 2000

[18] Pellizzari, M., Molinari, A., Biggi, A.: Optimization of the Hot Roll Performances through Microstructural Tailoring, at Conference of University Trento, Brescia, Italy, 2002

[19] Stevens-Oost, B.: Roll Quality and Roll Cost Development: Where do We Come from and Where are We Going?, at International Symposium "Rolls 2003" - International Convention Centre, Birmingham, United Kingdom, 2003

[20] STAS 9432 – 85: Rolling Mills. Half-Hard Cast Iron Rolls Specification

Controlling Communication and Mobility by Types with Behavioral Scheme

Martin Tomášek

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice
Letná 9, 042 00 Košice, Slovakia
e-mail: martin.tomasek@tuke.sk

Abstract: This paper presents a type system of mobile ambients suitable for expressing communication and mobility of mobile code application. The main goal is to avoid ambiguities and possible maliciousness of some constructions in mobile ambients. The type system presents behavioral scheme that statically defines and checks access rights for authorization of ambients and threads to move. We proved the soundness theorem for the type system and we demonstrated the system by showing how to model mobile code paradigms.

Keywords: ambient calculus, mobile code, type system

1 Introduction

The calculus of mobile ambients [1] is based on concurrency paradigm represented by the π-calculus [2]. It introduces the notion of an ambient as a bounded place where concurrent computation takes place, which can contain nested subambients in a hierarchical structure, and which can move in and out of other ambients, i.e., up and down the hierarchy what rearranges the structure of ambients. The communication can only occur locally within each ambient through a common anonymous channel. Communication between different ambients has to be performed by movement and by dissolution of ambient boundaries.

The ambition of mobile ambients is in general to express mobile computation and mobile computing. Mobile ambients can express in natural way dynamic properties (communication and mobility) of mobile code systems, but there is still question of deeper control and verification of mobility properties (like access rights or mobility control). Usual approaches apply type systems which adds more properties to the pure calculus. Our paper presents the type system for ambient calculus that abstracts various properties of mobility and communication as a behavioral scheme of a process.

2 The Ambient Calculus

Mobile ambients model several computational entities: mobile agents, mobile processes, messages, packets or frames, physical or virtual locations, administrative and security domains in a distributed system and also mobile devices. This variety makes that in principle there are no differences among various kinds of software components when expressing by mobile ambients. In mobile ambients there are implicitly two main forms of entities, which we will respectively call *threads* and *ambients*. Threads are unnamed sequences of primitive actions to be executed sequentially, generally in concurrency with other threads. They can perform communication and drive their containers through the spatial hierarchy, but cannot individually go from one ambient to another. Ambients are named containers of concurrent threads. They can enter and exit other ambients, driven by their internal processes, but cannot directly perform communication. It is very important to ensure indivisibility and autonomous behavior of ambients (this is also important e.g. for objects).

Communication between ambients is represented by the movement of other ambient of usually shorter life, which have their boundaries dissolved by an *open* action to expose their internal threads performing local communication operations. Such capability of opening an ambient is potentially dangerous [3, 4, 5]. It could be used inadvertently to open and thus destroy the individuality of an object or mobile agent. Remote communication is usually emulated as a movement of such ambients (communication packages) in the hierarchy structure.

We explore a different approach, where we intend to keep the purely local character of communication so that no hidden costs are present in the communication primitives, but without *open* operation. This solves the problem of dissolving boundaries of ambients, but disables interactions of threads from separate ambients. We have to introduce new operation *move* for moving threads between ambients. The idea comes from mobile code programming paradigms [6] where moving threads can express strong mobility mechanism, by which the procedure can (through *move* operation) suspend its execution on one machine and resume it exactly from the same point on another (remote) machine. This solves the problem of threads mobility and by moving threads between ambients we can emulate communication between the ambients.

Such adaptations of mobile ambients operations we can express computational entities of mobile programs in more natural way. Another purpose for this approach is to prefer simplicity and understandability of designed type system for mobile ambients later on.

We define abstract syntax and operational semantics of our calculus. It is based on abstract syntax and operational semantics of ambient calculus including our new constructions.

2.1 Abstract Syntax

The abstract syntax of the terms of our calculus in Table 1 is the same as that of mobile ambients except for the absence of *open* and the presence of the new operation *move* for moving threads between ambients. We allow synchronous output and the asynchronous version is its particular case.

Table 1
Abstract syntax

$M ::=$	**mobility operations**
$\mid\ n$	name
$\mid\ in\ M$	move ambient into M
$\mid\ out\ M$	move ambient out of M
$\mid\ move\ M$	move thread into M
$\mid\ M.M'$	path
$P ::=$	**processes**
$\mid\ \mathbf{0}$	inactive process
$\mid\ P \mid P'$	parallel composition
$\mid\ !P$	replication
$\mid\ M[P]$	ambient
$\mid\ (\nu n : \mathbf{P}[\mathcal{B}])P$	name restriction
$\mid\ M.P$	action of the operation
$\mid\ \langle M \rangle.P$	synchronous output
$\mid\ (n : \mu).P$	synchronous input

We introduce types already in the term syntax, in the synchronous input and in the name restriction. The defined terms are not exactly the terms of our calculus, since the type constructions are not yet taken into account, this is done by the typing rules in the next section.

2.2 Operational Semantics

The operational semantics is given by reduction relation along with a structural congruence the same way as those for mobile ambients.

Each name of the process term can figure either as free (Table 2a) or bound (Table 2b).

Table 2
Free (a) and bound (b) names

a)	b)
$fn(n) = \{n\}$	$bn(n) = \varnothing$
$fn(in\ M) = fn(M)$	$bn(in\ M) = bn(M)$
$fn(out\ M) = fn(M)$	$bn(out\ M) = bn(M)$
$fn(move\ M) = fn(M)$	$bn(move\ M) = bn(M)$
$fn(M.M') = fn(M) \cup fn(M')$	$bn(M.M') = bn(M) \cup bn(M')$
$fn(\mathbf{0}) = \varnothing$	$bn(\mathbf{0}) = \varnothing$
$fn(P \mid P') = fn(P) \cup fn(P')$	$bn(P \mid P') = bn(P) \cup bn(P')$
$fn(!P) = fn(P)$	$bn(!P) = bn(P)$
$fn(M[P]) = fn(M) \cup fn(P)$	$bn(M[P]) = bn(M) \cup bn(P)$
$fn((\nu n : \mathbf{P}[\mathcal{B}])P) = fn(P) - \{n\}$	$bn((\nu n : \mathbf{P}[\mathcal{B}])P) = bn(P) \cup \{n\}$
$fn(M.P) = fn(M) \cup fn(P)$	$bn(M.P) = bn(M) \cup bn(P)$
$fn(\langle M \rangle.P) = fn(M) \cup fn(P)$	$bn(\langle M \rangle.P) = bn(M) \cup bn(P)$
$fn((n : \mu).P) = fn(P) - \{n\}$	$bn((n : \mu).P) = bn(P) \cup \{n\}$

We write $P\{n \leftarrow M\}$ for a substitution of the capability M for each free occurrences of the name n in the term P. The similarly for $M\{n \leftarrow M\}$.

Structural congruence is shown in Table 3 and it is standard for mobile ambients. The (SAmbNull) rule is added to get a form of garbage collection, because of absence of the *open* operation.

In addition, we identify processes up to renaming of bound names (α-conversion) as shown in Table 4. By this we mean that these processes are understood to be identical (e.g. by choosing an appropriate representation), as opposed to structurally equivalent.

Table 3
Structural congruence

equivalence:	
$P \equiv P$	(SRefl)
$P \equiv Q \Rightarrow Q \equiv P$	(SSymm)
$P \equiv Q, Q \equiv R \Rightarrow P \equiv R$	(STrans)
congruence:	
$P \equiv Q \Rightarrow P \mid R \equiv Q \mid R$	(SPar)
$P \equiv Q \Rightarrow !P \equiv !Q$	(SRepl)
$P \equiv Q \Rightarrow M[P] \equiv M[Q]$	(SAmb)
$P \equiv Q \Rightarrow (\nu n : \mathbf{P}[\mathcal{B}])P \equiv (\nu n : \mathbf{P}[\mathcal{B}])Q$	(SRes)
$P \equiv Q \Rightarrow M.P \equiv M.Q$	(SAct)
$P \equiv Q \Rightarrow \langle M \rangle.P \equiv \langle M \rangle.Q$	(SCommOut)
$P \equiv Q \Rightarrow (n : \mu).P \equiv (n : \mu).Q$	(SCommIn)
sequential composition (associativity):	
$(M.M').P \equiv M.M'.P$	(SPath)
parallel composition (associativity, commutativity and inactivity):	
$P \mid Q \equiv Q \mid P$	(SParComm)
$(P \mid Q) \mid R \equiv P \mid (Q \mid R)$	(SParAssoc)
$P \mid \mathbf{0} \equiv P$	(SParNull)
replication:	
$!P \equiv P \mid !P$	(SReplPar)
$!\mathbf{0} \equiv \mathbf{0}$	(SReplNull)
restriction and scope extrusion:	
$n \neq m \Rightarrow (\nu n : \mathbf{P}[\mathcal{B}])(\nu m : \mathbf{P}[\mathcal{B}'])P \equiv (\nu m : \mathbf{P}[\mathcal{B}'])(\nu n : \mathbf{P}[\mathcal{B}])P$	(SResRes)
$n \notin fn(Q) \Rightarrow (\nu n : \mathbf{P}[\mathcal{B}])P \mid Q \equiv (\nu n : \mathbf{P}[\mathcal{B}])(P \mid Q)$	(SResPar)
$n \neq m \Rightarrow (\nu n : \mathbf{P}[\mathcal{B}])m[P] \equiv m[(\nu n : \mathbf{P}[\mathcal{B}])P]$	(SResAmb)
$(\nu n : \mathbf{P}[\mathcal{B}])\mathbf{0} \equiv \mathbf{0}$	(SResNull)
garbage collection:	
$(\nu n : \mathbf{P}[\mathcal{B}])n[\mathbf{0}] \equiv \mathbf{0}$	(SAmbNull)

Table 4

α-conversion

$(\nu n:\mathbf{P}[\mathcal{B}])P = (\nu m:\mathbf{P}[\mathcal{B}])P\{n \leftarrow m\} \quad m \notin fn(P)$	(SAlphaRes)
$(n:\mu)P = (m:\mu)P\{n \leftarrow m\} \quad m \notin fn(P)$	(SAlphaCommIn)

The reduction rules in Table 5 are those for mobile ambients, with the obvious difference consisting in the synchronous output and the missing *open* operation, and with the new rule for the *move* operation similar to the "migrate" instructions for strong code mobility in software agents.

Table 5

Reduction rules

basic reductions:	
$n[in\ m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	(RIn)
$m[n[out\ m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	(ROut)
$n[move\ m.P \mid Q] \mid m[R] \rightarrow n[Q] \mid m[P \mid R]$	(RMove)
$(n:\mu).P \mid \langle M \rangle.Q \rightarrow P\{n \leftarrow M\} \mid Q$	(RComm)
structural reductions:	
$P \rightarrow Q \Rightarrow P \mid R \rightarrow Q \mid R$	(RPar)
$P \rightarrow Q \Rightarrow n[P] \rightarrow n[Q]$	(RAmb)
$P \rightarrow Q \Rightarrow (\nu n:\mathbf{P}[\mathcal{B}])P \rightarrow (\nu n:\mathbf{P}[\mathcal{B}])Q$	(RRes)
$P' \equiv P, P \rightarrow Q, Q \equiv Q' \Rightarrow P' \rightarrow Q'$	(RStruct)

3 Type System

From the huge amount of complex behavioral properties of mobile processes we abstract (extract) the type system that is simple enough to be easily used for expressing communication and mobility properties of mobile ambients. The main goal of our abstraction was the control of communication and mobility. We defined some kind of access rights for movement of threads and ambients. Usual approach presents type systems with dependent types. We defined process types and operation types that are related to a behavioral scheme of the process. The behavioral scheme is a construction which controls the communication and mobility properties of the process.

3.1 Types and Behavioral Scheme

We define communication types where both peers, receiver and sender, must be of the same message type. This allows to keep the sense of communication. It also secures the communication while only exchange of the correct messages is allowed.

The restriction of the mobility operations is defined by types applying a *behavioral scheme*. The scheme allows setting up the access rights for traveling of threads and ambients in the ambient hierarchy space of the system.

Types are defined in Table 6 where we present communication types and message types.

Table 6
Types

$\kappa ::=$	**communication type**
$\mid \perp$	no communication
$\mid \mu$	communication of messages of type μ
$\mu ::=$	**message type**
$\mid \mathbf{P}[\mathcal{B}]$	process with behavioral scheme $\mathcal{B}$
$\mid \mathbf{O}[\mathcal{B} \mapsto \mathcal{B}']$	operation which changes behavioral scheme $\mathcal{B}$ to $\mathcal{B}'$

The behavioral scheme is the structure $\mathcal{B} = (\kappa, Reside, Pass, Move)$ which contains four components:

- κ is the communication type of the ambient's threads
- $Reside$ is the set of behavioral schemes of other ambients where the ambient can stay
- $Pass$ is the set of behavioral schemes of other ambients that ambient can go through, it must be $Pass \subseteq Reside$
- $Move$ is the set of behavioral schemes of other ambients where ambient can move its containing thread

3.2 Typing Rules

Type environment is defined as a set $\Gamma = \{n_1 : \mu_1, \ldots, n_l : \mu_l\}$ where each $n_i : \mu_i$ assigns a unique type μ_i to a name n_i.

The domain of the type environment is defined by:

1. $Dom(\varnothing) = \varnothing$
2. $Dom(\Gamma, n : \mu) = Dom(\Gamma) \cup \{n\}$

We define two type formulas for our ambient calculus:

1. $\Gamma \vdash M : \mu$
2. $\Gamma \vdash P : \mathbf{P}[\mathcal{B}]$

Typing rules are shown in Table 7 and they are used to derive type formulas of ambient processes. We say the process is *well-typed* when we are able to derive a type formula for it using our typing rules. Well-typed processes respect the communication and mobility restrictions defined in all behavioral schemes of the system. It means such a process has the correct behavior. The type assignment system is clearly syntax-directed and keeps the system simple enough.

Table 7

Typing rules

$\dfrac{n : \mu \in \Gamma}{\Gamma \vdash n : \mu}$	(TName)
$\dfrac{\Gamma \vdash M : \mathbf{P}[\mathcal{B}] \quad \mathcal{B} \in Pass(\mathcal{B}')}{\Gamma \vdash in\ M : \mathbf{O}[\mathcal{B}' \mapsto \mathcal{B}']}$	(TIn)
$\dfrac{\Gamma \vdash M : \mathbf{P}[\mathcal{B}] \quad \mathcal{B} \in Pass(\mathcal{B}') \quad Reside(\mathcal{B}) \subseteq Reside(\mathcal{B}')}{\Gamma \vdash out\ M : \mathbf{O}[\mathcal{B}' \mapsto \mathcal{B}']}$	(TOut)
$\dfrac{\Gamma \vdash M : \mathbf{P}[\mathcal{B}] \quad \mathcal{B} \in Move(\mathcal{B}')}{\Gamma \vdash move\ M : \mathbf{O}[\mathcal{B} \mapsto \mathcal{B}']}$	(TMove)
$\dfrac{\Gamma \vdash M : \mathbf{O}[\mathcal{B}'' \mapsto \mathcal{B}'] \quad \Gamma \vdash M' : \mathbf{O}[\mathcal{B} \mapsto \mathcal{B}'']}{\Gamma \vdash M.M' : \mathbf{O}[\mathcal{B} \mapsto \mathcal{B}']}$	(TPath)
$\dfrac{}{\Gamma \vdash \mathbf{0} : \mathbf{P}[\mathcal{B}]}$	(TNull)
$\dfrac{\Gamma \vdash P : \mathbf{P}[\mathcal{B}] \quad \Gamma \vdash P' : \mathbf{P}[\mathcal{B}]}{\Gamma \vdash P \mid P' : \mathbf{P}[\mathcal{B}]}$	(TPar)
$\dfrac{\Gamma \vdash P : \mathbf{P}[\mathcal{B}]}{\Gamma \vdash !P : \mathbf{P}[\mathcal{B}]}$	(TRepl)
$\dfrac{\Gamma \vdash P : \mathbf{P}[\mathcal{B}] \quad \Gamma \vdash M : \mathbf{P}[\mathcal{B}] \quad \mathcal{B}' \in Reside(\mathcal{B})}{\Gamma \vdash M[P] : \mathbf{P}[\mathcal{B}']}$	(TAmb)

$\dfrac{\Gamma, n : \mathbf{P}[\mathcal{B}'] \vdash P : \mathbf{P}[\mathcal{B}]}{\Gamma \vdash (\nu n : \mathbf{P}[\mathcal{B}'])P : \mathbf{P}[\mathcal{B}]}$	(TRes)
$\dfrac{\Gamma \vdash M : \mathbf{O}[\mathcal{B} \mapsto \mathcal{B}'] \quad \Gamma \vdash P : \mathbf{P}[\mathcal{B}]}{\Gamma \vdash M.P : \mathbf{P}[\mathcal{B}']}$	(TAct)
$\dfrac{\Gamma \vdash P : \mathbf{P}[\mathcal{B}] \quad \Gamma \vdash M : \mu \quad \kappa(\mathcal{B}) = \mu}{\Gamma \vdash \langle M \rangle.P : \mathbf{P}[\mathcal{B}]}$	(TCommOut)
$\dfrac{\Gamma, n : \mu \vdash P : \mathbf{P}[\mathcal{B}] \quad \kappa(\mathcal{B}) = \mu}{\Gamma \vdash (n : \mu).P : \mathbf{P}[\mathcal{B}]}$	(TCommIn)

3.3 Soundness of the System

The usual property of subject reduction holds, which guarantees the soundness of the system by ensuring that typing is preserved by computation.

Soundness theorem: Let $\Gamma \vdash P : \mathbf{P}[\mathcal{B}]$ for some $\mathcal{B}$. Then:

1 $P \equiv Q$ implies $\Gamma \vdash Q : \mathbf{P}[\mathcal{B}]$

2 $P \to Q$ implies $\Gamma \vdash Q : \mathbf{P}[\mathcal{B}]$

Proof: The proof is standard, by induction on the derivations of $P \equiv Q$ and $P \to Q$. Let's consider only rule (RMove):

We assume $P = n[\mathit{move}\ m.P' \mid P''] \mid m[P''']$, $Q = n[P''] \mid m[P' \mid P''']$, and $\Gamma \vdash n[\mathit{move}\ m.P' \mid P''] \mid m[P'''] : \mathbf{P}[\mathcal{B}]$. This is given by (TPar), so that $\Gamma \vdash n[\mathit{move}\ m.P' \mid P''] : \mathbf{P}[\mathcal{B}]$ and $\Gamma \vdash m[P'''] : \mathbf{P}[\mathcal{B}]$. These are given by (TAmb), so that $\Gamma \vdash n : \mathbf{P}[\mathcal{B}_n]$, $\Gamma \vdash \mathit{move}\ m.P' \mid P'' : \mathbf{P}[\mathcal{B}_n]$ and $\mathcal{B} \in \mathit{Reside}(\mathcal{B}_n)$ for some $\mathcal{B}_n$, and $\Gamma \vdash m : \mathbf{P}[\mathcal{B}_m]$, $\Gamma \vdash P''' : \mathbf{P}[\mathcal{B}_m]$ and $\mathcal{B} \in \mathit{Reside}(\mathcal{B}_m)$ for some $\mathcal{B}_m$. This ise given by (TPar), so that $\Gamma \vdash \mathit{move}\ m.P' : \mathbf{P}[\mathcal{B}_n]$, $\Gamma \vdash P'' : \mathbf{P}[\mathcal{B}_n]$ and this is given by (TAct), so that $\Gamma \vdash \mathit{move}\ m : \mathbf{O}[\mathcal{B}' \mapsto \mathcal{B}_n]$ and $\Gamma \vdash P' : \mathbf{P}[\mathcal{B}']$ for some $\mathcal{B}'$. This is given by (TMove), so that $\Gamma \vdash m : \mathbf{P}[\mathcal{B}_m]$, $\Gamma \vdash \mathit{move}\ m : \mathbf{O}[\mathcal{B}_m \mapsto \mathcal{B}_n]$ and $\mathcal{B}_m \in \mathit{Move}(\mathcal{B}_n)$, then $\mathcal{B}' = \mathcal{B}_m$ and $\Gamma \vdash P' : \mathbf{P}[\mathcal{B}_m]$. Then according (TAmb) $\Gamma \vdash n[P''] : \mathbf{P}[\mathcal{B}]$ where $\mathcal{B} \in \mathit{Reside}(\mathcal{B}_n)$ and $\Gamma \vdash m[P' \mid P'''] : \mathbf{P}[\mathcal{B}]$ where $\mathcal{B} \in \mathit{Reside}(\mathcal{B}_m)$ and we conclude $\Gamma \vdash n[P''] \mid m[P' \mid P'''] : \mathbf{P}[\mathcal{B}]$ from (TPar).

4 Expressing Mobile Code Paradigms

Now we can look to how our typed calculus can express mobile code paradigms. Let's assume three mobile code paradigms [7]:

- remote evaluation,
- code on demand, and
- mobile agent.

4.1 Remote Evaluation

Remote evaluation is performed when a client sends a piece of code to the server and server evaluates the code and client can get the results back from the server. Also very general client-server paradigm can be expressed similar way as remote evaluation.

We assume application of the server named *Server*, which executes transferred code *P* from the client application named *Client*. The result of the execution is sent back to the client as a message *M*.

$$Server = s[S]$$
$$Client = c[move\ s.P.move\ c.\langle M \rangle \mid (x : \mu).C]$$
$$System = Server \mid Client$$

In order to make the *System* well-typed we define following behavioral schemes of the processes in the system:

$$\mathcal{B} = (\bot, \emptyset, \emptyset, \emptyset)$$
$$\mathcal{B}_S = (\bot, \{\mathcal{B}\}, \emptyset, \emptyset)$$
$$\mathcal{B}_c = (\mu, \{\mathcal{B}\}, \emptyset, \{\mathcal{B}_s\})$$

As we can see schemes express that both *Server* and *Client* can be executed in the *System* and *Client* can move threads (code for remote evaluation) to the *Server*.

4.2 Code on Demand

Code on demand describes the situation where a client wants to perform a code that is presented by the server. Client asks for a code and server sends it to the client where it can be evaluated.

Similarly as for remote evaluation we assume application of the server named *Server*, which provides a code *P* to the client application named *Client*. Client application asks for the code and the result of execution is processed as message *M*.

$$Server = s[(p : \mathbf{O}[\mathcal{B}_c \mapsto \mathcal{B}_s]).p.P.\langle M \rangle \mid S]$$
$$Client = c[move\ s.\langle move\ c\rangle \mid (x : \mu).C]$$
$$System = Server \mid Client$$

In order to make the *System* well-typed we define following behavioral schemes of the processes in the system:

$$\mathcal{B} = (\bot, \varnothing, \varnothing, \varnothing)$$
$$\mathcal{B}_{\mathcal{S}} = (\mathbf{O}[\mathcal{B}_c \mapsto \mathcal{B}_s], \{\mathcal{B}\}, \varnothing, \{\mathcal{B}_c\})$$
$$\mathcal{B}_c = (\mu, \{\mathcal{B}\}, \varnothing, \{\mathcal{B}_s\})$$

As we can see schemes express that both *Server* and *Client* can be executed in the *System*. *Server* can receive path (sequence of movement operations) for moving the code to the *Client*. *Client* can send the request for the code to the *Server*.

4.3 Mobile Agent

Mobile agent is a paradigm where an autonomous code (agent) is sent from the client to the server. By autonomous we mean that the client and server do not need to synchronize the agent invocation and the agent is running independently and concurrently within the server's place.

We assume application of the server named *Server*, where the agent appication named *Agent* will be moved from its home application named *Home*. The process *P* of the agent is executed at the *Server* and after the execution, *Agent* is moved back *Home*. The movement of the *Agent* is defined by the path (sequence of *in*/*out* operations) which expresses travel plan of the agent.

$$Server = s[S]$$
$$Home = h[Agent \mid H]$$
$$Agent = a[out\ h.in\ s.P.out.s.in\ h]$$
$$System = Server \mid Home$$

In order to make the *System* well-typed we define following behavioral schemes of the processes in the system:

$$\mathcal{B} = (\bot, \varnothing, \varnothing, \varnothing)$$
$$\mathcal{B}_s = (\bot, \{\mathcal{B}\}, \varnothing, \varnothing)$$
$$\mathcal{B}_h = (\bot, \{\mathcal{B}\}, \varnothing, \varnothing)$$
$$\mathcal{B}_a = (\bot, \{\mathcal{B}, \mathcal{B}_s, \mathcal{B}_h\}, \{\mathcal{B}_s, \mathcal{B}_h\}, \varnothing)$$

As we can see schemes express that *Agent* can be executed either at the *Server* or *Home* places and also can move through those places.

Conclusions

We defined formal tool for expressing dynamics of mobile code applications, which is based on theory of mobile ambients. Presented changes to the ambient calculus are suitable for expressing different kinds of mobility and they avoid ambiguities and possible maliciousness of some constructions. The type system statically defines and checks access rights for authorization of ambients and threads to move by application of the process behavioral scheme. The usage of type system is limited by its very simplicity and it does not prevent more restrictive properties from being checked at runtime. We proved the soundness theorem for the type system and we demonstrated the system by showing how to model some common applications. We provided a simple language for distributed system of mobile agents. As an expressiveness test, we showed that well-known π-calculus of concurrency and mobility can be encoded in our calculus in a natural way [8].

Acknowledgement

This paper was supported by the grants: VEGA 1/2174/05, VEGA 1/2176/05, VEGA 1/3135/06, VEGA 1/4073/07, and KEGA 3/5166/07.

References

[1] Cardelli, L., Gordon, A. D.: Mobile Ambients. Theoretical Computer Science, Vol. 240, No. 1, 2000, pp. 177-213

[2] Milner, R., Parrow, J., Walker, D.: A Calculus of Mobile Processes, Parts 1-2, Information and Computation, Vol. 100, No. 1, 1992, pp. 1-77

[3] Levi, F., Sangiorgi, D.: Controlling Interference in Ambients. Proceedings of POPL'00, ACM Press, New York, 2000, pp. 352-364

[4] Bugliesi, M., Castagna, G.: Secure Safe Ambients. Proceedings of POPL'01, ACM Press, New York, 2001, pp. 222-235

[5] Bugliesi, M., Castagna, G., Crafa, S.: Boxed Ambients. In B. Pierce (ed.): TACS'01, LNCS 2215, Springer Verlag, 2001, pp. 38-63

[6] Fuggeta, A., Picco, G. P., Vigna, G.: Understanding Code Mobility. IEEE Transactions on Software Engineering, Vol. 24, No. 5, May 1998, pp. 342-361

[7] Ghezi, C., Vigna, G.: Mobile Code Paradigms and Technologies: A Case Study. Mobile Agents: 1st International Workshop MA'97, LNCS 1219, Springer-Verlag, 1997

[8] Tomasek, M.: Expressing Dynamics of Mobile Programs. PhD thesis, Technical University of Košice, 2004

Nonlinear Control of Induction Machines Using an Extended Kalman Filter

Salima Meziane, Riad Toufouti, Hocine Benalla

Laboratory of Electrical Engineering, Constantine University
Rue Bouaziz Saadi N°9
04000 Oum el Bouaghi
Algeria
E-mail: [meziane_elc, toufoutidz, benalladz]@yahoo.fr

Abstract: This paper, we propose the comparative study between the performances of the proposed controller and field oriented control. With an aim of improving the field oriented control, the control of the induction motor by Input-Output linearization techniques are used to track torque and rotor flux and the scheme is extended for speed control is presented. The methods are compared in terms of their ability to handle loads on the motor shaft, their speed tracking capability and their sensitivity to operating condition variations. To estimate the rotor flux, an open loop observer the Extended Kalman Filter (EKF) algorithm is used. A reduced dynamic motor model is used which reduces the computational requirements of the EKF. Simulation results show the effectiveness of the proposed method.

Keywords: nonlinear control, induction motors, field orientation, input output linearization, Kalman Filter

1 Introduction

Advanced control of electrical machines requires an independent control of magnetic flux and torque. For that reason it was not surprising, that the DC-machine played an important role in the early days of high performance electrical drive systems, since the magnetic flux and torque are easily controlled by the stator and rotor current, respectively.

The introduction of Field Oriented Control [1-5] meant a huge turn in the field of electrical drives, since with this type of control the robust induction machine can be controlled with a high performance. My orders have a great weakness opposite the exact knowledge of the reference mark (d, q), a low robustness against the parametric variations of the engine, and in addition with the mode of over peed where decoupling only becomes partial.

To improve the Field Oriented Control, full linearization state feedback control based on differential geometric theory [2], has been proposed in [7] for the electromagnetic torque control and in [8] for the adaptive speed control of a fifth order model of an induction motor. In this paper, a comparative study between the classical Field Oriented Control [4] and a newly proposed nonlinear controller has been carried out. The new controller is based on the theory of feedback linearization. The controller is used for the speed control of a fourth-order model of an induction motor. Since all the states are not available for direct measurement, a, open loop flux observer is proposed for the estimation of the rotor flux.

This paper is organized as follows. In Section 2, the dynamic model of the induction motor is described. The Field Oriented Control is reviewed in Section 3, and analysis of the necessity of a high performance control strategy for high speed ranges is carried on in the same section [1-5]. In Section 4, the nonlinear controller is designed for the speed and flux magnitude control of a fourth order model of an induction motor [1-8]. The observer required for the rotor flux estimation is presented in Section 5**.** Section 6 provides numerical simulation results, followed by the conclusion.

2 Induction Motor Modelling Using Vector Control Theory

One particular approach for the control of induction motors is the Field Oriented Control (FOC) introduced by Balaschke. Fig. 1 shows a block diagram of an indirect field-oriented control system for an induction motor [1-11]. In this system, the d-q coordinate's reference frame is locked to the rotor flux vector rotating at the stator frequency U, as shown in Fig. 1 [2]. This result in a decoupling of the variables so that flux and torque can be separately controlled by stator direct-axis current i_d, and quadrature-axis current i, respectively. The stator quadrature-axis reference i_{dsref} is calculated from torque reference input T_{ref} as [1-5]:

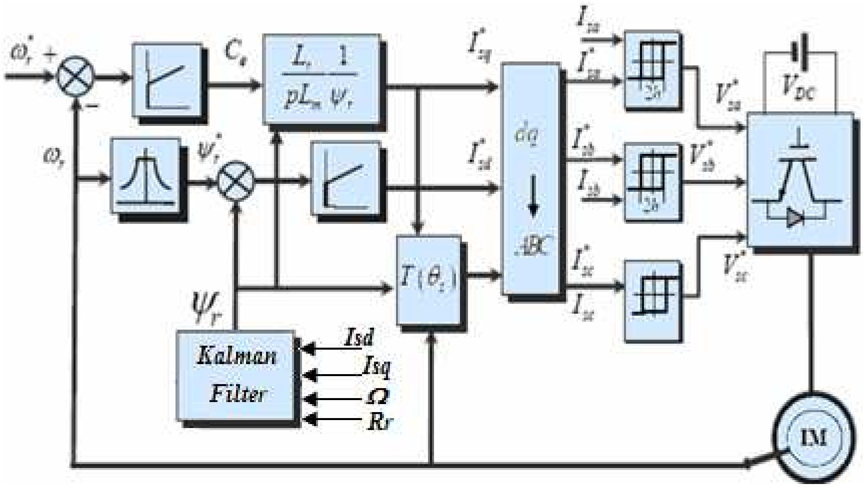


Figure 1

Indirect field-oriented control

The machine equations in the stator reference frame, written in terms of space vectors, are:

$$\overline{V}_s = R_s \overline{I}_s + \frac{d\overline{\varphi_s}}{dt} \tag{1}$$

$$0 = R_r \overline{I}_r + \frac{d\overline{\varphi_r}}{dt} - j\omega m \overline{\varphi}_r \tag{2}$$

$$\overline{\varphi}_s = L_s \overline{I}_s + M \overline{I}_r \tag{3}$$

$$\overline{\varphi}_r = L_r \overline{I}_r + M \overline{I}_r \tag{4}$$

$$\frac{d\Omega}{dt} = \frac{T - T_L}{J} \tag{5}$$

$$T = p\frac{M}{JL_r}(\varphi_{rd} i_{sq} - \varphi_{rq} i_{sd}) \tag{6}$$

Where is the pole pair number and

$$\sigma = 1 - \frac{M^2}{L_s L_r} \tag{7}$$

$$\varphi_r = (\varphi_{rd}, \varphi_{rq})^T \tag{8}$$

Assuming a rotor flux reference frame, and developing the previous equations with respect to the axis and axis components, leads to.

$$\frac{d\varphi_{rd}}{dt} + \frac{1}{\tau_r}\varphi_{rd} = \frac{M}{\tau_r} i_{sd} \tag{9}$$

$$T = p\frac{M}{JL_r}\varphi_{rd} i_{sq} \tag{10}$$

These equations represent the basic principle of the FOC: in the rotor flux reference frame, a decoupled control of torque and rotor flux magnitude can be achieved acting on the d and q axis stator current components, respectively. A block diagram of a basic DFOC scheme is presented in Fig. 1. The rotor flux estimation is carried out by [10].

$$\frac{d\overline{\varphi}_s}{dt} = \overline{v}_s - R_s \overline{i}_s \tag{11}$$

$$\overline{\varphi}_r = \frac{L_r}{M}(\overline{\varphi}_s - \sigma L_s \overline{i}_s) \tag{12}$$

The flux estimator has been considered to be ideal, being the effects due to parameter variations at low speed out of the major aim of this paper. The current controller has been implemented in the rotor flux reference frame using PI regulators with back emf compensation.

3 Technique Input-Output Linearizalion

The induction motor consists of three-phase stator windings and a rotor with short cut windings. Since the torque produced is a function of the difference between the mechanical speed and the angular speed of the supplied stator voltage, this results in a nonlinear model [1-7]. To reduce the complexity of a three-phase model, an equivalent two-phase representation is chosen. For the FOC this two-phase model is usually transformed in a rotating (d, q) reference frame [1, 3]. This allows a partial linearization of the model. This transformation is a source of problems but usually the FOC approach does not allow control the model in a stator fixed (α,β) reference frame [1]. Using nonlinear feedback allows to control the model in the stator fixed (α,β) reference frame avoiding the transformation in a rotating reference frame. The complete model in stator fixed (α,β) reference frame can is written in a linear form for a control in torque and flux:

$$S = \begin{cases} \dot{x} = f(x) + \sum_{i=1}^{p} g_i(x).u_i \\ y = \begin{bmatrix} h1(x) \\ h2(x) \end{bmatrix} = \begin{bmatrix} p\dfrac{M}{JL_r}(\varphi_{r\alpha} I_{s\beta} - \varphi_{r\beta} I_{s\alpha}) \\ \varphi_{r\alpha}^2 + \varphi_{r\beta}^2 \end{bmatrix} \end{cases} \tag{12}$$

Where

$$f(x) = \begin{bmatrix} -\gamma I_{s\alpha} + \dfrac{K}{Tr}\varphi_{r\alpha} + p\omega K\varphi_{r\beta} \\ -\gamma I_{s\beta} - p\omega K\varphi_{r\alpha} + \dfrac{K}{Tr}\varphi_{r\beta} \\ \dfrac{M}{Tr} I_{s\alpha} - \dfrac{1}{Tr}\varphi_{r\alpha} - p\omega\varphi_{r\beta} \\ \dfrac{M}{Tr} I_{s\beta} + p\omega\varphi_{r\alpha} - \dfrac{1}{Tr}\varphi_{r\beta} \\ p\dfrac{M}{JL_r}(\varphi_{r\alpha} I_{s\beta} - \varphi_{r\beta} I_{s\alpha}) - \dfrac{1}{J}(T_L) \end{bmatrix} \tag{13}$$

$$x = [I_{s\alpha} \quad I_{s\beta} \quad \varphi_{r\alpha} \quad \varphi_{r\beta} \quad \Omega]^T \;\; ; \; x = [U_{s\alpha} \quad U_{s\beta}]^T \tag{14}$$

$$g = \begin{bmatrix} \dfrac{1}{\sigma LS} & 0 & 0 & 0 & 0 \\ 0 & \dfrac{1}{\sigma LS} & 0 & 0 & 0 \end{bmatrix} \tag{15}$$

In let us introduce the following definitions:

$$Tr = \frac{L_r}{R_r} \quad ; \quad \sigma = 1 - \frac{M^2}{L_s L_r} \quad K = \frac{M}{\sigma L_s L_r} \quad ; \quad \gamma = \frac{Rs}{\sigma L_s} + \frac{R_r M^2}{\sigma L_s L_r^2} \tag{16}$$

We do not develop the details of input-output linearization techniques but directly show the application on the induction motor drive [1]. The quantities which will be controlled are differentiated with respect to time until the input *u* appears and the derivatives of the state variables are eliminated using the state-space representation of (12) [1-9].

The derivative of dregs of a function $h(x):\Re^n \rightarrow \Re$ along a field of vectors $f(x)=[f_1(x)\ldots f_n(x)]^T$ is given by:

$$L_f h(x)=\sum_{i=1}^{n}\frac{\partial h(x)}{\partial x_i}f_i(x) \tag{17}$$

Thus the derivatives of the outputs are given by

✓ ***For the first output*** $y_1 = C_{em}$

$$\dot{y}_1=\dot{h}_1(x)=L_f h_1(x)+L_g h_1(x)u \tag{18}$$

Let us calculate $L_f h_1(x)$ *et* $L_g h_1(x)$:

$$\bullet\ L_f h_1(x)=\sum_{i=1}^{n}\frac{\partial h_1(x)}{\partial x_i}f_i(x)=-p\frac{M}{L_r}[(\frac{1}{T_r}+\gamma)(\varphi_{r\alpha}I_{s\beta}-\varphi_{r\beta}I_{s\alpha}) + p\Omega(\varphi_{r\alpha}I_{s\alpha}+\varphi_{r\beta}I_{s\beta})+p\Omega K(\varphi_{r\alpha}^2+\varphi_{r\beta}^2)] \tag{19}$$

$$\bullet\ L_g h_1(x)=\sum_{i=1}^{n}\frac{\partial h_1(x)}{\partial x_i}g_i(x)=[-pK\varphi_{r\beta} \qquad pK\varphi_{r\alpha}] \tag{20}$$

✓ ***For the second output*** $y_1=\varphi_r^2$

$$\dot{y}_2=\dot{h}_2(x)=L_f h_2(x) \tag{21}$$

$$\ddot{y}_2=\ddot{h}_2(x)=L_f^2 h_2(x)+L_g L_f h_2(x)u \tag{22}$$

Let us calculate $L_f h_2(x), L_f^2 h_2(x)\, et\, L_g L_f h_2(x)$:

$$\bullet\ L_f h_2(x)=\sum_{i=1}^{n}\frac{\partial h_2(x)}{\partial x_i}f_i(x)=\frac{2}{L_r}[M(\varphi_{r\alpha}I_{s\alpha}+\varphi_{r\beta}I_{s\beta})-(\varphi_{r\alpha}^2+\varphi_{r\beta}^2)] \tag{23}$$

$$\bullet L_f^2 h_2(x) = \sum_{i=1}^{n} \frac{\partial (L_f h_2)(x)}{\partial x_i} f_i(x) = (\frac{4}{T_r^2} + \frac{2KM}{T_r^2})(\varphi_{r\alpha}^2 + \varphi_{r\beta}^2)$$
$$-(\frac{6M}{T_r^2} + \frac{2\gamma M}{T_r^2})(\varphi_{r\alpha} I_{s\alpha} + \varphi_{r\beta} I_{s\beta}) \tag{24}$$
$$+\frac{2pM\Omega}{T_r}(\varphi_{r\alpha} I_{s\beta} - \varphi_{r\beta} I_{s\alpha}) + \frac{2M^2}{T_r^2}(\varphi_{r\alpha}^2 + \varphi_{r\beta}^2)$$

$$\bullet L_g L_f h_2(x) = \sum_{i=1}^{n} \frac{\partial (L_f h_2)(x)}{\partial x_i} g_i(x) = [2R_r K\varphi_{r\alpha} \quad 2R_r K\varphi_{r\beta}] \tag{25}$$

The controller design is based on the fourth order dynamic model obtained from the *(d, q)* axis model of the motor under the field oriented assumptions so that either speed or flux magnitude control objective can be fulfilled [1].

The underlying design concept is to endow the closed loop system with high performance dynamics for high speed ranges while maximizing power efficiency and keeping the required stator voltage within the inverter ceiling limits [1]. In addition to filtering those control objectives, our control design aims to reduce the complexity of the control scheme, saving thereby the computation time of the control algorithm, which is an improvement over previous work found in the technical literature [1, 3]. The outputs to be controlled are the speed *w* and the square of the rotor flux magnitude = 4**.** The output vector is the sum of the relative degrees of the torque r_1=1 and flux r_2=2 is lower than the n=5 degree system *S*. we obtain a not observable dynamics of order 2.

Define the change of coordinates [11]:

$$\begin{aligned} z_1 &= h_1(x) \\ z_2 &= h_2(x) \\ z_3 &= L_f h_2(x) \\ z_4 &= \arctan(\frac{\varphi_{r\beta}}{\varphi_{r\alpha}}) \\ z_5 &= \Omega \end{aligned} \tag{26}$$

Let us note that in the configuration where $\sum_{i=1}^{p} ri \prec n$, only the variables z_1, z_2, and z_3 are fixed. The variables z_4 and z_5 are selected arbitrarily here; the followed approach is that of [1] considering z_4 the rotor angle of flux.

Thus the derivatives of the outputs are given in the new coordinate system *S* by:

$$
\begin{aligned}
\dot{z}_1 &= L_f h_1(x) + L_g h_1(x).u \\
\dot{z}_2 &= L_f h_2(x) = z_3 \\
\dot{z}_3 &= L_f^2 h_2(x) + L_g L_f h_2(x).u \\
\dot{z}_4 &= p z_5 + \frac{R_r}{p}\frac{z_1}{z_2} \\
\dot{z}_5 &= \frac{1}{J}(z_1 - T_L - f z_5)
\end{aligned} \tag{27}
$$

This system can be written as:

$$
\begin{bmatrix} \dot{z}_1 \\ \ddot{z}_2 \end{bmatrix} = \begin{bmatrix} L_f h_1 \\ L_f^2 h_2 \end{bmatrix} + D(x).u \tag{28}
$$

Where

$$
D(x) = \begin{bmatrix} L_f h_1 \\ L_g L_f h_2 \end{bmatrix} \tag{29}
$$

The decoupling matrix D(x) is singular if and only if φ_r^2 is zero which only occurs at the start up of the motor [11]. That is, to filtering this condition one can use in a practical setting, an open loop controller at the start up of the motor, and then switch to the nonlinear controller as soon as the flux goes up to zero. If the decoupling matrix is not singular, the nonlinear Sate feedback control is given by [1-7]:

$$
\begin{bmatrix} u_{s\alpha} \\ u_{s\beta} \end{bmatrix} = D^{-1} \begin{bmatrix} -L_f h_1 + v_1 \\ -L_f^2 h_2 + v_2 \end{bmatrix} \tag{30}
$$

With

$$
\begin{cases} v_1 = \dot{y}_1 \\ v_2 = \ddot{y}_2 \end{cases} \tag{31}
$$

This system can be schematized by the figure

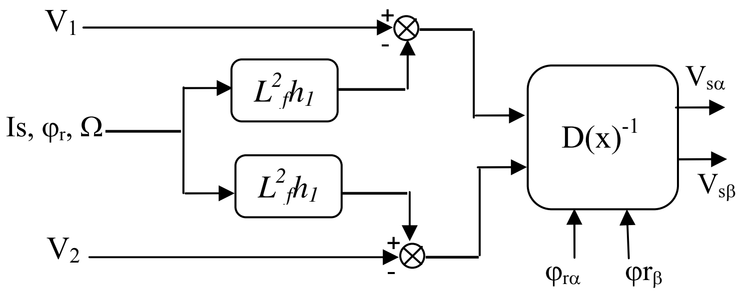


Figure 2
Nonlinear controllers

The dynamic ones $\dot{z}_4$ *et* $\dot{z}_5$ are made not observable by the return of linearization state. From the point of view of the state and not of the input-outputs, it should then be shown that the dynamic ones of zero and stable. In let us choose the balance point of following [1-8]:

$$[z_1, z_2, z_3, z_4, z_5]^T = [0, h, 0, z_4, z_5]^T \tag{32}$$

The dynamics of the zeros becomes:

$$\dot{z}_4 = pz_5 \tag{33}$$

$$\dot{Z}_5 = \frac{1}{J}(-T_L - fZ_5) \tag{34}$$

Where z_4 is the angle of rotor flux varying between 0 and 2π. The equation (34) and a stable first order are linear dynamics. The dynamics of zero is thus stable. The inversion of the D(*x*) matrix could in theory reveal a problem bus determine it is expressed according to the model of rotor flux [1].

$$\det(D) = -2pK^2 R_r(\varphi_{r\alpha}^2 + \varphi_{r\beta}^2) = -2pK^2 R_r \varphi_r^2 \tag{35}$$

If φ_r^2 the matrix is no null is invertible however the mathematical description of the singularity 3rd of starting is not problem irreversible. It is a question, indeed, of adopting a sentence of fluxing of the machine then to initialize the observer of flow, which is generally carried out. [1-5] In order to obtain a good continuation of flux and couple towards their respective reference, several strategies are possible. That given in [1-3], is to recall briefly here. The variables v_1 and v_2 can be calculated in the following way [1, 3]:

$$v_1 = -k_a(C_{em} - C_{emref}) + \dot{C}_{emref}$$
$$v_2 = -k_{b1}(\varphi_r^2 + \varphi_{rref}^2) - k_{b2}(\dot{\varphi}_r^2 + \dot{\varphi}_{rref}^2) + \ddot{\varphi}_{rref}^2 \tag{36}$$

The profit $K_{a,}$ K_{b1} and K_{b2} can be selected such as $k_a + S$ and $k_a + k_{b1}S + k_{b2}S^2$ are the polynomials of becoming HURWITZ not the equation of error of continuation:

$$\begin{cases} e_1 = C_{em} - C_{emref} \\ e_2 = \varphi_r^2 - \varphi_{rref}^2 \\ \dot{e}_1 + k_a e_1 = 0 \\ \ddot{e}_2 + k_{b1}e_2 + k_{b2}e_2 = 0 \end{cases} \tag{37}$$

Generally, an integral action is added in order to reject constant disturbances; as shown by Figure 3.

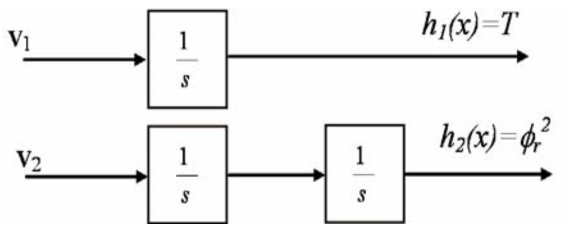


Figure 3
Linearized systems

Where T_{ref} and Φ_{ref} are torque and flux references. Note that the references for torque and rotor flux have to be once or twice differentiated. Thus we implement a second-order state variable filter for the flux where the states give the flux [1, 3].

Reference and their derivatives, since we want to control speed, we implement a PI controller which gives the reference for the torque T_{ref} and compensates variations of the load torque T_L. The gain k_p is put on the speed measure to limit the overshoot in the speed response [1-7] (see Figure 4). For the safety of the inverter, it is advisable to limit the stator current I_s of the induction motor. That keeping the torque constant while having a constant rotor flux norm results in a constant stator current. Thus, due to the decoupling of the nonlinear controller, to limit the stator current, the torque reference is limited while keeping the flux reference constant [1-8]. Torque and flux will follow the references due to the tracking capabilities of the decoupling controller.

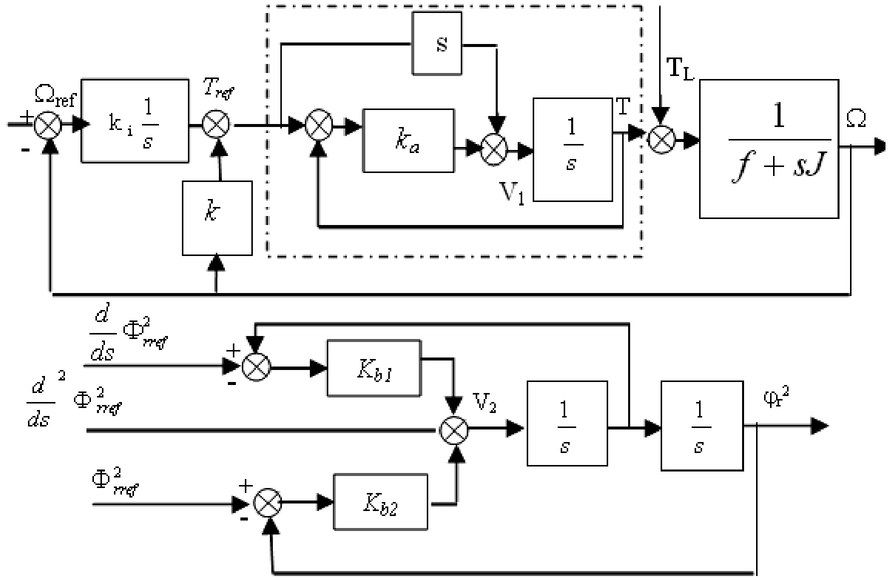


Figure 4
Speed and rotor flux-controller of input output linearization

4 Flux Observer Kalman Filter

Kalman Filter is a recursive mean squared estimator. It is capable of producing optimal estimates of system states that are not measured. The elements of the covariance matrices *Q* and *R* serve as design parameters for convergence of the system. The Kalman Filter approach assumes that the deterministic model of the motor is disturbed by centred white noise viz. the state noise and measurement noise [9]. Given the discrete Linear Time Varying (LTV) model depending on the measured speed [10].

$$S = \begin{cases} x_{k+1} = A_k x_k + B_k u_k + w_k \\ y_k = C x_k + v_k \end{cases} \tag{38}$$

With A, B, C expressions given in appendix

$$x_k = [\varphi_{s\alpha} \quad \varphi_{s\beta} \quad \varphi_{r\alpha} \quad \varphi_{r\beta}]^T;\ u_k = [u_{s\alpha} \quad u_{s\beta}]^T;\ y_k = [i_{s\alpha} \quad i_{s\beta}]^T \tag{39}$$

w_k and v, are mutually independent noises with covariance

$$E(w_k w_k^T) = Q_k\,, \quad E(v_k v_k^T) = R_k\,. \tag{40}$$

The Kalman (one step ahead) predictor minimises in a least square sense the covariance:

$$P_{k/k-1} = E\left\{(x_k - \hat{x}_{k/k-1})^T (x_k - \hat{x}_{k/k-1})\right\} \tag{41}$$

And produces the optimal estimate

$$\begin{aligned} \hat{x}_{k+1/k} &= A_k \hat{x}_k + B_k u_k + K_k (y_k - C\hat{x}_k) \\ K_k &= A_k P_{k/k-1} C^T (CP_{k/k-1} C^T + R_k)^{-1} \\ P_{k/k-1} &= A_k P_{k/k-1} A_k^T - A_k P_{k/k-1} C^T \\ &\quad (CP_{k/k-1} C^T + R_k)^{-1} A_k P_{k/k-1} A_k^T + Q_k \end{aligned} \tag{42}$$

5 Simulation Results

To validate the performances of the proposed controller, we provide a series of simulations and a comparative study between the performances of the proposed control strategy and those of the classical Field Oriented Control.

A 1.5 kW induction motor with controller is simulated using the nonlinear controller and the following motor parameters:

Vn =230V,Lr =0.225H , Ls =0.225H,Lm =0.214H,

Rs =2.89Ω , Rr =2.39Ω , P =2 ,

The controller and Kalman filter are implemented using a sampling period of 0.1 ms. In order to show the robustness of the controller with respect to variation of rotor resistance and load structure, we then have the simulation cases with desired speed controls as follows:

- The speed command is designed, but the rotor resistance is increasing 50% during the time interval. Figure 5 shows the effects of the proposed control scheme. Obviously, the input stator voltage is increased to suppress the variation of the rotor resistance.
- The speed command is designed, but both Rr and TL, are changed. Both Rr and TL are increasing 50% during the time interval. As shown in Figure 6.

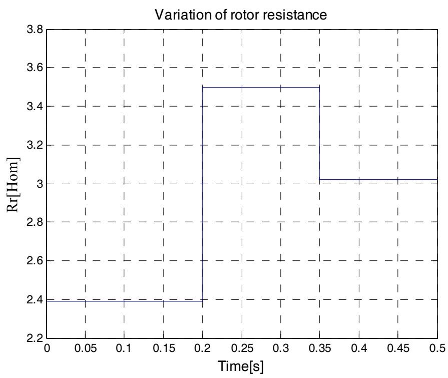


Figure 5
Variation of rotor resistance.

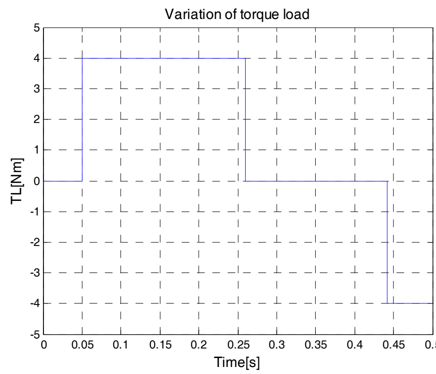


Figure 6
Variation of torque load

The results of simulation of proposed nonlinear controller NL_FOC and shift classical Field Oriented Control FOC of induction motor is shown in the figures (7 to 16) respectively. In the first case where one considers only one variation of torque load TL shown in the figures (7a, 8a, 9a and 11a), results of simulation the time histories of speed, flux magnitude tracking and current behavior are reported on the figures (7 to 10), for of the Field oriented Control (FOC) and nonlinear controller (NLC) on the figures (7b, 8b, 9b and 11b). As the figures show, it is observed that the speed tracks the reference values adequately well, for two methods. That is, with load torque perturbation. Figure 7a, show the flux presented the ripple around that for reference in FOC method, but the response of the flux are very good in NLC, we can observe an optimal reduction of the electromagnetic torque distortion, as shown in Figure 11b. It also appears that the rotor speed fits to the speed reference trajectory. The applied load torque has no effect on the flux and its effect on the speed is rapidly compensated. In the second test of simulation we make the resistances variations (R_r and TL).

The simulation results on the figures (12-16) shows that a good tracking performance is achieved and the above results demonstrate that the proposed controller has strong robustness properties in the presence of load disturbance and parameter variations. Consequently, the use of the proposed feedback nonlinear control scheme can solve the control problem of induction machines in the presence of uncertainties in load torque and resistance parameters variations without rotor resistance estimation.

5.1 Variations of Load Torque

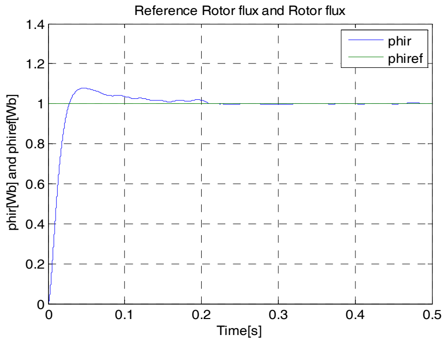


Figure 7a

Magnitude of rotor flux in FOC

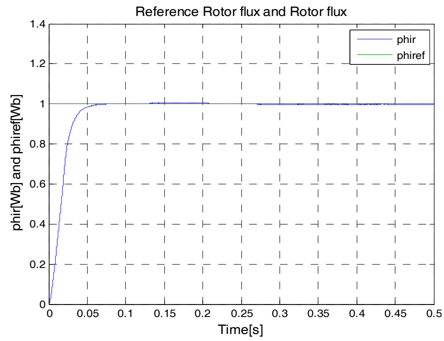


Figure 7b

Magnitude of rotor flux in NLC

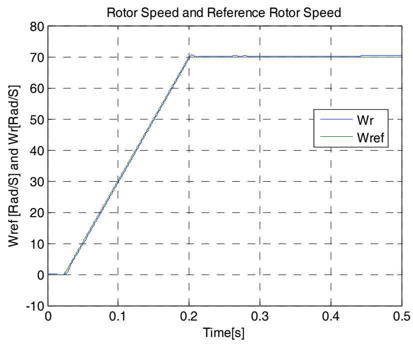


Figure 8a

Rotor speed in FOC

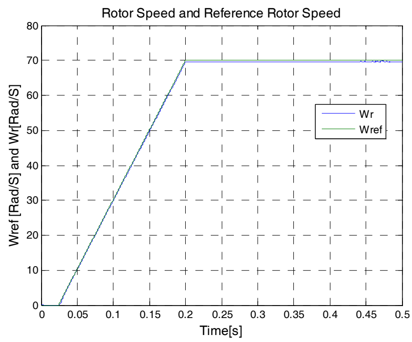


Figure 8b

Rotor speed in NLC

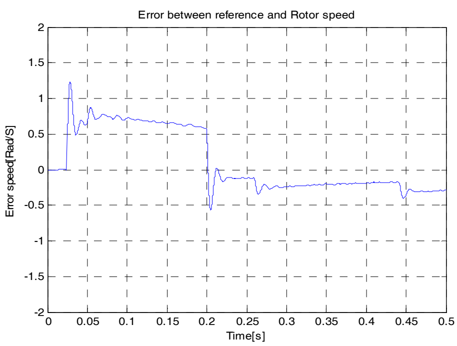


Figure 9a

Error between reference and Rotor speed in FOC

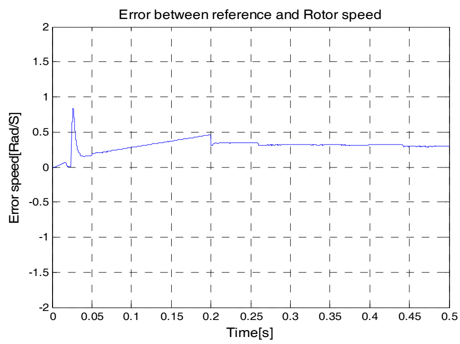


Figure 9b

Error between reference and Rotor speed in NLC

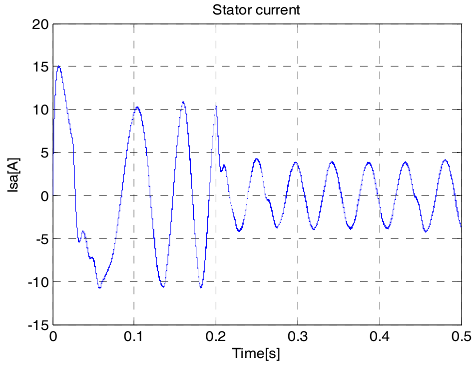


Fig. 10a: Stator current in FOC

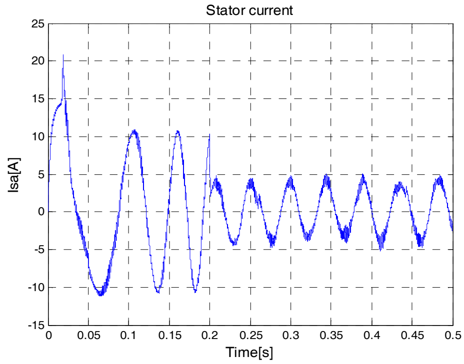


Fig. 10b: Stator current in NLC

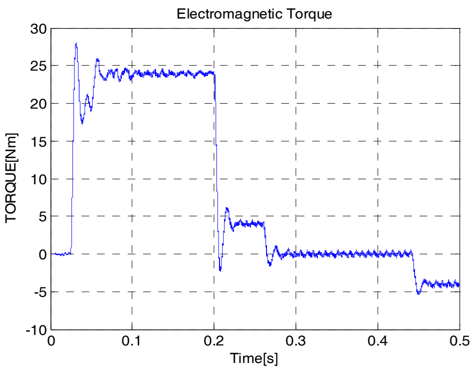


Figure 11a
Electromagnetic torque in FOC

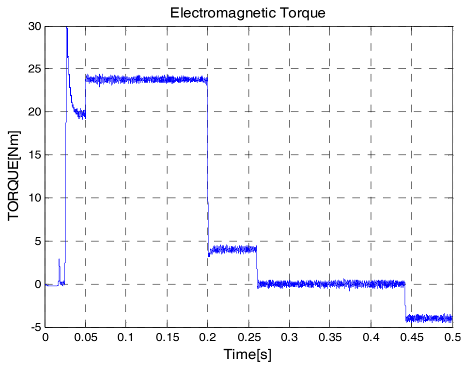


Figure11b
Electromagnetic torque in NLC

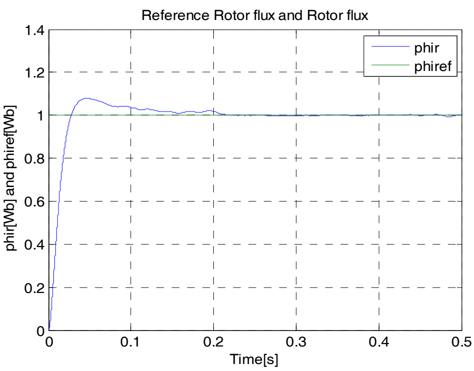


Figure 12a
Magnitude of rotor flux in FOC

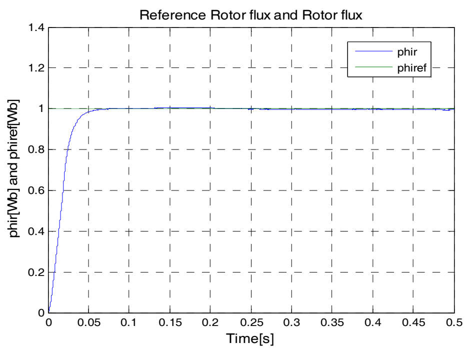


Figure 12b
Magnitude of rotor flux in nlC

5.2 Variations of Rotor Resistance and Torque Load

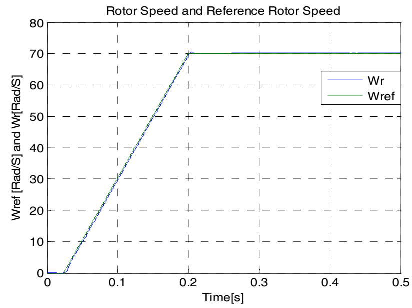


Fig. 13a: Rotor speed in FOC

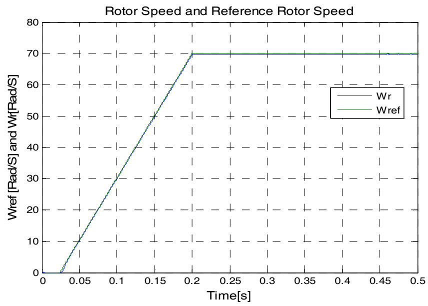


Fig. 13b: Rotor speed in NLC

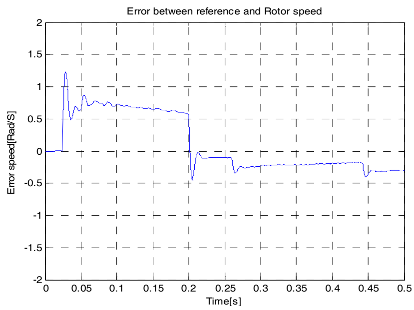


Fig. 14a: Error between reference and Rotor

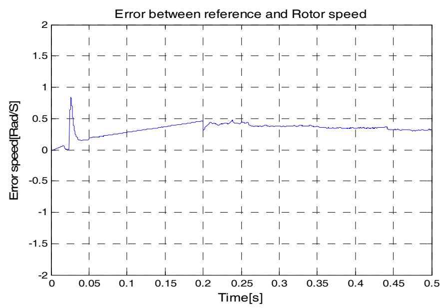


Fig. 14b: Error between reference and Rotor

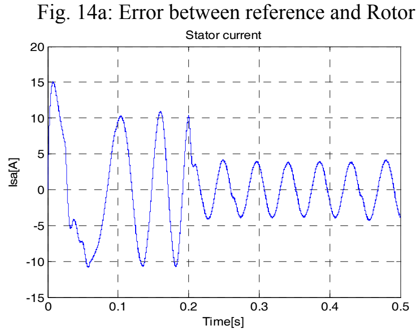


Fig. 15a: Stator current in FOC

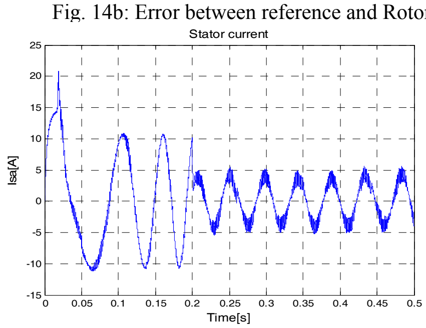


Fig. 15b: Stator current in NLC

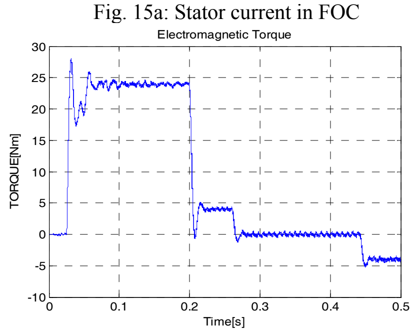
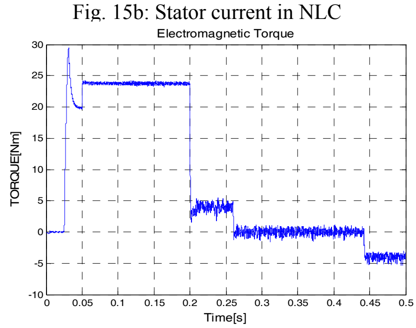


Fig. 16a: Electromagnetic torque in FOC



Conclusion

In this paper, two control techniques have been compared for induction motors': classical Field Oriented control, and input-output linearization control, proposed by the current authors. From the comparative study, one can conclude that the two methods demonstrate nearly the same dynamic behaviour. However, the input-output linearization controller shows better performance than the Field Oriented controller in speed tracking at high speed ranges. The numerical simulations validate the performances of the proposed method and even in the unknown parameter case and achieve better speed and rotor flux tracking.

References

[1] Marino, R., Peresada, S., Valigi, P., Adaptive Input-Output Linearizing Control of Induction Motors, *IEEE Automat. Contr., Vol. 38, No. 2*, Feb. 1993, pp. 208-221

[2] A. Merabet, M Ouhrouche, R. T. Bui, H. Ezzaidi, Nonlinear PID Predictive Control of Induction Motor Drives, *IFAC Workshop on NMPC for Fast Systems. October 9-11, 2006, Grenoble, France*

[3] Raumer, T., von., J., Dion, M., and Dugard, L., Adaptive Nonlinear Speed and Torque Control of Induction Motors, in Proc. *European Contr. Conf., Groningen, Netherlands,* 1993

[4] Fekih, A., Fahmida, N., Chowdhury, On Nonlinear Control of Induction Motors: Comparison of two Approaches, *Proceeding of the 2004 American Control Conference Boston, Massachusetts*, June 30-July 2, 2004

[5] Hou, Lee., Ban, Li., Chen, Fu., Nonlinear Control of Induction Motor with Unknown Rotor Resistance and Load Adaptation, *Proceedings of the American Control Conference Arlington,* VA 2001, June 25-27

[6] Thomas Von Raumer., J., Dion, M., and Dugard, Luc, Thomas, Jean Luc.: Applied Nonlinear Control of an Induction Motor Using Digital Signal Processing, *IEEE Transactions On Control Systems Technology,* December 1994,Vol. 2, No. 4

[7] Fekih, A., Ennaceur, Benhadj Braiek, Design of a Multimodel Approach for Nonlinear Control of Induction Motor Drives, *Proceedings of the IEEE SMC* 2002, TP 1I7

[8] Thomas Von Raumer., Dion, Jean Michel. and Dugard, Luc., Adaptive Nonlinear Control Of Induction Motors With Flux Observer, *The authors are also with the GDR CNRS Autometique*

[9] Ramdane, H., Redouane, T., Boucher, P., Dumur, D. , End Point Constraints Nonlinear Predictive Control with Integral Action for Induction Motor, *Proceedings of the 2004 IEEE International Conference on Control Applications Taipei*, Taiwan, September 2-4, 2004

[10] Thongam, J. S., Ouhrouche, M., Beguenane, R., Sensorless Indirect Vector Control of Induction Motor Drive Robust Against Rotor Resistance Variation, *Proceeding of Power and Energy Systems, 2004, p. 448*

[11] Maaziz, M. K., Boucher, P., Dumur, D., Flux and Speed Tracking of an Induction Motor Based on Nonlinear Predictive Control and Feedback Linearization, *Proceedings of the American Control Conference San Diego, California, June 1999*

Video Traffic Prediction Using Neural Networks

Miloš Oravec, Miroslav Petráš, Filip Pilka

Department of Telecommunications, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology
Ilkovičova 3, 812 19 Bratislava, Slovak Republic
milos.oravec@stuba.sk, mirops@gmail.com, pilka_filip@nextra.sk

Abstract: In this paper, we consider video stream prediction for application in services like video-on-demand, videoconferencing, video broadcasting, etc. The aim is to predict the video stream for an efficient bandwidth allocation of the video signal. Efficient prediction of traffic generated by multimedia sources is an important part of traffic and congestion control procedures at the network edges. As a tool for the prediction, we use neural networks – multilayer perceptron (MLP), radial basis function networks (RBF networks) and backpropagation through time (BPTT) neural networks. At first, we briefly introduce theoretical background of neural networks, the prediction methods and the difference between them. We propose also video time-series processing using moving averages. Simulation results for each type of neural network together with final comparisons are presented. For comparison purposes, also conventional (non-neural) prediction is included. The purpose of our work is to construct suitable neural networks for variable bit rate video prediction and evaluate them. We use video traces from [1].

Keywords: data prediction, video traffic, neural network, multilayer perceptron, radial basis function network, backpropagation through time

1 Prediction of Video Traffic

The role of dynamic allocation of bandwidth is to allocate resources for variable-bit rate (VBR) video streams while capturing the bursty character of video traffic. By using prediction schemes, it is possible to increase utilization of network resources and to fulfill QoS (quality of service) requirements [2].

Multimedia traffic, especially MPEG video traffic became dominant component of network traffic. It is due to excessive use of services like video-on-demand (VoD), videoconferencing, broadcast and streaming video. Periodic correlation structure, complex bit rate distribution and noisy streams are some characteristics of MPEG video traffic [3]. Some traffic and congestion control procedures must be used, among which connection admission control (CAC), usage parameter control (UPC), traffic shaping, congestion indication, priority control, packet discarding are examples of the most important procedures.

The design of a bandwidth allocation scheme for VBR video is very difficult [4]. This is due to bursty character of such traffic, while at the same time VBR video requires strongest QoS characteristics such as delay, loss and jitter.

Neural networks are generally considered to be one of the most effective tools for prediction. Due to their analogy with biological neural networks (human brain), they seems to be suitable to solve prediction related tasks. For prediction, herein we use feedforward networks – multilayer perceptron (MLP) and radial basis functions (RBF) network, and recurrent backpropagation-through-time (BPTT) network.

2 Neural Networks

2.1 Neuron, Neural Network and Learning

The origin of artificial neural networks (ANN) [5] was inspired by the biological nervous system. The main inspiration was the human brain and the way it processes information. The human brain consists of very large number of elements (neurons), which are massively interconnected.

The basic element of each ANN is the neuron. The basic scheme of the neuron is shown in Fig. 1. Connecting such elements in various ways leads to different architectures of neural networks.

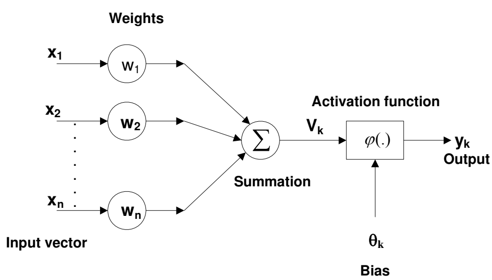


Figure 1
Basic model of neuron

The ANN learns by example. Learning (in the terminology of neural networks) is the process by which the weights are adapted. This process is represented by a learning algorithm. Many learning algorithms exist [5], [6], they differ in the way they adjust the weights of particular neurons. In this paper, we use the supervised learning paradigm [5], [6].

2.2 Multilayer Perceptron

The multilayer perceptron (MLP) [5], [6] is probably the best-known and frequently used neural network. It consists of one input and one output layer and it can contain one or more hidden layers of neurons. For this type of networks, the sigmoidal activation functions (including most popular logistic function) are mainly used.

MLP is trained by the backpropagation algorithm [5], [6]. The error signal of neuron *j* is defined as the difference between its desired and actual output:

$$e_j(n) = d_j(n) - y_j(n) \tag{1}$$

From error signal, the local gradient needed for the backpropagation algorithm can be computed:

$$\delta_j(n) = e_j(n)\varphi_j'(v_j(n)), \tag{2}$$

where $v_j(n)$ is the inner activity of the neuron.

Then the weight adaptation is done as follows:

$$\Delta w_{ji}(n) = \eta\delta_j(n)y_i(n), \tag{3}$$

where $\Delta w_{ji}(n)$ is the weight adjustment in time n, η is the learning rate, $\delta_j(n)$ is the local gradient and $y_i(n)$ is the input signal of neuron *j*.

In general, we can write

$$w_{ji}(n+1) = w_{ji}(n) + \Delta w_{ji}(n) \tag{4}$$

If the neuron is the output neuron of the network, then we can use the presented algorithm to compute the weights adaptation. But if it is a hidden neuron, its desired output is not known and this is why the error signal for the hidden neuron has to be calculated recursively from the error signals of the neurons directly connected to the neuron (from output layer).

2.3 RBF Networks

When we look at the design of the neural networks from the perspective of approximation in multidimensional space, then learning is equivalent to finding such a plane in the multidimensional space which best approximates the training data. Neurons in the hidden layer represent a set of functions which represents the basis functions for the transformation of input vectors to the space of hidden neurons. These functions are called the radial basis functions (RBF) [7], [8], [5], [6].

The interpolation scheme using RBFs can be represented as follows

$$f(\mathbf{x}) = \sum_{j=1}^{N} w_j \phi\left(\left\|\mathbf{x} - \mathbf{c}_j\right\|\right) \tag{5}$$

Function $f(x)$ is the interpolation function which uses N radial basis functions ϕ_i, where $\phi_i : R^p \to R, i = 1,\ldots,N$ and $\phi_i = \phi(\|\mathbf{x} - \mathbf{c}_i\|)$, $\phi : R^+ \to R$, $\mathbf{x} \in R^p$, $\|\cdot\|$ is norm on R^p (often Euclidian), $\mathbf{c}_i \in R^p$ are centers of RBFs, $\mathbf{w}_j$ are coefficients for linear combination of RBFs which we want to find.

Since $f(\mathbf{x}_i) = d_i, \forall i = 1,\ldots,N$, we get the equation (5) in matrix representation

$$\begin{bmatrix} \phi_{11} & \cdots & \phi_{1N} \\ \vdots & \ddots & \vdots \\ \phi_{N1} & \cdots & \phi_{NN} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ \vdots \\ d_N \end{bmatrix}, \tag{6}$$

where the elements of the matrix are

$$\phi_{ij} = \phi\left(\left\|\mathbf{x}_i - \mathbf{c}_j\right\|\right), \; i, j = 1,\ldots,N \,. \tag{7}$$

When ϕ is a regular matrix we can find one exact solution. Many functions guarantee the regularity of the matrix. The most common is the Gaussian function

$$\phi(x) = \exp\left(-\frac{x^2}{2\sigma^2}\right), \tag{8}$$

where σ is the width of the Gaussian RBF.

Presented scheme can be easily extended to mapping $F : R^p \to R^q$, F and $\boldsymbol{d}$ are then in the form $(f_1,\cdots,f_q)$, $(d_1,\ldots,d_q)$, respectively.

The fact, that the number of RBFs is the same as the number of data points that are to be interpolated is a main disadvantage of interpolation scheme. Typically, there is a smaller number of RBFs compared to the number of given data points. Then we speak about approximation scheme, the matrix ϕ is no more square and its inverse matrix does not exist. Solution can be found by least-squares optimalization method, by pseudoinverse matrix or by RBF neural network with hidden neurons representing radial basis functions.

The training process of RBF network then consists of three steps. The first step is to find the centers of the basis functions. The second step adjusts additional parameters of RBFs (if any). The third step serves for output weights computation. More information about the training can be found in [7] and [8].

2.4 BPTT Networks

The backpropagation through time (BPTT) neural network [5], [9], [10], [11] belongs to a class of recurrent neural networks. Such networks contain feedback connections. Their training algorithms often compute the gradient of an error measure in weight space. BPTT learning algorithm is based on unfolding the temporal operation of the network into a multilayer feedforward network, where one layer is added at every time step. In this manner, the network is converted from a feedback system to purely feedforward system. The gradients of weights for a recurrent network are approximated using a feedforward network containing a fixed number of layers. More details about the BPTT networks and training algorithms can be found in [9], [10], [11].

2.5 Optimal Linear Prediction

One of the methods to predict the future samples of a time series is the autocorrelation method of autoregressive (AR) modeling. The idea of this method is to find the best coeficients of a prediction filter [12].

If we assume as in [13] that the video transmition rate sequence for linear prediction is $\{x(n)\}$, then the estimated (predicted) series can be expessed as follows:

$$\hat{x}(n+1) = -\sum_{i=1}^{m} a(i).x(n+1-i) \qquad (9)$$

Coeficients $a(i)$ are the coeficients of the prediction filter and m is the order of AR model. The coeficients $a(i)$ are computed by the following equations:

$$\boldsymbol{X}.\boldsymbol{a} = \boldsymbol{b} \qquad (10)$$

where

$$\boldsymbol{X} = \begin{bmatrix} x(1) & 0 & \dots & 0 \\ x(2) & x(1) & \ddots & 0 \\ \vdots & x(2) & \ddots & x(1) \\ x(m) & \vdots & \ddots & x(2) \\ \vdots & x(m) & \ddots & \vdots \\ 0 & \dots & 0 & x(m) \end{bmatrix}, \boldsymbol{a} = \begin{bmatrix} 1 \\ a(2) \\ \vdots \\ a(p+1) \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (11)$$

and m is the length of input vector $\boldsymbol{x}$. Through the least squares problem using the equation

$$\boldsymbol{X}^H.\boldsymbol{X}.\boldsymbol{a} = \boldsymbol{X}^H.\boldsymbol{b} \qquad (12)$$

we come to the Yule-Walker equations [12]

$$\begin{bmatrix} r(1) & r(2)^* & \cdots & r(p)^* \\ r(2) & r(1) & \ddots & \vdots \\ \vdots & \ddots & \ddots & r(2)^* \\ r(p) & \cdots & r(2) & r(1) \end{bmatrix} \cdot \begin{bmatrix} a(2) \\ a(3) \\ \vdots \\ a(p+1) \end{bmatrix} = \begin{bmatrix} -r(2) \\ -r(3) \\ \vdots \\ -r(p+1) \end{bmatrix}, \qquad (13)$$

where the factor r is an autocorrelative estimation of the input vector x. This equation can be solved using the Levinson's recursion [12].

3 Simulation Results

3.1 Used Data Set and Simulation Tools

The data used for the training and test sets is taken from the video stream files of Telecommunication Networks Group, Technical University of Berlin, Germany [1]. We used trace file from MPEG-4 Jurassic Park I movie in high quality. Both the training and the test set consist of 2000 patterns (first 2000 representing the training, next 2000 the test set) and they are shown in Fig. 2. Traces from [1] were also used etc. in [2], [3].

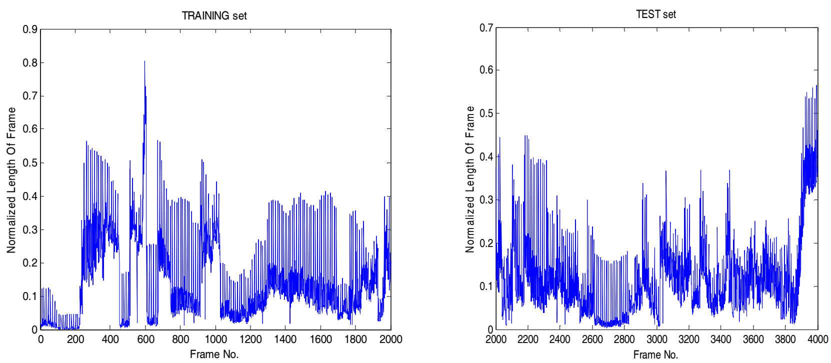


Figure 2

Training (left) and test (right) sets – originals

Our predictions were based on taking N previous patterns to predict one following pattern. Since we used a supervised learning paradigm, our networks worked with desired values of target patterns during the training process.

All our simulations were done using Stuttgart Neural Network Simulator [14] and Matlab environment [15].

3.2 Prediction by MLP

In order to achieve good results, probably one of the most important problems is to choose the appropriate configuration of neural network. During training of MLPs, we tried many types of configurations for our predictions. There were notable differences of prediction errors among them. We chose the MSEnorm (normalized mean square error) as an objective criterion to compare them.

We made experiments with the number of input neurons changing from 1 to 7 and also experiments with various number of hidden neurons and number of hidden layers. We achieved the best results of training the MLP network using network configuration 3-10-1 (which means: 3 input neurons, 10 neurons in hidden layer, 1 output neuron), Levenberg-Marquardt training algorithm and learning-rate parameter 0.1. The results of the prediction for the training and test set are shown in Fig. 3. The detail of the test set can be seen in Fig. 4.

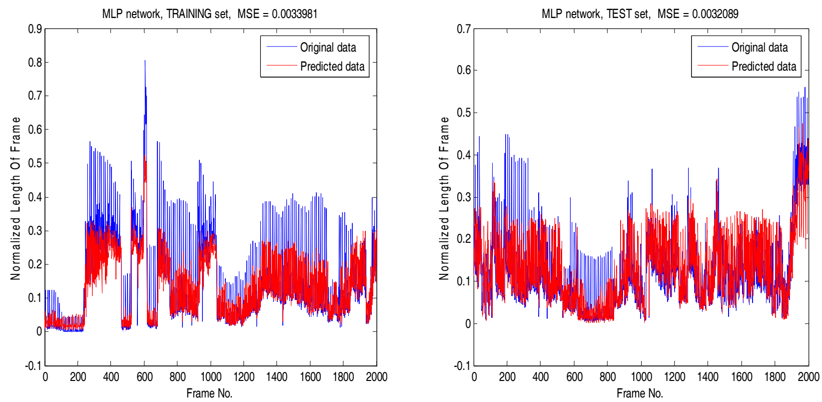


Figure 3

Prediction results for training (left) and test (right) set by MLP network 3-10-1

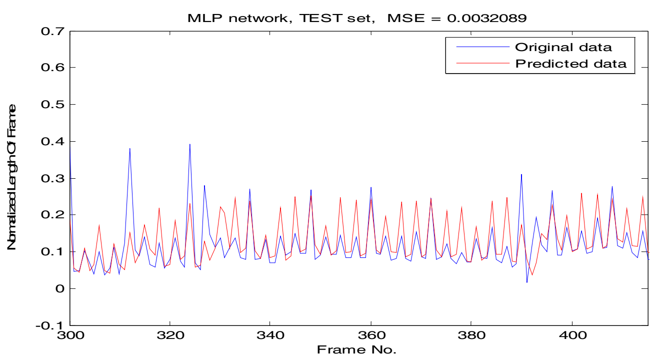


Figure 4

Prediction results for test set by MLP network 3-10-1– detail

In order to make the prediction more effective, it is possible to take also the character of the time series into account. For our data, approximately each 12th pattern forms a peak (in other words, the distance of the consecutive peaks is mostly 12 patterns). This is why we chose also 12 input patterns (besides our examined 3 inputs) for the prediction. The result for test set can be seen in Fig. 5 for MLP with configuration 12-20-10-1 (again, this is the most appropriate configuration after doing examination of various MLP configurations).

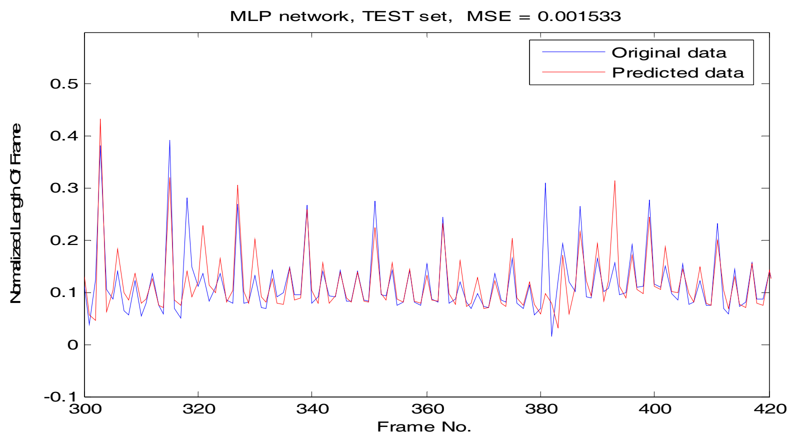


Figure 5
Prediction results for test set by MLP network 12-20-10-1– detail

3.3 Prediction by RBF Networks

In this section, we present the results of the RBF neural networks. We have tried different network configurations. Similar to searching optimal network configuration for MLPs, we tried various number of input and hidden neurons (unlike MLP, RBF network contains exactly one hidden layer). Fundamental question when using RBF networks is the number of hidden neurons to be used. The number of hidden neurons for our experiments altered from 10 to 2000. We have found out that the best approach for our data is to use 500 hidden neurons. We present best results achieved by network configuration 3-500-1. The results for the training and the test sets for this network are shown in Fig. 6. Fig. 7 contains detail for the test set.

Similar to MLP, we examined also RBF networks with 12 inputs (due to peak nature of video stream). The result is shown in Fig. 8.

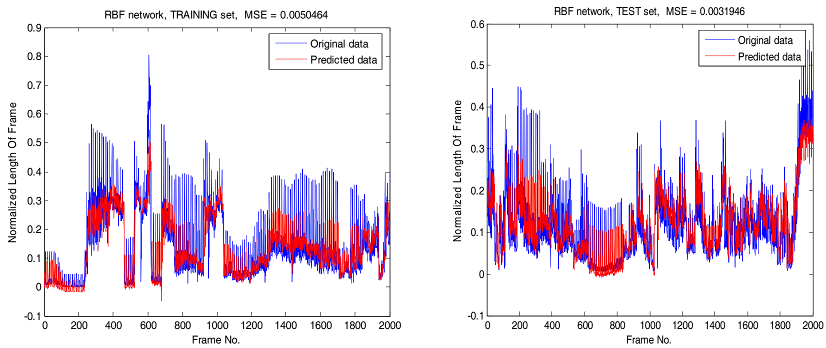


Figure 6

Prediction results for training (left) and test (right) set for RBF network 3-500-1

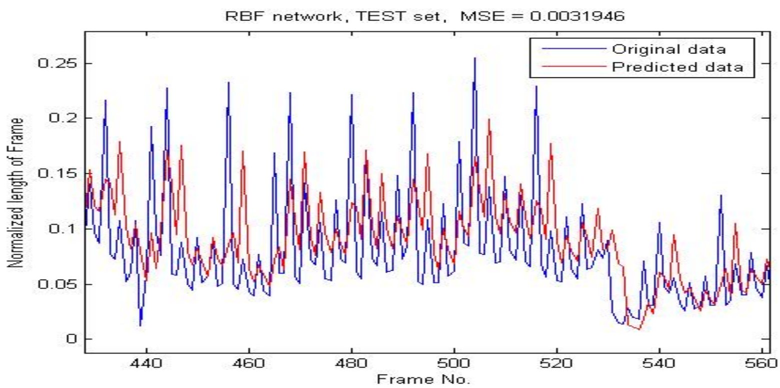


Figure 7

Prediction results for test set by RBF network 3-500-1 – detail

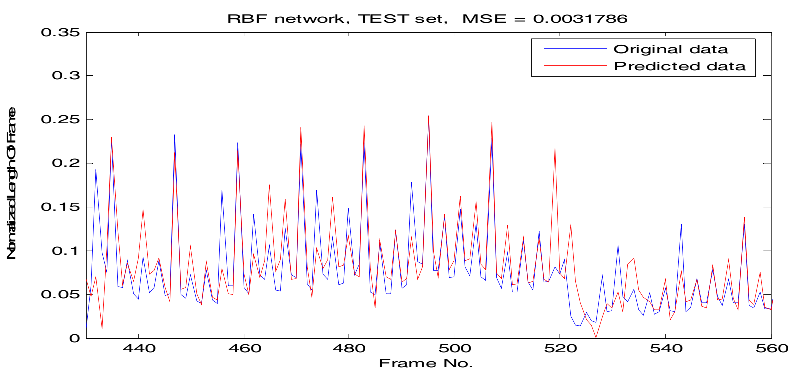


Figure 8

Prediction results for test set by RBF network 12-500-1 – detail

The comparison between RBF network of the configuration 3-500-1 and 3-2000-1 network is shown in Fig. 9. By comparing the values of the MSE for both networks, one can see that 3-500-1 RBF network behaves better.

As we mentioned earlier, our training and test sets both have 2000 elements. It means that if we create a RBF network with configuration 3-2000-1 then each hidden neuron represents one element of the training set. This approach is known as an interpolation scheme. The approximation scheme means that we use less then 2000 neurons in the hidden layer.

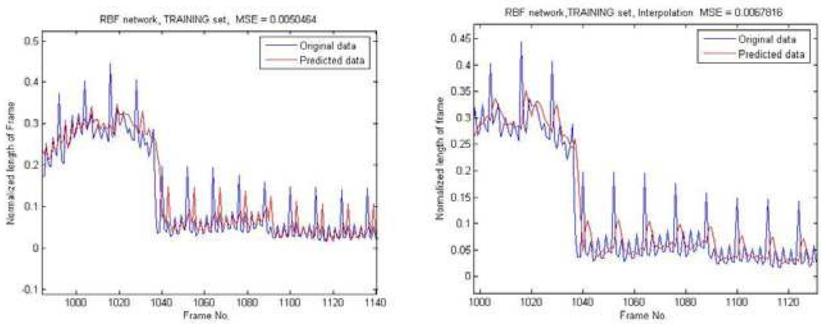


Figure 9

The comparison of two RBF networks (left: 3-500-1, right 3-2000-1) – details

In both graphs of Fig. 9 the details are shown, so we can see not only the MSE of the two networks, but it is also possible to compare the subjective performance.

3.4 Prediction by BPTT Network

For the BPTT networks, we used similar approach as for other neural network models. At first, we tried different configurations, until we found one with the best values of MSE. Starting with five hidden neurons only, the results were unsatisfactory, so we chose other configurations. The configuration containing 20 hidden neurons gave best results.

The configuration used by the BPTT is 3-20-1. The results are shown in Fig. 10 with detail included in Fig. 11. Comparing the values of MSE, we can see that the results for the BPTT networks are comparable with those of other neural networks.

Fig. 12 shows detail of the prediction using BPTT network with 12 inputs (motivation for using 12 inputs is the same as for MLP and RBF networks discussed above).

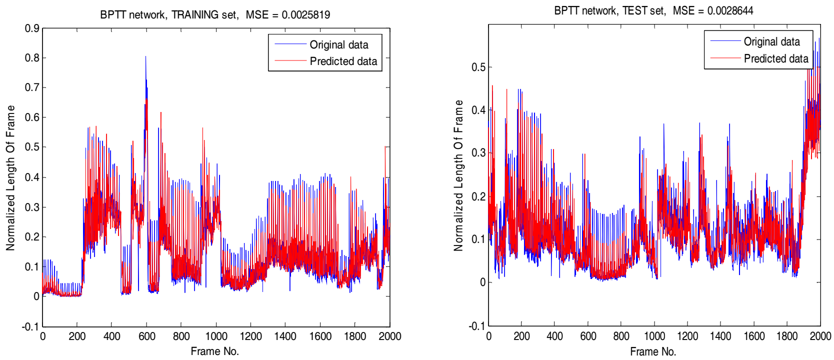


Figure 10

Prediction results for training (left) and test (right) set for BPTT network 3-20-1

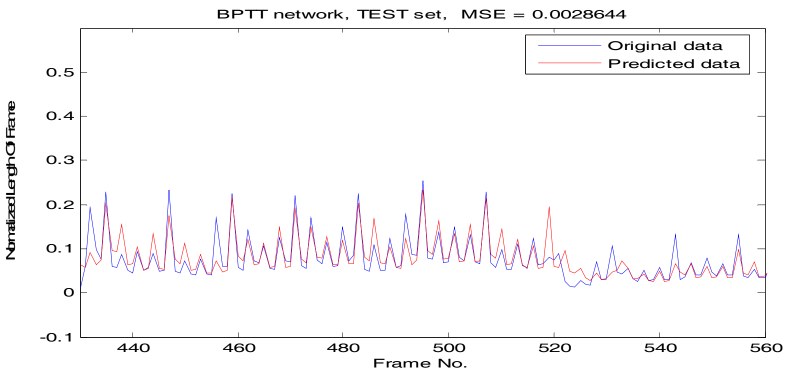


Figure 11

Prediction results for test set by BPTT network 3-20-1 – detail

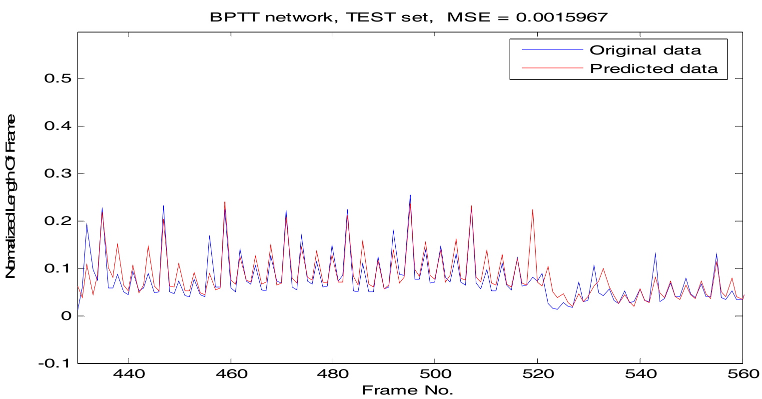


Figure 12

Prediction results for test set by BPTT network 12-20-1 – detail

3.5 Predictions Using Moving Average

Since presented trace files contain fast changes (peaks), one can expect better results when processing them not directly but using some technique to make the input signal smoother (simpler). Therefore we propose simple method of using moving averages as the input to neural network to eliminate the fast changing of signal. The principle of the moving averages (MA) is shown in Fig. 13.

We take *N* patterns of the original signal and compute the moving averages (summation of patterns divided by *N*, we move forward over one pattern) and we predict next pattern from *M* values of MA. Since at the time when we are predicting the next MA we know the previous patterns of the original signal too, we can obtain the next pattern of the original signal easily. We just need to multiply the predicted MA by *N* and subtract the last N-1 known patterns of the original signal (see Fig. 13).

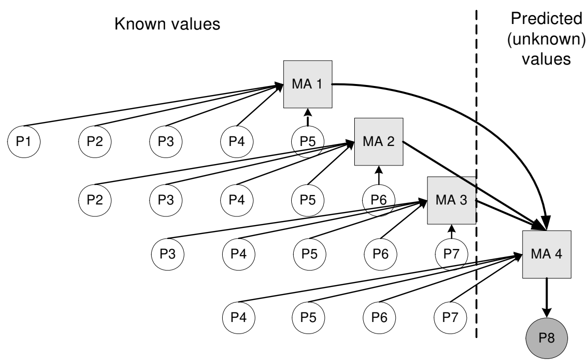


Figure 13
The principle of the moving average prediction

We can see the method of prediction of 8^{th} pattern (P8) of the original signal in Fig. 13. Each MA is calculated from five corresponding patterns (e.g. MA1 = 1/5*Σ P(1-5)) and the next MA is predicted from three previous MA. It means, when we already know MA1 to MA3, we predict MA4 from them. Because the original patterns P4-P7 are known at that time, we just need to use inverse sequence of steps – we multiply MA4 by five and subtract the sum of P4-P7. The result is the next pattern of the original signal, i.e. P8.

The main advantage of this prediction method is that the input to the neural network is not the original data, but much less dynamic data (especially if we compute the moving averages for such number of patterns that each MA contains one peak only) – then MA input is evidently smoother than the original signal. One of the disadvantages is that small error of MA prediction can cause large error of the original signal during its reverse calculation from MA.

Moving averages from 12 patterns of training and test sets are shown in Fig. 14.

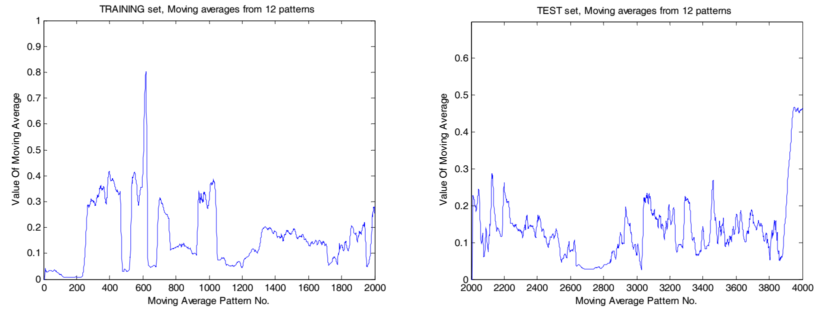


Figure 14

Moving averages of 12 patterns for training (left) and test (right) set

Prediction results using moving average for MLP network with configuration 3-10-1 are shown in Fig. 15 (with detail of the test set shown in Fig. 16).

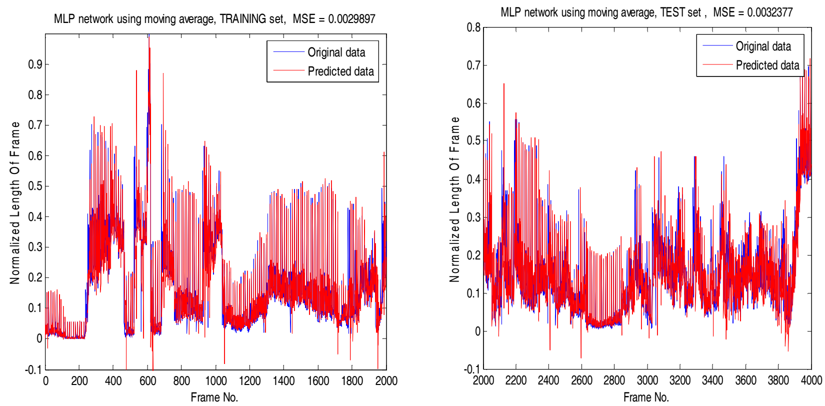


Figure 15

Prediction results for training (left) and test (right) set using moving averages for MLP network 3-10-1

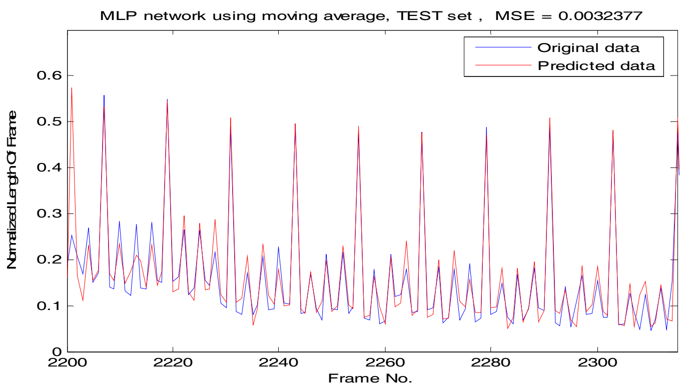


Figure 16

Prediction results for test set using moving averages for MLP network 3-10-1 – detail

We applied the concept of moving average also for MLP with 12 inputs because of periodic appearance of peaks in time series. The result is shown in Fig. 17.

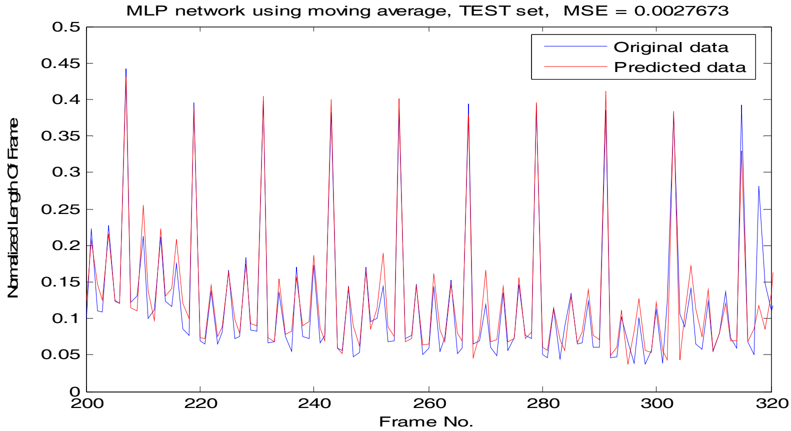


Figure 17

Prediction results for test set by using moving averages on MLP network 12-20-10-1 – detail

The results for RBF network using the concept of moving average are shown in Fig. 18. The presented network configuration is 3-500-1. Fig. 19 contains detail of moving average simulation for RBF network with 12 inputs.

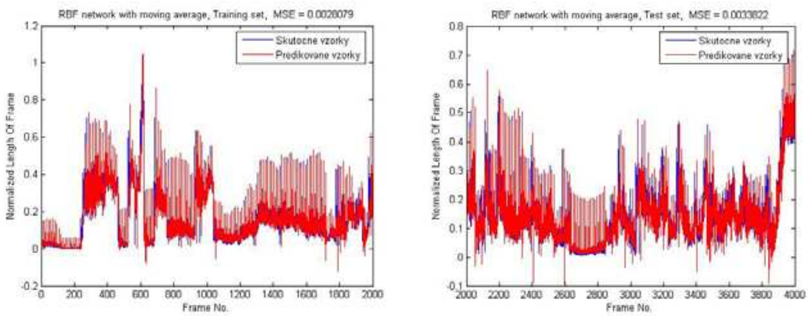


Figure 18

Prediction results for training (left) and test (right) set for RBF network 3-500-1

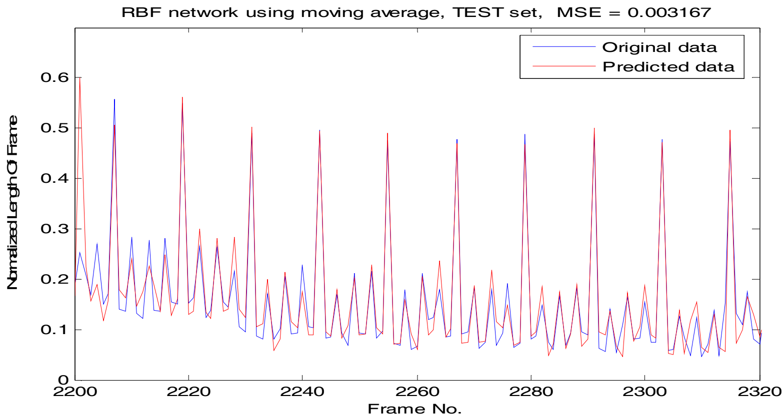


Figure 19

Prediction results for test set by using moving averages on RBF network 12-500-1 – detail

3.6 Comparison

Tab. 1 and Fig. 20 show best prediction results obtained by networks with three inputs. We got the best results using BPTT network of configuration 3-20-1 for both sets of data (normalized MSE for the training set and test set is 0.0025819 and 0.0028644, respectively). Although the results for other types of networks are little bit worse, the differences are visually not so significant. Tab. 1 also includes results from the prediction using LP (linear predictor) with 12 and 75 coefficients. In Fig. 20, the results for LP with 75 coefficients are shown.

Table 1

The comparison of MSE for the networks with 3 inputs and comparison with LP

Network	Training set	Test set
MLP 3-10-1	0,0033981	0,0032089
RBF 3-500-1	0,0050464	0,0031946
BPTT 3-20-1	0,0025819	0,0028644
MLP-MA 3-10-1	0,0029892	0,0032377
RBF-MA 3-500-1	0,0028079	0,0033822
LP - 12 coeff.	0,0159024	0,0085851
LP - 75 coeff.	0,0065945	0,0057113

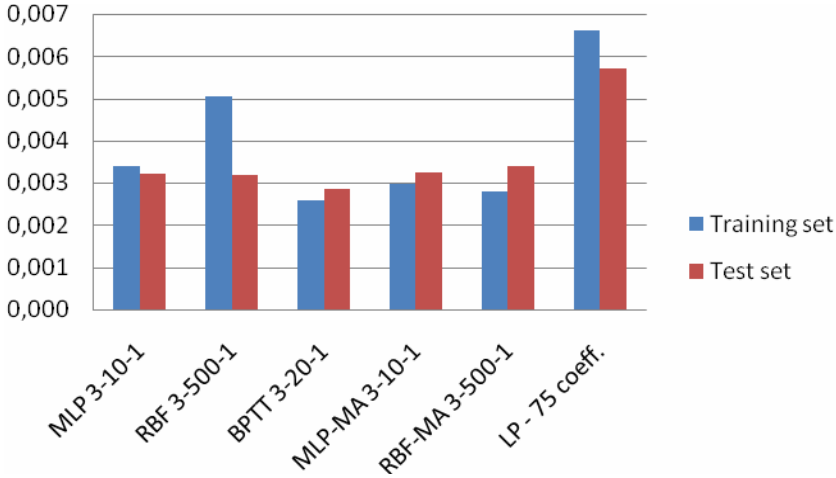


Figure 20

The comparison of results of test set for all types of networks using 3 inputs and comparison with LP

As we already discussed, our examined streams contain peaks, approximately at each 12th position. This is why we simulated also neural networks with 12 input neurons. The values of normalized MSE for networks using 12 inputs are shown in Tab. 2 and in Fig. 21 (also the comparison with LP is present).

Fig. 22 presents summary of best results obtained using neural networks with 3 and 12 input patterns and the comparison with the linear predictor using 75 prediction coefficients. It can be seen that better results can be obtained using 12 inputs. Since there was exactly one peak in each set of patterns fed to the network, it has learned the positions of the peaks better. Of course, the disadvantage of this way of prediction is that it strongly depends on used video stream – and different streams can have different distance between the peaks (eventually, the stream does not need to contain peaks in such periodic manner).

Table 2

The comparison of MSE of the networks with 12 inputs and LP

Network	Training set	Test set
MLP 12-20-10-1	0,0012937	0,0015330
RBF 12-500-1	0,0005020	0,0031786
BPTT 12-20-1	0,0010890	0,0015967
MLP-MA 12-20-10-1	0,0025311	0,0027673
RBF-MA 12-500-1	0,0028079	0,0033822
LP - 12 coeff.	0,0159024	0,0085851
LP - 75 coeff.	0,0065945	0,0057113

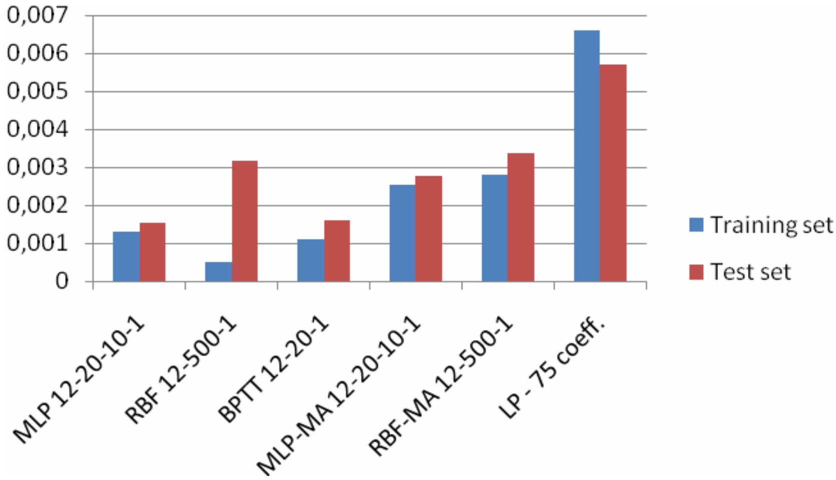


Figure 21

The comparison of results of test set for all types of networks using 12 inputs and comparison with LP

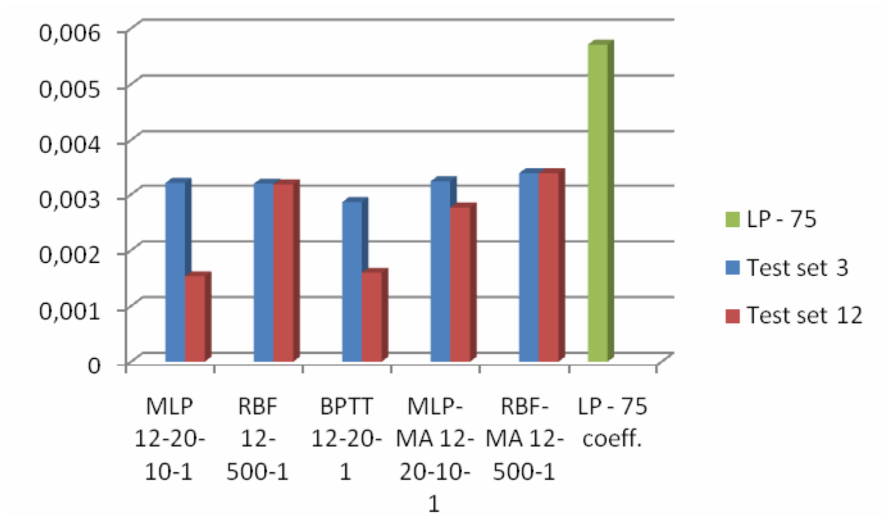


Figure 22

The comparison of results of test set of networks with 3 and 12 inputs and comparison with LP

Conclusions

Efficient data compression methods followed by efficient prediction schemes are very important in order to achieve required QoS of multimedia traffic.

The goal of this paper was to predict the video time series for an efficient bandwidth allocation of the video signal using neural networks. The application of presented methods is in traffic and congestion procedures of communication networks.

We have tried many configurations and types of neural networks for video stream data prediction. First, we tried to find suitable network configurations. This process led us to using three input patterns in each step of prediction. As we can see in Fig. 20, the best results of prediction for test set were achieved by using BPTT network with configuration 3-20-1, while the results achieved by MLP with configuration 3-10-1 and RBF with configuration 3-500-1 were a bit worse and both comparable.

Although the results for the networks using MA were very similar to the networks without MA concept, the network training was much faster (for both MLP and RBF network) which could be useful for adaptive prediction systems. Especially RBF network training using MA takes approximately 50% time duration comparing to "direct" approach to prediction.

For comparison purposes, we tried to predict the data using 12 input patterns. We chose 12 patterns in order to have just one peak in each step of prediction. Of course, the number of input patterns was found empirically and it depended on the behavior of the time series. Although better prediction results can be obtained in such manner, it is evident that the mentioned number of input patterns is not suitable for any type of time series.

In order to compare neural and conventional prediction, we presented the results of linear predictions. As can be seen from Figs. 20, 21 and 22 the results of neural networks are significantly better.

From Fig. 22, we can see that the best results of prediction for three inputs can be achieved by BPTT network. The results for other types of network were for a certain extent worse; especially the result of test set for RBF network was obviously the worst (even though the result for the training set was the best of all networks).

Besides efficient prediction schemes also efficient data (herein video data) compression methods must be used. These compression methods must take QoS requirements into account. Optimization of compressed bit flows is necessary; such optimization can be based on channel capacity (allocation of sufficient bit flow through a channel) or on receiver quality requirements. Example of such optimized coder can be found in [16].

Yet other approaches lowering requirements for channel coding are used. MPEG (as well as JPEG) standard uses block-based coding techniques. For high compression ratios, the effect of block loss or random bit error during transmission can cause a serious problem. It is possible to use error concealment algorithms [17] in block-based image coding systems, where the information of pixels surrounding a damaged block is used to reconstruct the damaged or lost blocks.

At the same time, it is necessary to take into account some other facts. Recent papers show that video traffic is of self-similar nature. In [18], VBR (variable bit rate) MPEG-4 video is studied from the point of view of self-similarity. Authors show that modeling video sources by short-range dependent models can be unsatisfactory. VBR video may exhibit scaling behavior and thus long-range dependence must be considered. Due to its nature, wavelet-based methods can serve as a suitable tool to evaluate self-similarity [18], [19] and to determine its parameters like H (Hurst) parameter [19].

Acknowledgement

Research described in the paper was done within the project of the Slovak Grant Agency VEGA No. 1/3117/06.

References

[1] MPEG-4 and H.263 Video Traces for Network Performance Evaluation, http://www-tkn.ee.tu-berlin.de/research/trace/trace.html, or http://trace.eas.asu.edu/TRACE/trace.html

[2] Yao Liang, Mei Han: Dynamic Bandwidth Allocation Based on Online Traffic Prediction for Real-Time MPEG-4 Video Streams, EURASIP Journal on Advances in Signal Processing, Volume 2007, 2007, pp. 51-60, http://www.ari.vt.edu/People/YLiang_Docs/YLiang02ASP2007.pdf

[3] Abdennour, A.: Short-term MPEG-4 Video Traffic Prediction using ANFIS, International Journal of Network Management, Volume 15, Issue 6, 2005, pp. 377-392, http://www3.interscience.wiley.com/cgi-bin/fulltext/112139016/PDFSTART

[4] Guoqiang Maoy, Huabing Liu: Real Time Variable Bit Rate Video Traffic Prediction, Int. Journal of Communication Systems, Vol. 20, Issue 4 (April 2007), pp. 491-505, http://www.ee.usyd.edu.au/~guoqiang/papers/Mao05realSub.pdf

[5] Haykin, S.: Neural Networks - A Comprehensive Foundation, New York: Macmillan College Publishing Company, 1994

[6] M. Oravec, J. Polec, S. Marchevský a kol.: Neural Networks for Digital Signal Processing (in Slovak language), Faber, Bratislava, 1998

[7] Hlaváčková, K., Neruda, R.: Radial Basis Function Networks, Neural Network World, No. 1, 1993, pp. 93-102

[8] Poggio, T., Girosi, F.: Networks for Approximation and Learning, Proc. of the IEEE, Vol. 78, No. 9, Sept. 1990, pp. 1481-1497

[9] Williams, R. J., Peng, J.: An Efficient Gradient-based Algorithm for On-Line Training of Recurrent Network Trajectories, Neural Computation, *2*, 1990, pp. 490-501, ftp://ftp.ccs.neu.edu/pub/people/rjw/fast-bptt-nc-90.ps

[10] Williams, R. J., Zipser, D.: Gradient-based Learning Algorithms for Recurrent Networks and their Computational Complexity. In: Y. Chauvin and D. E. Rumelhart (Eds.) Back-propagation: Theory, Architectures and

Applications, Hillsdale, NJ: Erlbaum, 1995, ftp://ftp.ccs.neu.edu/pub/people/rjw/bp-chap-95.ps

[11] Saliza Ismail, Abdul Manan bin Ahmad: Recurrent Neural Network with Backpropagation Through Time Algorithm for Arabic Recognition, 2004, http://www.scs-europe.net/services/esm2004/

[12] Vaidyanathan, P. P.: The Theory of Linear Prediction, Morgan & Claypool Publishers, 2008

[13] Wang, Z., Li, M., Kok, C. W.: Online Smooth Transmission with Rate Prediction for Video Streaming, Accepted by IEEE, Consumer Communications & Network Conference, Jan. 2007

[14] Zell A. et al: The SNNS Neural Network Simulator, University of Stuttgart, http://www-ra.informatik.uni-tuebingen.de/SNNS/

[15] Matlab, http://www.mathworks.com/products/matlab/

[16] Polec, J., Ondrušová, S., Kotuliaková, K., Karlubíková, T.: Hierarchical Transform Coding Using NURBS Approximation, Proceedings Elmar-2008: 50th International Symposium ELMAR-2008, Zadar, Croatia, 10-12 September 2008, Croatian Society Electronics in Marine, 2008, ISBN 978-953-7044-09-1, Vol. 1, pp. 65-68

[17] Pavlovičová, J., Polec, J., Keleši, M., Mokoš, M.: Detection of Transmission Errors in Block Codes Images, In: Journal of Electrical Engineering, ISSN 1335-3632, Vol. 56, No. 5-6 (2005), pp. 121-127

[18] Arifler, D., Evans, B. L.: Modeling the Self-Similar Behavior of Packetized MPEG-4 Video Using Wavelet-based Methods, Proc. IEEE Int. Conf. on Image Processing, Sept. 22-25, 2002, Vol. 1, Rochester, NY USA, pp. 848-851

[19] Vargic, R., Kotuliak, I.: Wavelet-Domain-based Estimation of Self-similarity for Multimedia Traffic, In: Proc. of Redzur International Workshop on Speech and Signal Processing, May 11, 2007, pp. 15-17, Bratislava, Slovakia

Applying Web-Mining Methods for Analysis of Student Behaviour in VLE Courses

Lajos Izsó

Institute of Applied Pedagogy and Psychology, Budapest University of Technology and Economics, izsolajos@erg.bme.hu

Péter Tóth

Centre for Teacher Training and Engineering Education, Budapest Tech Polytechnical Institution, toth.peter@tmpk.bmf.hu

Abstract: In Hungary, a lot of electronic-based syllabuses have been developed during the last 8-10 years at a most considerable cost. However, not much has been heard of their success or efficiency, as would be supported by scientific testing. First, the present study is to provide a survey of our project, which aimed at creating an integrated electronic learning environment. The processing of an education technology syllabus integrated in the Moodle virtual learning environment system took place in the passing academic year. The analysis of student behaviour in the learning environment is based on log files created on the server during the course of interaction between learners and the electronic syllabus. We analysed the learning activity of the students in the learning environment in exact numeric terms by using methods of web-mining. The second part of the study presents some of our first empirical results in this field.

Keywords: virtual learning environment, collaborative learning, constructivist pedagogy, web-mining method

1 Preliminaries

At one of the legal predecessors of the Centre for Teacher Training and Engineering Education, at the Institute for Engineering Education, an electronic syllabus package of four modules was developed in 2004 as a result of a project supported by ”Apertus” Public Foundation. The objective of the project realised under the leadership of one of the authors was the development of an electronic syllabus package with unlimited availability in space and time together with accompanying methodological aids in the topics of education technology and multimedia. The main application areas of the syllabus package are teacher training and in-service teacher training. As a result of the project, the processing of the education technology and multimedia syllabuses was completed in distant teacher training, and took place in a blended form in full time teacher training. At the beginning of the term students received the electronic syllabus on CD first, and later it was made also downloadable from an FTP server.

The electronic syllabus package was comprised of the following parts:

– Basic skills module: to introduce the elements of education technology and media development as well as the related requirements.
– Module for editing media: to aid the acquisition of skills necessary for editing digital media. This module has two parts: one of them presents the editing tools of media independent of time (still image and figure), the other presents those of media dependent on time (audio and video).
– Module for multimedia editing: to aid the acquisition of the frame program ”Authorware” necessary for the development of electronic learning program.

Two methodological aids were also developed accompanying the syllabus modules. Students of technical teacher training taking part in the training may, after graduation, participate in adult retraining and in-service training, too, where electronic-based distance learning may play a decisive role. Therefore we deemed important to elaborate recommendations helping the teacher's (tutor's/instructor's) work, which process methodological questions in connection with the development and application of electronic syllabi. In addition, a methodological guide was developed to provide more information about the individual characteristics of independent learning, with a decisive role in adult education, as well as a questionnaire to assess and evaluate learning styles.

The other project relevant to our research focuses on the pedagogical and methodological examination of the adaptability to teacher training of virtual learning environments. The specific aims of this project are

– to develop a methodology for assessing institutional requirements for networked learning and for selecting and implementing appropriate solutions, including the choice of VLE (Virtual Learning Environment);
– to create staff development and training programs to support the management and use of virtual and networked learning;
– to increase trans-national collaboration in vocational initial teacher training and develop capacity to deliver programs where this takes place;
– to investigate the specific application of VLEs in vocational initial teacher training, and to revise curricula to maximise benefits to teaching and learning processes;
– to compile and analyze data comparing various VLEs, and to disseminate this with a view to standardizing policy in vocational initial teacher training.

As a result of the ”Leonardo da Vinci” project led by Dr. Pál Pentelényi and realized in an international co-operation (involving Hungary, Finland, England, the Netherlands, Portugal and Greece), the three electronic syllabus modules were developed: Basic Teaching Skills, Computer Mediated Skills and European Collaboration. These competency modules were integrated with BlackBoard and Moodle systems. Students of technical teacher training from England, Finland, Portugal and Hungary collaborated in processing the syllabus. Tutoring students' work created an excellent opportunity to become familiar with and analyse a virtual learning environment. More can be read about the results of the project ”Virtual Electronic Learning Vocational Initial Teacher Training” (VELVITT) on the homepage velvitt.banki.hu and in the publication edited by Pál Pentelényi [1].

2 The structure of the Integrated Electronic Learning Environment

As an integration of the results of the above mentioned two projects in the autumn of 2006 we made an endeavour to create a virtual learning environment of our own (Moodle). We launched our first electronic courses, based on content developments supported by ”Apertus” Public Foundation. The first virtual course processes media independent of time. Besides electronic syllabus development we considered it important to reflect on the design aspects of the VLE system. Therefore we formed a team responsible for the creation and operation of the system to test the quality of the completed electronic course [6].

This quality testing must cover the content and structure of the syllabus as well as some elements of the user interface, such as the ergonomic, psychological and pedagogical examination of the graphical layout of the screen or the evaluation of interactive and navigational possibilities offered by the system.

Figure 1 shows the structure of the course under scrutiny. Having entered Moodle, the learner selects first the course to be studied then the most appropriate one to his needs of the particular units of the syllabus, of the learning aids and of the communicative forms on the home page. Six such objects have been incorporated in the virtual course under scrutiny, namely an electronic syllabus, a glossary, a self-check test, a forum, a check test and an uploading assignment.

The structural principles of virtual learning environments are regulated by the SCORM standard [7]. This standard distinguishes between sequence and navigation. They are to be provided only if the syllabus developer wishes to realise a navigational route more complex than the content hierarchy. An example for this may be a conditional branching from one knowledge base to another one depending on whether the student has obtained a given test score or not.

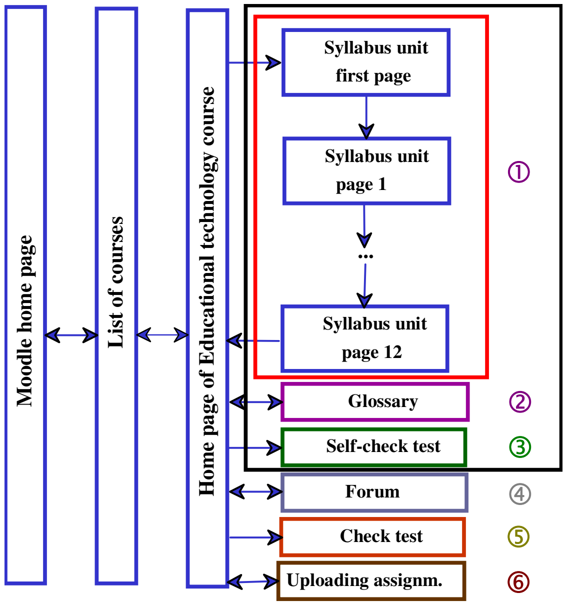


Figure 1
The structure of the syllabus content

The SCORM standard recognises two kinds of navigation: one within and the other among content objects. Navigation within the content object is realised by the object itself, while navigation among content objects is realised by the system shell, which is Moodle in the present case.

Navigation within the content object always has to be created by the syllabus developer and it may be a simple hyperlink, pointing at one syllabus element after another, Java script and frame based, Java applet based and plug-in based.

Navigation among content objects may be of the following type (Fig. 2):

- *linear (a)*

  It is a step-by-step survey of the syllabus content. Having studied a page the learner will either move on to the next page or return to the previous one.

- *hierarchical (b)*

  It is an upgraded version of linear navigation. According to its simplest version, the learner may select from a hierarchical table of contents the subsequent syllabus unit, which Moodle is to represent.

- *grid-like (c)*

  It is the two-dimensional arrangement of syllabus elements. With this type, students may make their selection from the possible hierarchies. It may prove practical in the case of arranging syllabus elements in space and/or time.

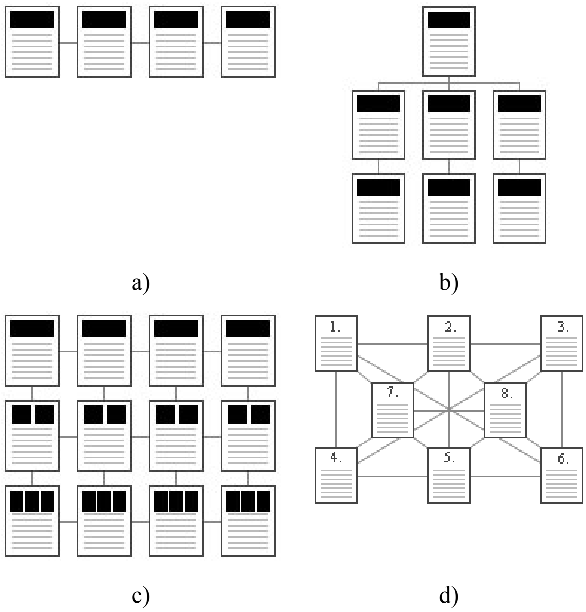


Figure 2
The basic type of navigation

- *net-like (d)*

  Net-like navigation makes it possible for students to navigate from any page to another one within the syllabus content. This yields a freedom of browsing and exploring the syllabus. The net-like structure faces the student with a decision to select the suitable direction of progress. Besides its flexibility, it may often result in losing one's way, in "lapses". This method may prove most effective in the case of students with a good learning ability.

3 Collaborative Learning in VLE

Self-directed studying must not be confused with independent studying that the student does on his or her own without supervision or contact with fellow students. Moreover, a clear distinction needs to be made between using the Internet as a channel for distributing material and teaching an online course. One of the clear strengths of an online course is the opportunity for adding the element of social networking to distance learning. The teacher has got an essential role in this, but also ways of interacting with other learners should be provided. Gilly Salmon points out that the groups of learners always tend to exchange views and ideas, and that people enjoy learning from the experience of others. She emphasises that to offer these benefits to learners, a considerable amount of group work is needed to be included within any learning program. [8]

According to Salmon, the key to active and interactive online learning lies in bringing students into greater interaction and group participation. She thinks that from these modest beginnings a new body of practice will build up around "e-tivities" that will transfer to new technologies, as they become available. Salmon also mentions that the need for skilful e-moderations will not disappear, regardless of how sophisticated and fast-moving the technological environments become. E-moderator adds the real value to learning techniques by designing and running "e-tivities". In her opinions the students learn to use the system through five stages. Each stage requires participants to master certain technical skills. Each stage calls for different e-moderating skills. At first at stage 1 (*Access and motivation*), the participants interact only with one or two others. After stage 2 (*Online socialisation*), the number of others with whom they interact, and also the interaction frequency, gradually increases, although stage 5 often results in a return to more individual pursuits. At stage 3 (*Information exchange*), the participants engage in mutual exchange of information. The form of co-operation occurs, whereby each person supports the other participants' goals. At 4th stage (*Knowledge construction*) the course-related group discussions develop and the interaction becomes more collaborative. Collaboration requires an active sharing of information and intellectual resources amongst the participants. Finally, at stage 5 (*Development*) the participants look for more benefits from the system to help themselves in achieving personal goals and reflect on the learning processes. [8]

According to constructivist pedagogical approach, learning environments should keep the activity, intentionality and collaboration for students. Activeness means that the student is in a key role in his or her own learning. The student is actively engaged in the learning process, processing information. Activeness leads to students taking responsibility in their learning. Intentionality refers to the learners' active attempts to achieve a particular cognitive goal. Striving to reach the goal makes the learner think – and thus also learn – more. Collaboration comes from the students' natural tendency to form communities in which the members can benefit from each others' skills and social support.

The most VLE system has some communicative and non-communicative elements for collaboration.

1 Chat

The Chat module allows participants to have a real-time synchronous discussion via the web. This is a useful way to get a different understanding from each other and also get the topic being discussed – the mode of using a chat room is quite different from the asynchronous forums. The Chat module contains a number of features for managing and reviewing chat discussions.

2 Forum

This activity can be the most important – it is here that most discussion takes place. A forum can be structured in different ways, and can include peer rating of each posting. The postings can be viewed in a variety for formats, and can include attachments. By subscribing to a forum, participants will receive a copy of each new posting by email. A teacher can impose subscription on everyone if they want to.

3 Assignment

Assignment allows the teacher to specify a task that requires students to prepare digital content (any format) and submit it by uploading it to the server. Typical assignment includes essays, projects, reports, and so on. This module includes grading facilities.

4 Workshop

A Workshop is a peer assessment activity with a huge variety of options. It allows participants to assess each other's project achievements, as well as exemplar projects, in a number of ways. It also co-ordinates the collection and the distribution of these assessments in various ways.

4 Web-Mining Method

Data mining involves the use of sophisticated data analysis tools to discover previously unknown, valid patterns and relationships in large data sets. These tools can include statistical models, mathematical algorithms, and machine learning methods (algorithms that improve their performance automatically through experience, such as neural networks or decision trees). Consequently, data mining consists of more than collecting and managing data, it also includes analysis and prediction.

Web-mining is a very effective data mining approach developed in the internet-based segments of the business world. In fact, it is applying data mining for sophisticated traffic analysis of websites based on the so-called “logfiles” continuously being created on the server machine of the content provider [2] [3]. Its aim is to increase the efficiency of the given web-sites.

These “logfiles” of different formats are in principle very rich sources of information about the activities of visitors. The problem here is just the contrary of the usual: we have not too little, but too much information. The size of the logfiles in the business sphere – depending, of course, on the number of actual visitors and the studying time – is quite often in the range of several hundreds of MBs or even GBs. This huge body of mainly technical information has to be filtered, transformed, and processed so that valuable piece of information characterising the visitors' behaviour and motivation could be extracted. [4]

For practical web-mining purposes, based on our earlier experiences, we propose the Clementine WebMining tools [5]. The "user-centred" philosophy of these tools is in perfect harmony with the concepts of modern marketing, ergonomics, and pedagogy.

This new approach, as opposed to the traditional "page-centred" philosophy, puts the users' goals and intentions to the centre, and designs the services of the system accordingly. As a matter of fact, as we experienced, quite successfully.

Several examples of analytic approaches that proved to be successful and which are supported by very sophisticated software modules – called here "streams" – often containing intelligent learning algorithms:

- Visit and User Segmentation (E-ChannelUser RFM Classifications, User Mode Determination, Visit Branding).
- Web Site Activity and User Behavior (Visit Activity Variances, Identifying Undesirable Behavior, Lifetime Conversion Tracking, Points of Abandonment, User Activity Focus, Visit Activity Funnels, Navigational Usage).
- Activity Sequence Analysis (Most Common Activity Sequences, Eventstream Visualisation).
- Propensity Analysis.
- Advanced User Segmentation.
- Targeting Online Promotional Activity, and Campaign Performance Measurement.

Some part of these streams can immediately be adapted to important problems in the world of Internet-based teaching materials.

5 Analysis of Student Behaviour by Web-Mining Method

At the Institute of Applied Pedagogy and Psychology of the Budapest University of Technology and Economics – with the active participation of the Centre for Teacher Training and Engineering Education of the Budapest Tech Polytechnical Institution – a research group has been composed aiming at revealing advantageous practical application possibilities of web-mining for the usage of educational materials based on Clementine tools. This group has about 14-16 active members, mainly from the staff members and PhD students of the two institutions.

The professional study of the learners' interaction with Internet-based educational materials makes possible identifying the real actual usage modes, from which well-established conclusions can be made concerning both the efficiency of these materials and the concrete obstacles of efficiency. By the way of careful targeted redesign these identified obstacles can be removed.

In the process of developing electronic educational materials appropriate web-mining methods therefore can provide the possibility of exact intermediary "sampling measurements": the development process after each "sampling measurement" can adopt a course determined by how the learners received the actual version of the material provided. To the identified professional, scientific, didactic, usability, software ergonomic (mainly concerning navigation and information presentation) etc. problems quick redesign answers can be given, which in turn can be tested in the next cycle.

This approach is radically different from the traditional ones, as it is not based on some "representative sampling" concerning the interaction of the learners and the material, but of the contrary, all interactions of all learners can be analysed at the fine resolution of single keystrokes and mouse clicks. This analysis, therefore, is not based on samples more or less representative of the target population (the set of possible learners), it is rather all inclusive in this respect.

An important requirement of successful web-mining is that the analyst be able to interpret the results gained from different models in terms of learner/user experiences. This also means: only those could be really successful who knows the web-mining methods and tools, the actual educational material, and also the actual learners equally deeply enough.

6 Case Studies

Now we present our first results in this field.

Approx. 50 learners of technical teacher training took part simultaneously in processing the Educational technology course, all their activities performed in Moodle learning environment were registered in a log file ("combine log file") by the server. The processing of this log file was performed by the SPSS Clementine web-mining programme. Here we are going to present the first results exposed by quality testing in connection with the students' learning activity, the structure of the syllabus as well as the navigational opportunities.

The course comprises three modules (basics, digital image editing, digital picture editing) and 2-3 syllabus units per module. As shown in Fig. 1, syllabus units are linear and normally of 15-18 screen pages in size.

Visit Page Funnel

Period: 11-febr.-07 - 25-?pr.-07

Page	% of Total Visits That Started Activity	# of Visits	% of Visits Started	# of Visits Dropped	Dropoff %
/moodle/file.php/37/idofuggetlen/dk1/001.html		144	100.00	-	-
/moodle/file.php/37/idofuggetlen/dk1/022.html		45	31.25	99	68.75
/moodle/file.php/37/idofuggetlen/dk2/001.html		23	15.97	22	48.89
/moodle/file.php/37/idofuggetlen/dk2/018.html		15	10.42	8	34.78
/moodle/file.php/37/idofuggetlen/dk3/001.html		12	8.33	3	20.00
/moodle/file.php/37/idofuggetlen/dk3/012.html		11	7.64	1	8.33
/moodle/mod/quiz/view.php		6	4.17	5	45.45
/moodle/mod/quiz/attempt.php		5	3.47	1	16.67
/moodle/mod/quiz/review.php		5	3.47	0	0.00

Report generated on 10-j?n.-07

self check test → view.php, attempt.php & review.php

Figure 3
The complete learning process in the module ”Digital picture editing”

Visit Page Funnel

Period: 11-febr.-07 - 25-?pr.-07

Page	% of Total Visits That Started Activity	# of Visits	% of Visits Started	# of Visits Dropped	Dropoff %
/moodle/file.php/37/idofuggetlen/dk1/001.html		144	100.00	-	-
/moodle/file.php/37/idofuggetlen/dk1/002.html		65	45.14	79	54.86
/moodle/file.php/37/idofuggetlen/dk1/003.html		54	37.50	11	16.92
/moodle/file.php/37/idofuggetlen/dk1/006.html		45	31.25	9	16.67
/moodle/file.php/37/idofuggetlen/dk1/012.html		33	22.92	12	26.67
/moodle/file.php/37/idofuggetlen/dk1/022.html		28	19.44	5	15.15
/moodle/mod/quiz/view.php		10	6.94	18	64.29
/moodle/mod/quiz/attempt.php		8	5.56	2	20.00

Report generated on 10-j?n.-07

self check test → view.php & attempt.php

Figure 4
Details of the learning process in the first syllabus unit

By the application of the stream called “Visit Page Funnel” in the SPSS Clementine programme the realisation of the parts of learning processes became demonstrable. It can be seen on Fig. 3 that the three subsequent syllabus units and the self-check test of digital picture editing were processed by a very small percent of students during a visit. ”Dropping off” was most significant (68,75%) during the first syllabus unit within the complete learning cycle. A modified adjustment of stream could also prove that giving up the processing of the syllabus is typical throughout the syllabus unit, although to a decreasing extent. ”Dropping off” is most significant after pages 1-2 (Fig. 4). In fact, this screen page can be regarded as the table of contents for the particular syllabus unit. Traditional lecture notes are also typically processed in the way that students, before settling down to actual studying, open the table of contents at random, leaf through the syllabus to see whatever there is to be learnt. This is what happened here as well.

Visit Page Funnel

Period: 11-febr.-07 - 25-?pr.-07

Page	% of Total Visits That Started Activity	# of Visits	% of Visits Started	# of Visits Dropped	Dropoff %
/moodle/file.php/37/idofuggetlen/da1/001.html		70	100.00	-	-
/moodle/file.php/37/idofuggetlen/da1/018.html		33	47.14	37	52.86
/moodle/file.php/37/idofuggetlen/da2/001.html		24	34.29	9	27.27
/moodle/file.php/37/idofuggetlen/da2/016.html		18	25.71	6	25.00
/moodle/file.php/37/idofuggetlen/da3/001.html		15	21.43	3	16.67
/moodle/file.php/37/idofuggetlen/da3/012.html		10	14.29	5	33.33

Report generated on 10-j?n.-07

Figure 5

The complete learning process in the module ”Digital image editing”

Visit Page Funnel

Period: 11-febr.-07 - 25-?pr.-07

Page	% of Total Visits That Started Activity	# of Visits	% of Visits Started	# of Visits Dropped	Dropoff %
/moodle/file.php/37/idofuggetlen/da1/001.html		70	100.00	-	-
/moodle/file.php/37/idofuggetlen/da1/002.html		38	54.29	32	45.71
/moodle/file.php/37/idofuggetlen/da1/003.html		36	51.43	2	5.26
/moodle/file.php/37/idofuggetlen/da1/006.html		34	48.57	2	5.56
/moodle/file.php/37/idofuggetlen/da1/012.html		27	38.57	7	20.59
/moodle/file.php/37/idofuggetlen/da1/018.html		24	34.29	3	11.11

Report generated on 10-j?n.-07

Figure 6

Details of the learning process in the first syllabus unit

The number of students who started the learning process by first doing the self-check test and then continued by one of the syllabus units is negligible. However, the role of the test was examined from another point of view, too. We did not provide the module ”Digital image editing” with a self-check test, being curious to see its effect on the learning process. It turned out that several electronic messages arrived through the system, enquiring about the lack of the self-check test. According to Figures 5-6, ”dropping off” during both the complete learning cycle or a part of it decreased in the case when no self check test was attached to the particular module.

With the help of “Clickstream Visualisation” it can be shown where students have arrived from at a selected screen page and where they are moving forward during a particular learning process. The most frequent direction of syllabus processing is indicated by the bold line in Fig. 7. The figure shows that the number of those students who within the first syllabus unit of the module ”Digital picture editing” moved from screen page 1 to page 3 is quite high (33). The reason for this must obviously be that students supposed screen page 2 to belong to the introductory part of the chapter and they wanted to move on to the substantial parts.

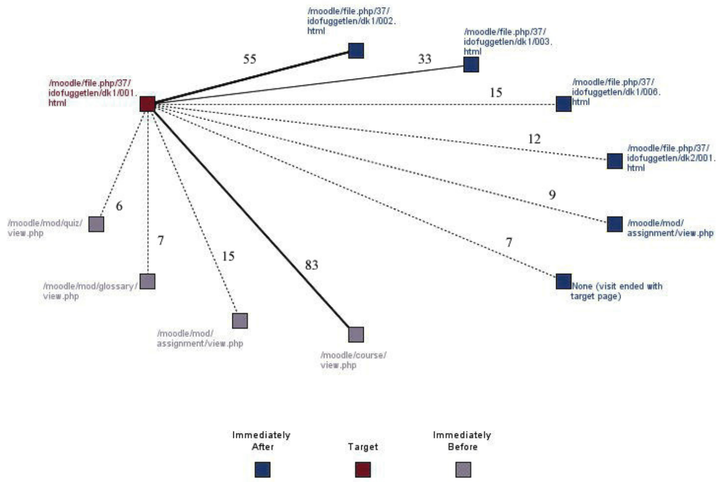


Figure 7

The representation of the learning process in the first syllabus unit

The streams called “Most Common Clickstreams” are appropriate for the presentation and analysis of complete learning processes. This helps to follow through learners’ progress in processing the syllabus. It may be shown where the learning process interrupted and whether the time dependent media (narrative audio, animation, video) in the electronic syllabus were eventually played or not. It can be seen that ”dropping off” was more significant in the case of longer syllabus units. It is practical to maximise the syllabus units in 14-15 screen pages.

Animations incorporated in the syllabus greatly facilitated the understanding of the syllabus, since processes of program application had to be learnt and animation is an excellent tool for this. Animations can be replayed and stopped at any number of times. Students certainly took these opportunities. The stream called “Most Common Clickstreams” is highly appropriate for the examination of these processes.

According to Fig. 8, narrative explanations also played a decisive role in processing the syllabus. Primarily explanations to aid understanding were audible here. This list was made by “Page Usage Metrics” stream.

Top 100 Pages By Average Time Spent Per Page

Period: 11-febr.-07 - 25-?pr.-07

Page	Average Time Spent on the Page (Seconds)
/moodle/file.php/37/idofuggetlen/dk3/hangok/003.mp3	1041
/moodle/file.php/37/idofuggetlen/da1/fogalomtar.html	660
/moodle/mod/forum/discuss.php	591
/moodle/login	558
/moodle/file.php/37/idofuggetlen/e2/hangok/008.mp3	555
/moodle/file.php/37/idofuggetlen/da2/hangok/006.mp3	498
/moodle/file.php/37/idofuggetlen/dk1/fogalomtar.html	460
/moodle/file.php/37/idofuggetlen/dk1/hangok/015.mp3	455
/moodle/file.php/37/idofuggetlen/da2/hangok/013.mp3	337
/moodle/file.php/37/idofuggetlen/da3/hangok/004.mp3	321
/moodle/mod/forum/view.php	319
/moodle/file.php/37/idofuggetlen/dk1/hangok/014.mp3	280
/moodle/file.php/37/idofuggetlen/e1/hangok/009.mp3	270
/moodle/file.php/37/idofuggetlen/dk2/hangok/004.mp3	266
/moodle/mod/forum/post.php	257
/moodle/file.php/37/idofuggetlen/da1/hangok/017.mp3	251
/moodle/mod/glossary/index.php	244

Figure 8

The series of subsequent learning operations in the first syllabus unit

Conclusions

The following conclusions are to be drawn after the evaluation of the results.

- Within a particular module students do not prefer the complete learning cycle, that is, the subsequent processing of syllabus units and the completion of the self-check test. During one visit, the processing of one, or occasionally two syllabuses was dominant.
- The so-called reverse learning cycle, that is, the preference of self-check tests to information imparting parts was not typical.
- Within a particular syllabus unit, mainly with respect to page 1-2, a significant ”dropping off” was seen, which is to be interpreted as students' orientation preceding actual learning.
- The analysis also indicated that time dependent media (narrative audio, animation, video) play a decisive role in processing the syllabus. They significantly promote the understanding of the syllabus.

Taking the above aspects into consideration, the directions of the electronic learning environment upgrade may be formulated as follows.

- It is practical to maximise the syllabus units in 14-15 screen pages.
- It is practical to incorporate self-check tests, similarly to the glossary, in the electronic syllabus, particularly at the end of the syllabus unit, the accomplishment of which is the prerequisite for opening the following syllabus unit.

- During development, a page finder window was also placed at the bottom of each page. This interfered with the sequential processing of the syllabus and resulted several times in the omission of substantial pages. A little more detailed exposition of the first, i.e. the introductory page makes it redundant to break the sequence.
- The inclusion of further time dependent media in the electronic syllabus makes learning more productive and efficient in this environment.

References

[1] Pál Pentelényi (ed.): Virtual Learning Environments – Training Material. Ligatura Ltd., Budapest, 2006, p. 131

[2] János Abonyi (ed.) Adatbányászat a hatékonyság eszköze. (Data mining as Device of the Efficiency). ComputerBooks, Budapest, 2006, p. 400

[3] László Ketskeméty, Lajos Izsó: Bevezetés az SPSS programrendszerbe. Módszertani útmutató és feladatgyűjtemény statisztikai elemzésekhez (Introduction to SPSS. A Methodological Guide and a Workbook for Statistical Analyses). Eötvös Kiadó, Budapest, 2005, p. 459

[4] Lajos Izsó: Applying Web-Mining Methods for Quality Assurance of Internet-based Educational Materials. (A revised and extended version of the plenary presentation given at the "13th Multimedia in Education Conference") Journal of Applied Multimedia, 2/II/2007, pp. 15-20

[5] SPSS Inc.: Web Mining for Clementine® 1.5. User's Guide Powered by NetGenesis, Chicago, 2005, p. 89

[6] Péter Tóth: Virtual Learning Approach in Vocational Initial Teacher Training. Proceedings of the International Conference on Engineering Education, 2007, Coimbra, ID 76

[7] SCORM 2004 3rd Edition, version 1.0, ADL, 2006, http://adlnet.gov

[8] Gilly Salmon: E-tivities. The Key to Active Online Learning. Taylor & Francis, London, 2002, p. 244

Predictive Control Design Based on Neural Model of a Non-linear System

Anna Jadlovská, Nikola Kabakov, Ján Sarnovský

Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic
anna.jadlovska@tuke.sk, nikola.kabakov@gmail.com, jan.sarnovsky@tuke.sk

Abstract: This paper provides two approaches for design of Generalized Predictive Control (GPC) algorithm for a non-linear dynamic system. In the classical approach of the GPC strategy the recursive method of least square is considered for calculating of the linearized model parameters from the known analytic description of a particular non-linear system. The other purpose of this paper is to show an intelligent approach in which a feed-forward neural network (Multi Layer Perceptron – MLP) is used for modeling the same non-linear system within the frames of the predictive control. The possibility of on-line estimation of an actual parameters from the off-line trained neural model of the non-linear system by means method of the instantaneous linearization in each sample point is considered in GPC algorithm design. The validity of classical and neural GPC strategy is tested by computer simulations in Matlab/Simulink language using architecture of S-functions of the library PredicLib.

Keywords: non-linear dynamic system, neural model, parametric estimation, predictive control algorithm

1 Introduction

Model Predictive Control (MPC) is one of the most wide spread advanced control techniques for dynamic systems in the industry. The main idea of MPC algorithms is to solve an optimization problem in order to find the control vector trajectory that optimizes the cost function over a future prediction horizon [2], [3].

This paper provides two approaches for the design of Generalized Predictive Control (GPC) algorithm for non-linear systems. In the classical approach of GPC strategy (part 2) the recursive least square method (RLSM) is considered for the calculation of the linearized model parameters from the known analytic description of the dynamic system [8]. In the intelligent approach is a feed-forward neural network (Multi Layer Perceptron – MLP) used for modeling and the estimation of the actual parameters of the non-linear system which are considered in GPC design (part 4-6). The neural model of non-linear system is typically trained in advance, but the GPC controller is designed on-line using the parameter estimation from the neural model. The main idea of this paper is to show how on-line estimation of the actual parameters from off-line trained neural model using the gain matrix is applied in the GPC algorithm [5]. The neural model of non-linear system is linearized by means method of the instantaneous linearization in each sample point and the result of this linearization technique – the estimated parameters from neural ARX model (NARX) of the system are used for design of the GPC algorithm (part 6). The practical simulations by the language Matlab/Simulink, Neural Toolbox and the library PredicLib [6] of this paper illustrate that the classical GPC and the neural GPC strategies using linearization technique can be used for predictive control of a particular non-linear dynamic system.

2 GPC Algorithm - Classical Approach

The calculating of the cost function for optimal control for the *k*-th step can be considered as

$$J(k) = \sum_{j=N_1}^{N_2} e^2(k+j/k) + \sum_{j=1}^{N_u} \lambda \Delta u^2(k+j/k) \tag{1}$$

where for the predictive horizons holds $N_1 \geq 1$, $N_2 \geq N_1, 1 \leq N_u \leq N_2$, λ is the positive weight coefficient and $e(k+j/k) = \hat{y}(k+j/k) - r(k+j/k)$ is the prediction error, where $\hat{y}(k+j/k)$ is the predicted system output value and $r(k+j/k)$ is the system output required value. By [8] the most SISO (Single-Input/Single-Output) systems when is considered the operation around a particular set-point and after linearization can be described by the linear discrete *AutoRegresive model with eXternal input* (ARX). Using the modification of ARX model we can get CARIMA (*Controlled AutoRegresive Integrate Moving Average*) model which is used in GPC algorithm [2] and is described by eq. (2)

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) + \frac{C(q^{-1})}{\Delta}\xi(k) \tag{2}$$

where $A(q^{-1})$, $B(q^{-1})$, $C(q^{-1})$ are the polynomials of the delay operator q^{-1}:

$$A(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{na} q^{-na}, B(q^{-1}) = b_0 + b_1 q^{-1} + b_2 q^{-2} + \ldots + b_{nb} q^{-nb},$$
$$C(q^{-1}) = 1 + c_1 q^{-1} + c_2 q^{-2} + \ldots + c_{nc} q^{-nc}$$

In the equation (2) $\xi(k)$ is a white noise with zero mean and $\Delta = 1 - q^{-1}$. According [2] the future output value of the system is given by (3)

$$\hat{y}(k+j/k) = G_j(q^{-1})\Delta u(k+j-1) + \Gamma_j(q^{-1})\Delta u(k-1) + F_j(q^{-1})y(k) \quad (3)$$

In the equation (3) the polynomials $G_j(q^{-1})$, $\Gamma_j(q^{-1})$ and $F_j(q^{-1})$ are calculated by solving of Diophantine equations:

$$C(q^{-1}) = E_j(q^{-1})A(q^{-1})\Delta + q^{-j}F_j(q^{-1})$$

$$B(q^{-1})E_j(q^{-1}) = G_j(q^{-1})C(q^{-1}) + q^{-j}\Gamma_j(q^{-1})$$

The predictor (3) can be written as

$$\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{\Delta u} + \boldsymbol{F}(q^{-1})y(k) + \boldsymbol{\Gamma}(q^{-1})\Delta u(k-1) \quad (4)$$

where

$$\hat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}(k+N_1/k) \\ \hat{y}(k+N_1+1/k) \\ \vdots \\ \hat{y}(k+N_2/k) \end{bmatrix}, \quad \boldsymbol{\Delta u} = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix},$$

$$\boldsymbol{G} = \begin{bmatrix} g_{N_1} & g_{N_1-1} & \cdots & g_{N_1-N_u+1} \\ g_{N_1+1} & g_{N_1} & \cdots & g_{N_1-N_u+2} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_2} & g_{N_2-1} & \cdots & g_{N_2-N_u+1} \end{bmatrix}, \quad \boldsymbol{\Gamma}(q^{-1}) = \begin{bmatrix} \Gamma_{N_1}(q^{-1}) \\ \Gamma_{N_1+1}(q^{-1}) \\ \vdots \\ \Gamma_{N_2}(q^{-1}) \end{bmatrix}, \quad \boldsymbol{F}(q^{-1}) = \begin{bmatrix} F_{N_1}(q^{-1}) \\ F_{N_1+1}(q^{-1}) \\ \vdots \\ F_{N_2}(q^{-1}) \end{bmatrix}.$$

Because two last items of (4) depends only on the previous states, we can include those into one item $\boldsymbol{f}$, then the equation of the predictor is $\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{\Delta u} + \boldsymbol{f}$. The cost function (1) can be written in the matrix form for the computing of an optimal control

$$J = (\boldsymbol{G\Delta u} + \boldsymbol{f} - \boldsymbol{r})^T(\boldsymbol{G\Delta u} + \boldsymbol{f} - \boldsymbol{r}) + \lambda \boldsymbol{\Delta u}^T\boldsymbol{\Delta u} \quad (5)$$

where $\boldsymbol{r} = [r(k+N_1), r(k+N_1+1), ..., r(k+N_2)]^T$ is the reference trajectory and λ is the weight coefficient.

The equation (5) can be written as

$$J = \frac{1}{2}\boldsymbol{\Delta u}^T\boldsymbol{H}\boldsymbol{\Delta u} + \boldsymbol{b}^T\boldsymbol{\Delta u} + \boldsymbol{f}_0 \quad (6)$$

where $\boldsymbol{H} = 2(\boldsymbol{G}^T\boldsymbol{G} + \lambda\boldsymbol{I})$, $\boldsymbol{b}^T = 2(\boldsymbol{f} - \boldsymbol{r})^T\boldsymbol{G}$, $\boldsymbol{f}_0 = (\boldsymbol{f} - \boldsymbol{r})^T(\boldsymbol{f} - \boldsymbol{r})$.

The minimum of the cost function J can be found by making gradient of J equal to zero, which leads to

$$\boldsymbol{\Delta u} = -\boldsymbol{H}^{-1}\boldsymbol{b} = -(\boldsymbol{G}^T\boldsymbol{G} + \lambda\boldsymbol{I})^{-1}\boldsymbol{G}^T(\boldsymbol{f} - \boldsymbol{r}). \quad (7)$$

The result of this equation is the trajectory consisting from the increments of the control signal and the first of them is applied on the system and is given by:

$\Delta u(k) = \boldsymbol{K}(\boldsymbol{r} - \boldsymbol{f})$, where $\boldsymbol{K}$ is the first row of the matrix $(\boldsymbol{G}^T\boldsymbol{G} + \lambda\boldsymbol{I})^{-1}\boldsymbol{G}^T$, so an actual controller output value is $u(k) = u(k-1) + \boldsymbol{K}(\boldsymbol{r} - \boldsymbol{f})$.

For the systems with the constrains on the controller output value, on the controller increment output value or on the system output value, the vector $\Delta\boldsymbol{u}$ is calculated by function ***quadprog*** of Optimization Toolbox of the language Matlab

$\Delta\boldsymbol{u}$ = ***quadprog*** ($\boldsymbol{H}$, $\boldsymbol{b}^{\mathrm{T}}$, $\boldsymbol{L}_{CON}$, $\boldsymbol{v}$, $\boldsymbol{U}_{MIN}$, $\boldsymbol{U}_{MAX}$), where $\boldsymbol{U}_{MIN} \le \Delta\boldsymbol{u} \le \boldsymbol{U}_{MAX}$, $\boldsymbol{L}_{CON}\,\Delta\boldsymbol{u} \le \boldsymbol{v}$.

The vectors $\boldsymbol{U}_{MIN}$ and $\boldsymbol{U}_{MAX}$ are the column vectors those elements are minimal and maximal values of $\Delta\boldsymbol{u}(k)$. With using the matrix $\boldsymbol{L}_{CON}$ and the vector $\boldsymbol{v}$ can be defined the system of an inequalities which insures that constrain conditions will be satisfied [6].

$$L_{CON1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}, \; v_1 = \left[\left(u_{MAX} - u(k-1) \right) . \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \right]$$

$$L_{CON2} = \begin{bmatrix} -1 & 0 & \cdots & 0 \\ -1 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \cdots & -1 \end{bmatrix}, \; v_2 = \left[\left(u_{MIN} - u(k-1) \right) . \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix} \right]$$

$$L_{CON3} = G, \;\; v_3 = \left[y_{MAX} . \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} - \boldsymbol{F}(q^{-1}).y(k) - \boldsymbol{\Gamma}(q^{-1}).\Delta u(k-1) \right]$$

$$L_{CON4} = -G, \;\; v_4 = \left[y_{MIN} \cdot \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix} + \boldsymbol{F}(q^{-1}).y(k) + \boldsymbol{\Gamma}(q^{-1}).\Delta u(k-1) \right]$$

$$L_{CON} = \begin{bmatrix} L_{CON1} \\ L_{CON2} \\ L_{CON3} \\ L_{CON4} \end{bmatrix}, \;\; v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$$

In the programmable environment Matlab/Simulink was designed GPC algorithm for the dynamic non-linear SISO system using S-functions.

Algorithm GPC for the calculation of the control signal value for the k - th step:

1 reading of the polynomials $A(q^{-1})$, $B(q^{-1})$, $C(q^{-1})$ of the linearized discrete model (2) of the non-linear system, the reference vector trajectory $\boldsymbol{r}(k)$ and the output of the system $\boldsymbol{y}(k)$,

2 cyclical calculating of the polynomials $G_j(q^{-1})$, $\Gamma_j\ (q^{-1})$ and $F(q^{-1})$ by solving Diophantine equations for $j = 1, 2, .., N_2$,

3 creating of the matrixes $\boldsymbol{G}$, $\boldsymbol{\Gamma}$ and $\boldsymbol{F}$,

4 if is required the constrain for the values of $u(k)$, $\Delta u(k)$ or $y(k)$, then continue by the step 8,

5 calculating of the feedback gain of the control vector $\boldsymbol{K}$,

6 calculating of the controller output increment $\Delta u(k) = \boldsymbol{K}\ (\boldsymbol{w} - \boldsymbol{f})$,

7 continue by the step 12,

8 creating of the matrix $\boldsymbol{L_{CON}}$ and vectors $\boldsymbol{v}$, $\boldsymbol{U_{MIN}}$, $\boldsymbol{U_{MAX}}$,

9 calculating of the matrixes $\boldsymbol{H}$ and $\boldsymbol{b}^{\mathrm{T}}$,

10 $\boldsymbol{\Delta u} = quadprog(\boldsymbol{H}, \boldsymbol{b}^{\mathrm{T}}, \boldsymbol{L_{CON}}, \boldsymbol{v}, \boldsymbol{U_{MIN}}, \boldsymbol{U_{MAX}})$,

11 $\Delta u(k)$ is the first element of the vector $\boldsymbol{\Delta u}$,

12 $u(k) = u(k\text{-}1) + \Delta u(k)$, $k = k+1$ and continue by the step 1.

3 Verification of GPC Algorithm for a Non-linear System by Simulation

The simulation model of the non-linear system consists of two tanks, one cylindrical and the other spherical with the free outflow. The inflow to the first tank is controlled by the relative open of the valve V_1. The outflow from the first tank is an inflow to the second one.

The physical variables and the parameters of the hydraulic system which is described in Fig. 1 are:

$\boldsymbol{h_1(t)}$ [m] - the water level in cylindrical tank,

$\boldsymbol{h_{1MAX}(t)}$ [m] - the max water level in cylindrical tank,

$\boldsymbol{h_2(t)}$ [m] - the water level in the spherical tank,

$\boldsymbol{h_{2MAX}(t)}$ [m] - the max water level in the spherical tank,

$\boldsymbol{M_1\ (t)}$ [kg/s] - the mass inflow to the cylindrical tank,

$\boldsymbol{M_2\ (t)}$ [kg/s] - the mass inflow to the spherical tank,

$M_3(t)$ [kg/s] - the mass outflow from the spherical tank,

$M_{Z1}(t)$ [kg/s] - the disturbance mass inflow to the cylindrical tank,

$M_{Z2}(t)$ [kg/s] - the disturbance mass inflow to the spherical tank,

$u_1(t)$ - the rise of an input outlet of the cylindrical tank,

S [m^2] - the tank's bottom area,

S_1 [m^2] - the area of the outflow of the cylindrical tank,

R [m] - the diameter of the spherical tank, ***S_2*** [m^2] is the outflow space of spherical tank.

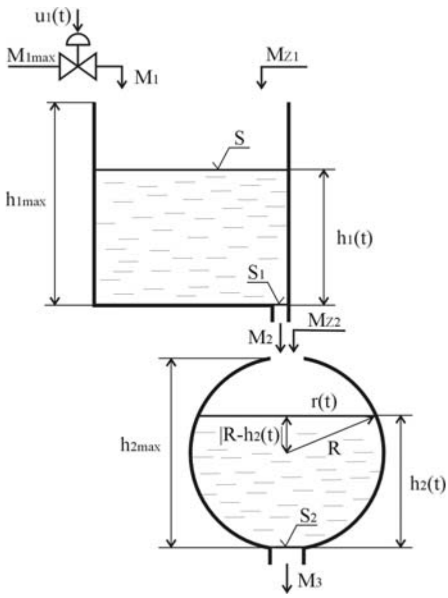


Figure 1
Model of the hydraulic system

For changing the levels $h_1(t)$ and $h_2(t)$ according to [6] it holds that

$$\rho S \frac{dh_1(t)}{dt} = M_1(t) - M_2(t) + M_{Z1}(t)$$
$$\rho\pi(2Rh_2(t) - h_2^2(t))\frac{dh_2(t)}{dt} = M_2(t) - M_3(t) + M_{Z2}(t) \tag{8}$$

The mass flows $M_1(t)$, $M_2(t)$ and $M_3(t)$ can be described by the equations (9), where k_{V1} is the constructing constant of the input outlet of the cylindrical tank and $f(u_1(t))$ is characteristic function of the valve V_1

$$M_1(t) = k_{V1} f(u_1(t)) M_{1MAX}, \quad M_2(t) = \rho S_1\sqrt{2gh_1(t)}, \quad M_3(t) = \rho S_2\sqrt{2gh_2(t)} \tag{9}$$

The mass inflow $M_1(t)$ depends on the relative open value of $u_1(t) \in \langle 0,1 \rangle$ and on the value of maximal mass inflow M_{1MAX}. Let the function $f(u_1(t))$ be linear. Then the equation for the mass inflow can be written as

$$M_1(t) = k_{V1} u_1(t) M_{1MAX} \tag{10}$$

After inducting equations (9) and (10) to the equations (8) we obtain non-linear differential equations (11) describing dynamics of the changing levels in the simulation model - two tanks without interaction

$$\begin{aligned} \frac{dh_1(t)}{dt} &= \frac{k_{V1} u_1(t) M_{1MAX} - \rho S_1 \sqrt{2gh_1(t)} + M_{Z1}(t)}{\rho S} \\ \frac{dh_2(t)}{dt} &= \frac{\rho S_1 \sqrt{2gh_1(t)} - \rho S_2 \sqrt{2gh_2(t)} + M_{Z2}(t)}{\rho \pi (2Rh_2(t) - h_2^2(t))} \end{aligned} \tag{11}$$

By expansion to Taylor series for the set-point SP = $[u_{10,}\ h_{10},\ h_{20}]$ we obtain the linearized model, which can be written by Laplace transformation as transfer functions $F_{H_2/U_1}(s)$ and then using Z-transformation can be obtained the discrete transfer function of the dynamic system $F(z) = B(z)/A(z)$.

For verification of designed GPC algorithm was used the simulation language Matlab/Simulink. The functional block of GPC controller is included into the library PredicLib [6]. The parameters of the simulation model of the hydraulic system are $M_{1MAX} = 500$kg/s, $h_{1MAX} = 6$m, $S = 7.07$m^2, $S_1 = 0.0314$m^2, $R = 1.5$m, $S_2 = 0.0314$ m^2 and $\rho = 1000$ kg/m^3.

For the calculation of the linearized model parameters the recursive method of the least square was used for the set-point SP that corresponds to the actual value of the system output $h_2(t)$. After conversion of the discretized transfer function to time area operator z is substituted with operator q. By this way the polynomials $A(q^{-1})$ and $B(q^{-1})$ of the discrete CARIMA model (2) can be acquired. The polynomial $C(q^{-1})$ is time invariant and is chosen to be equal to one. The orders of discrete model´s polynomials are $na = 2$, $nb = 2$ and $nc = 0$.

The prediction horizon for the output of the system is chosen on ten steps forward and for the control signal on five steps, so $N_1 = 1$, $N_2 = 10$, $N_u = 5$. The sample period is $T_{VZ} = 10$s.

Next is necessary to consider the constrains for the control signal value and the system output value because $u_1(t) \in \langle 0,1 \rangle$ and $h_2(t) \in \langle 0,3 \rangle$. The weight coefficient for the control signal value increment $\Delta u(t)$ is λ=0.1.

The control scheme of the simulation in an environment Matlab/Simulink is described in Fig. 2.

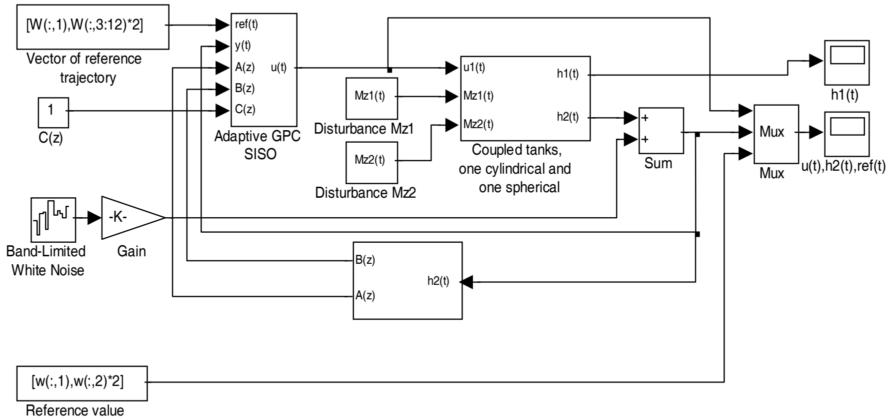


Figure 2
The simulation predictive control scheme – classical approach

On the output of the system have effect disturbance in the form of the white noise that simulates the measurement error. Also the disturbance mass inflows M_{Z1} and M_{Z2} act on the system.

The disturbance mass inflow to the cylindrical tank has value M_{Z1} = 50kg.s^{-1} and acts on in time from 2500s to 3500s. The disturbance mass inflow to the spherical tank have value M_{Z2} = 20kg.s^{-1} and acts on in time from 6500s to 7500s.

The result of tracking of the reference trajectory *ref*(*t*) by the system's output $h_2(t)$ using GPC algorithm is on Fig. 3. The output of the GPC controller – the optimal control signal $u_1(t)$ is described in Fig. 4.

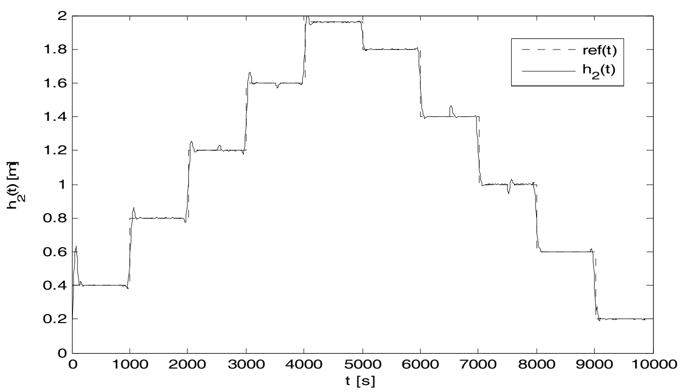


Figure 3
Tracking the reference trajectory *ref*(*t*) by the output of the non-linear system $h_2(t)$

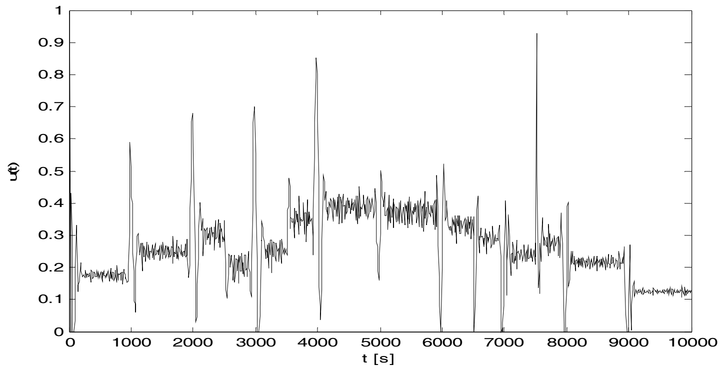


Figure 4

The GPC controller output – control signal $u_1(t)$

4 Predictive Control Structure Using Parameter Estimation from a Neural Model

Next we will consider about the GPC algorithm design for a non-linear system (hydraulic two tanks system) using the neural model of NARX structure [2], [7]. The predictive control scheme using the estimation the system parameters from off-line trained neural NARX model, which are applied in the algorithm of GPC, is illustrated on Fig. 5.

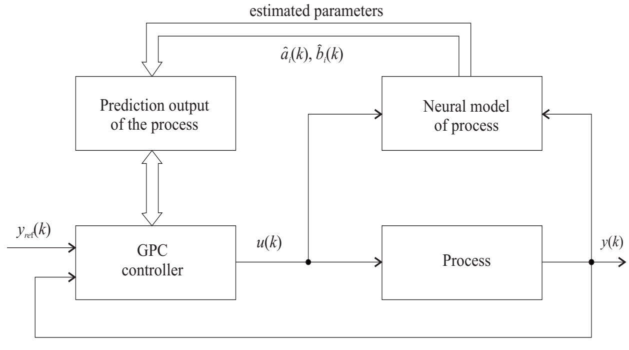


Figure 5

The predictive control scheme based on the neural model - intelligent approach

The loop consists of the controlled non-linear hydraulic system described by the equations (11), the neural process model of NARX structure and GPC controller. Because GPC algorithm needs parameters of a linear model of the dynamic system, in this paper a linear model is extracted from the non-linear neural NARX model by calculation of the gain matrix (part 5). The estimated parameters of the dynamic system from off-line trained neural model are applied for the calculation of the predictor by algebraic theory which is used in GPC algorithm [5], [6]. The optimal predictive controller output vector Δu_{opt} is determined by minimization of the cost function (1). The first term in the cost function refers to the square variation of the predicted system output from the desired reference trajectory, while the second term is added in order to limit the controller output, greater λ yields less active the controller output. The first element of the calculated optimal controller output vector is directed to the hydraulic system input. The remaining vector elements are not utilized and the entire procedure is repeated at the time $t = (k+1)T_{VZ}$, (the principle of receding horizon) [3]. The first prediction horizon N_1 is usually chosen to be 1. The choice of the second prediction horizon N_2 and the control horizon N_u are: N_2 is usually chosen as to cover the most of the control system's transient, while N_u, which denotes the significance of the future controller outputs shouldn't be greater then $N_2/2$.

5 Neural Model of a Non-linear Dynamic System

In this part we will discuss some basic aspects of the non-linear system identification using from among numerous neural networks structures only Multi-Layer Perceptron – MLP (a feed-forward neural network) [1], [9] with respect to the model based neural predictive control, where the control law is based upon the neural model. We will use in this paper a feed-forward neural network MLP with a single hidden layer. This structure is shown in the matrix notation in Fig. 6 [5], [7]. The matrix $\boldsymbol{W}_1$ represents the input weights, the matrix $\boldsymbol{W}_2$ represents the output weights, $\boldsymbol{F}_h$ represents a vector function containing the non-linear (*tanh*) neuron functions. The "1" shown in Fig. 6 together with the last column in the matrix $\boldsymbol{W}_1$ gives the offset in the network. The network input is represented by vector $\boldsymbol{Z}_{in}$ and the network output is represented by vector $\hat{\boldsymbol{Z}}_{out}$. The mismatch between the desired output $\boldsymbol{Z}_{out}$ and an aproximated output $\hat{\boldsymbol{Z}}_{out}$ is the prediction error $\boldsymbol{E}$.

The output from the neural network MLP can be written as

$$\boldsymbol{X}_2 = \boldsymbol{W}_2 \boldsymbol{F}_h \left(\boldsymbol{W}_1 \begin{bmatrix} \boldsymbol{Y}_0 \\ 1 \end{bmatrix} \right). \tag{12}$$

From a trained MLP by Back-Propagation Error Algorithm (BPA – the first-order gradient method) a gain matrix $\boldsymbol{M}$ can be found by differentiating with respect to the input vector of the network. The gain matrix $\boldsymbol{M}$ can be calculated using (12) as

$$\boldsymbol{M} = \frac{d\hat{\boldsymbol{Z}}_{out}}{d\boldsymbol{Z}_{in}} = \frac{d\boldsymbol{X}_2}{d\boldsymbol{Y}_0^T} = \frac{d\boldsymbol{X}_2}{d\boldsymbol{Y}_1^T} \cdot \frac{d\boldsymbol{Y}_1}{d\boldsymbol{X}_1^T} \cdot \frac{d\boldsymbol{X}_1}{d\boldsymbol{Y}_0^T} = \boldsymbol{W}_2 . \boldsymbol{F}_h{}'\left(X_1\right) . \boldsymbol{W}_1^* \tag{13}$$

where $\boldsymbol{W}_1^* \subset \boldsymbol{W}_1$ (excluding the last column).

The above mentioned the gain matrix $\boldsymbol{M}$ allows an on-line estimation of the actual parameters from an off-line trained neural model of the non-linear system.

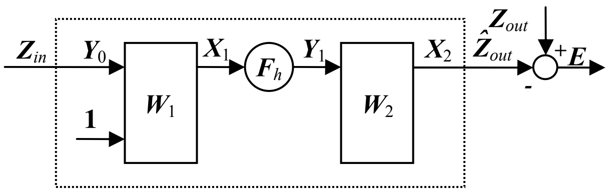


Figure 6

The matrix block diagram of the neural network MLP

With the inspiration from linear ARX model described in [8], next we will use a general model structure suitable for representing the dynamics of a wide range of non-linear system – neural ARX model (NARX) [4], [10], which is defined by

$$\hat{\boldsymbol{Y}}(k) = \boldsymbol{F}_N\left(\boldsymbol{Y}(k-1),\ldots,\boldsymbol{Y}(k-p),\boldsymbol{U}(k-1),\ldots,\boldsymbol{U}(k-m),\boldsymbol{\theta}\right)$$

$$\boldsymbol{Y}(k) = \hat{\boldsymbol{Y}}(k) + \boldsymbol{E}(k) \tag{14}$$

where $\boldsymbol{F}_N$ is the unknown non-linear vector function to be approximated, $\boldsymbol{E}(k)$ is the prediction error, p and m denote the number of delayed outputs and inputs. The non-linear mapping $\boldsymbol{F}_N$ can be approximated by a feed-forward neural network, e.g. MLP. The feed-forward neural network $\boldsymbol{F}_N$ is configured to represent the NARX model by applying p delayed values of the system output and m delayed values of the system input to the network inputs and assigning it's output $\hat{\boldsymbol{Y}}(k)$ to be $\boldsymbol{Y}(k)$, $\boldsymbol{\theta}$ is the vector of the network parameters.

An optimal value of the network parameters is usually obtained by using a training algorithm BPA that minimizes the following cost function:

$$J_{NN}(\boldsymbol{\theta}) = \sum_{i=1}^{N}\left(\boldsymbol{Y}(i) - \hat{\boldsymbol{Y}}(i,\boldsymbol{\theta})\right)^2 \tag{15}$$

on the basis of this gradient with respect to the network parameters. N in equation (15) is the length of input-output data set used for the network training.

We will consider that the neural NARX model has the input vector $\boldsymbol{Z}_{in}$ and the output vector $\hat{\boldsymbol{Z}}_{out}$:

$$\begin{aligned} \boldsymbol{Z}_{in}(k) &= \{\boldsymbol{Y}(k-1),\ldots,\boldsymbol{Y}(k-p),\ldots,\boldsymbol{U}(k-1),\ldots,\boldsymbol{U}(k-m)\} \\ \hat{\boldsymbol{Z}}_{out}(k) &= \hat{\boldsymbol{Y}}(k) \end{aligned} \tag{16}$$

After training the neural network MLP by BPA the actual gain matrix $\boldsymbol{M}(k)$ can be on-line estimated and calculated for neural ARX model as

$$\begin{aligned} \boldsymbol{M}(k) &= \frac{d\hat{\boldsymbol{Z}}_{out}(k)}{d\boldsymbol{Z}_{in}^{T}(k)} = \frac{d\hat{\boldsymbol{Y}}(k)}{d\{\boldsymbol{Y}(k-1)\ \ldots\ldots\ldots\ \boldsymbol{U}(k-m)\}^{T}} = \\ &= \{-\hat{a}_1(k)\ldots-\hat{a}_p(k)\ \hat{b}_1(k)\ldots\hat{b}_m(k)\} \end{aligned} \tag{17}$$

where $\hat{a}_i(k)$ for i=1,…, p, $\hat{b}_i(k)$ for $i = 1, \ldots, m$ are estimated parameters of the neural NARX model for step k. Using the method of an instantaneous linearization we can extract the parameters of the dynamic system from an off-line trained neural model using the gain matrix [5], [6] instead an estimation parameters of the linear model by the method of the least square. The GPC algorithm uses these estimated parameters from the neural model for control signal calculation.

6 Simulation Results of GPC Algorithm based on Neural Model – Intelligent Approach

The results of the estimation the system parameters from an off-line trained neural NARX model and their application in GPC algorithm using an algebraic theory are presented for a non-linear test SISO system – two tanks system (11). The output of the system is the water level in the second tank $y(t) = h_2(t)$. . We consider NARX model with 4 inputs and 6 neurons in the hidden layer. The activation functions in the hidden layer are „*tanh*“ functions and in the output layer is selected a linear function. The actual values of the estimated parameters can be obtained from the gain matrix $M(k)$ by (17).

The GPC algorithm in the *k - th* step consists the following calculations:

1 the real output of the system (11) $y(k)$ is measured,

2 the calculation of the linearized parameters $\hat{a}_i, \hat{b}_i$ from the neural NARX model by the gain matrix (17),

3 the estimated parameters are implemented in the GPC algorithm and an optimal control value is obtained by minimized the cost function (1),

4 an optimal control value is applied on the system input,

5 $k = (k+1)T_{VZ}$ and go to step 1.

The implementation of the resulting GPC controller including a neural network model for parameter estimation is shown on Fig. 7.

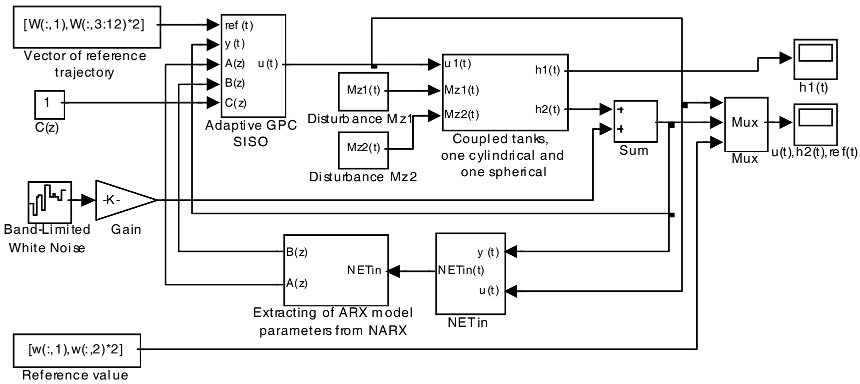


Figure 7

The GPC control scheme using neural model

The presentation of the results of GPC strategy using on-line parameter estimation from an off-line trained neural model is illustrated on Figs. 8 and 9, where the noise of measurement – the white noise is added to the output of the system (11) at $N_1 = 1, N_2 = 10, N_u = 5, \lambda = 0.1$. The disturbances input flows are added to the input of the non-linear system.

The plot on Fig. 8 compares the reference output of the system $h_{2ref}(t)$ and the actual output of the closed-loop system $h_2(t)$. A perfect model-following behaviour is achieved, although we can see an oscillating control signal on Fig. 9. This simulation example shows the possibility of an application of the neural modeling using the structure ARX known from the theory of the linear identification and also the possibility to apply GPC algorithm known from linear control theory for the control of non-linear SISO systems which have no the hard nonlinearities in the control structure in Fig. 5.

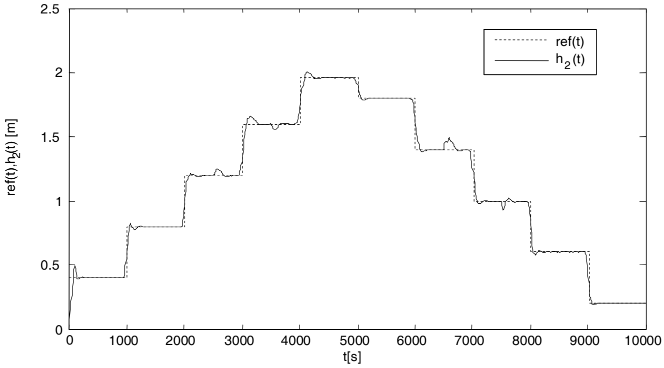


Figure 8

Tracking the reference trajectory *ref*(t) by the output of the non-linear system $h_2(t)$ using the parameter estimation from neural NARX model

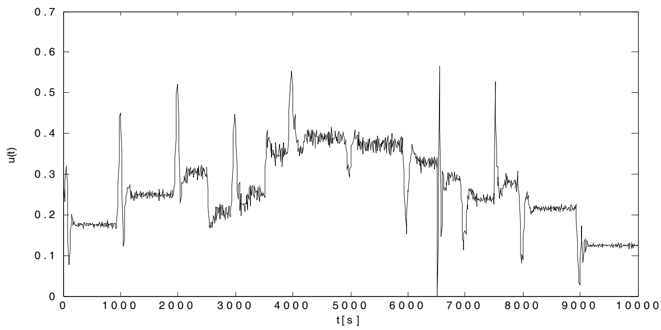


Figure 9

The control signal of the GPC controller based on parameter estimation from neural NARX model

The library *PredicLib* is created as a software tools by Simulink and Matlab language built-in functions. It contains blocks of Model Predictive Control (MPC) algorithms, concrete GPC algorithms for SISO and MIMO systems and MPC algorithms based on the state space model of the systems. PredicLib (Fig. 10) contains also the blocks for an extracting of the parameters from neural NARX model and extracting of the discrete state space model parameters from neural *Nonlinear Innovation State Space* (NISS) model of the controlled system, a block of the discrete Kalman estimator and a block for conversion of the discrete state space model to the transfer function. Some demo simulations of MPC algorithms for the control of the linear and the non-linear systems are included in the library. Some of the predictive control algorithms blocks demo applications created by using Matlab Web Server are accessible Virtual laboratory *CyberVirtLab* [*http://cyberneticsmws.fei.tuke.sk/MatlabWebServer_welcome*].

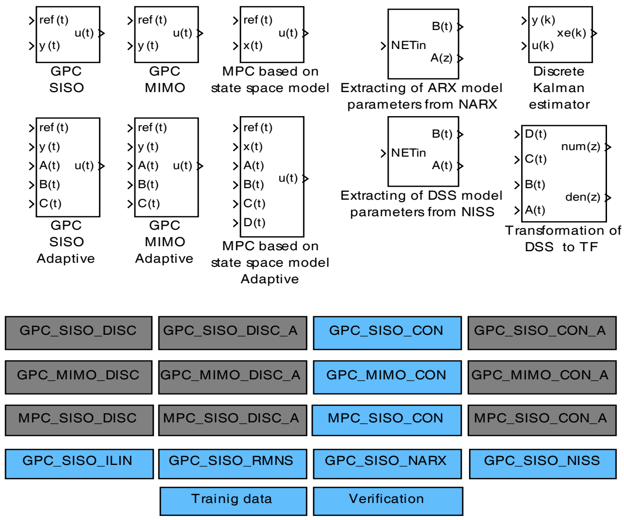


Figure 10

Screenshot of the library PredicLib from Simulink

Conclusion

In this paper are presented two approaches - classical and intelligent for the design of GPC algorithm for a non-linear system. The disadvantage of the classical approach of GPC strategy using as linearization technique for an estimate parameters of the dynamic model is that analytic description of non-linear system must be given. If analytical model of the dynamic system is unknown the better solution is to use an intelligent approach – GPC based on the neural NARX or NISS model for MIMO non-linear systems. In this paper is neural NARX model trained as an one-step predictor for a non-linear SISO system. After training this NARX model was used for on-line estimation of the system parameters which allow to calculate a linear predictor. This linear predictor of the system was used for solving of an optimization problem of GPC algorithm.

The practical simulations by the language Matlab/Simulink, Neural Toolbox and PredicLib illustrate, that this intelligent neural GPC control strategy using linearization technique by the gain matrix produces more excellent performance for control of the non-linear system as GPC strategy using of the classical approach.

Applying the principle of an instantaneous linearization to the GPC design gives tremendous advantages over the conventional nonlinear predictive control design.

Acknowledgment

This research has been supported by the Scientific Grant Agency of Slovak Republic under project Vega No. 1/0617/08 Multiagent Network Control Systems with Automatic Reconfiguration.

References

[1] S. Chen, S.Billings, P. Grant: Non-linear System Identification Using Neural Networks, International Journal of Control 51, No. 6, pp. 1191-1214, 1990

[2] D. W. Clarke, C. Monthadi, P. S. Stuffs: Generalized Predictive Control, Automatica, Vol. 23, pp. 137-148, part 1, part 2, 1987

[3] C. E. Garcia, D. M. Prett, M. Morari: Model Predictive Control: Theory and Practise – A Survey, Automatica, Vol. 25, pp. 335-348, 1989

[4] A. Jadlovská, J. Sarnovský: Using of Dynamic Neural Models in Feedforward Control Structure", in Automatizace - Journal of Automation, No. 4, Vol. 46, pp. 236-241, Praha, ISSN 0005-125X, (in Slovak), 2003

[5] A. Jadlovská: Modeling and Control of Dynamic Processes Using Neural Networks, FEI-TU, Košice, Informatech Ltd., p. 173, ISBN 80-88941-22-9, (in Slovak), 2003

[6] N. Kabakov: Predictive Control of Non-linear Processes, Thesis to Dissertation Exam, FEI-TU Košice, 53 pages, (in Slovak), 2006

[7] I. J. Leontaritis: Input-Output Parametric Models for Non-linear Systems, Part 1, Part 2, International Journal of Control, No. 2, Vol. 41, pp. 303-344, 1985

[8] L. Ljung: System Identification, Theory for User, Prentice Hall, 1987

[9] K. S. Narendra, A. U. Levin: Identification and Control of Dynamical Systems Using Neural Networks, IEEE Transaction on Neural Networks 1, No. 1, pp. 4-27, 1990

[10] M. Norgaard: NNSID Toolbox, Technical report, Department of Automation, DTU, 1997

The Application of Multiple Variable Methods in the Segmentation of the Domestic Consumer Market According to Value System

Mónika Fodor

Department of Marketing and Management, Károly Róbert College
Mátrai u. 36, H-3200 Gyöngyös, Hungary, mfodor@karolyrobert.hu

Ágnes Csiszárik-Kocsir

Faculty of Economics, Budapest Tech
Tavaszmező u. 17, H-1084 Budapest, Hungary, kocsir.agnes@kgk.bmf.hu

Abstract: In our present paper we examine the role of the value system and lifestyle as reflected in our primary research carried out among a representative consumer sample.
In our opinion the differentiating effect of the individual value system as secondary segmentation criterion can be reflected in the market of life assurance, the basis of our research. By means of factor analysis we created value system structures and with the help of cluster analysis the main criteria of value system-based segments were outlined.
As a result of the research we concluded that the individual value system was one of the most determining factors in taking out life assurance. The consumer segments formed alongside the value system have more differentiated attitudes than those of life assurance. Getting to know these groups more thoroughly and deeply offers opportunity for the „suppliers" of the life assurance market to create an effective marketing mix.

Keywords: value system, factor analysis, cluster analysis

1 Introduction

Value system and lifestyle can affect human behaviour as well as consumer behaviour. That can be the reason why the traditional, socio-demographic criteria are not enough for the segmentation of the consumer market and to explore the inner logic of the factors determining consumer behaviour based on the marketing practice these days [2]. The conditions of life and value structures are playing a more and more significant role in the formation of consumer preferences so their segmentation role has also been given a greater emphasis. [1]

In our primary research we examined the role of the value system and lifestyle on a consumer sample. In our mind the differentiating impact of the individual's value system as a secondary segmentation criterion can also be felt on the market of those taking out life assurance, which is the basis of our research, as well. By means of factor analysis value system structures have been formed and using the method of cluster analysis the main criteria of value system-based segments were characterised.

In our opinion the individual's value judgement has an impact on the concept of safety and the need for safety. In our present study we present the differences shown in the comparative analysis of consumer groups formed as a result of multi-variable methods and their willingness for taking out life assurance. The objective of the research is to survey the savings habits and the knowledge and opinion of population of the country about life assurance. In the survey we also examined what the proportion of savings was when compared to total revenue, how well-known the different forms of savings were, why the respondents put aside funds, what forms of life assurance they knew, what they had taken out so far and what factors influenced them in decision making, what the main reasons for the acceptance or refusal of life assurance were.

2 The Market of Life Insurance

2.1 Theoretical Framework to Interpret the Life Insurance Market

Life insurance policies occupy a more and more essential role in the fast changing life of society. Modern world poses a multitude of risks to the sense of security and health of individuals due to the extreme stress and strain, therefore a greater and greater part of the people strives to reduce the risk they and their families face – even though to some extent. The need for security is not a product of the 21th Century; it has a prominent place in Maslow's 'need – hierarchy' model as early as in the middle of the 20th Century (Maslow 1943). According to this, the aim of the insurance is to protect the policy-holder from the harmful effects of an uncertain future. In insurance four roles must be mentioned: the insurer, the policy-holder, the signatory and the recipient. In most cases, these four roles do not make up four separate participants, because the policy-holder can be the same (Szabó – Viharos, 2001).

There are two great segments in the insurance market: life insurance policies and non-life-insurance policies. In this study we are going to deal with the former.

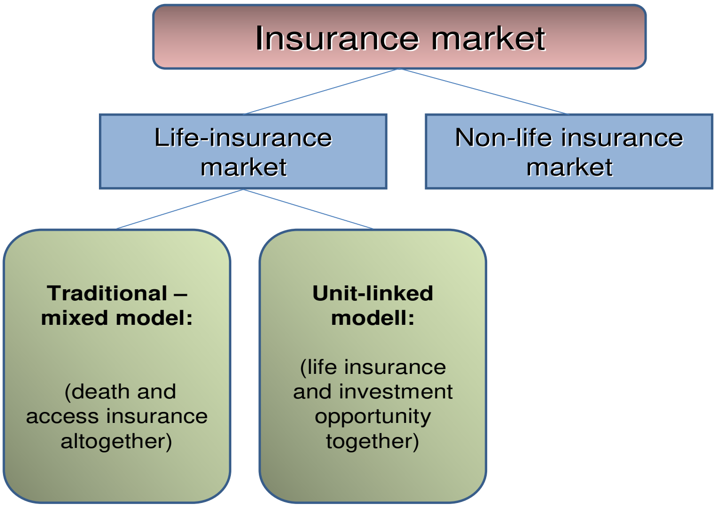


Figure 1
Segments of the insurance market

Source: Own compilation based on MABISZ (2008)

Consequently, insurance is a risk management method, which is employed both by the insurer and by the policy-holder.

In sum, we can say that an unit-linked insurance form combined with investment is more profitable than the traditional model. However, these forms, due to their high capital intensity, are less popular than the traditional comparable forms. Both parties are anxious to handle and minimize risks. The risks run by either party are shown in following risk matrix:

Table 1
Risk matrix

Source: Dögei (2003)

	INSURER	CUSTOMER
Traditional life insurance	• death • investment • premium risk	• death • premium risk
Unit-linked life insurance	• death	• death • Investment • premium risk

The insurance market, similarly to other markets, has a demand and supply side. The meeting of the two sides is supported by mediators, who are insurance agents or insurance brokers. This study is aimed at analysing the demand side, focusing on the household sector. The actors of the demand and supply side of insurance are shown in the diagram below:

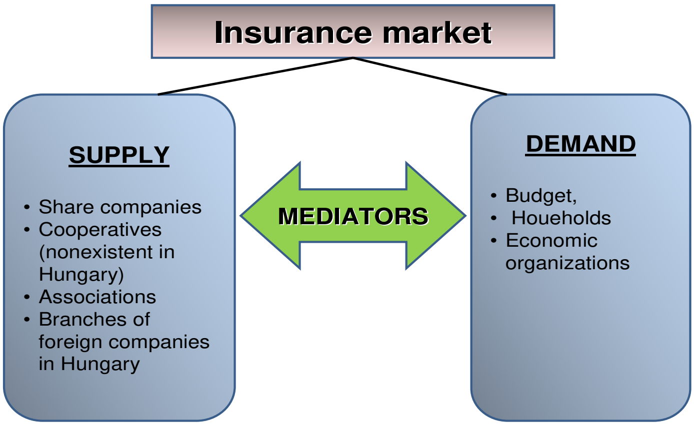


Figure 2
The demand and supply side of the insurance market

Source: Own compilation

2.2 Characterization of the Domestic Market of Life Insurance

In compliance with the information given in the introduction of this chapter, we can say that the life insurance market is becoming more and more prominent within the insurance market, which is fully proven by the share in the incomes of the Hungarian insurance companies. The proportion of the year 2002 of 40.9% increased, to a great extent, to 54.7% in 2007, which is explained by the increasing number of environmental risks and potential risk factors. Nevertheless, a number of employers take out a policy for their employees as an „extra provision". The ratio of the life insurance incomes is shown in the diagram below:

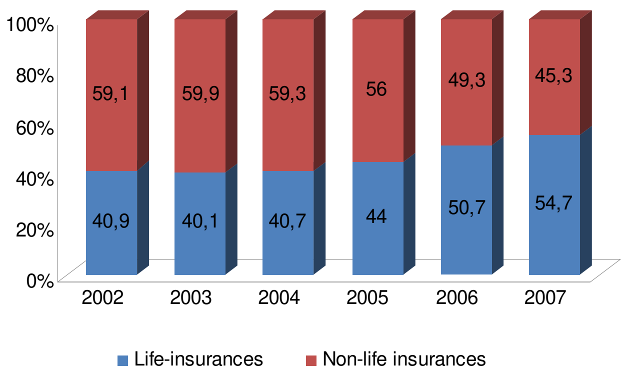


Figure 3
The ratio of life insurance policies in the total income

Source: Based on MABISZ (2004, 2005, 2006, 2007, 2008) own compilation

The headway of the life insurance branch is even more clearly shown by the increase of the earned premiums from this form of insurance:

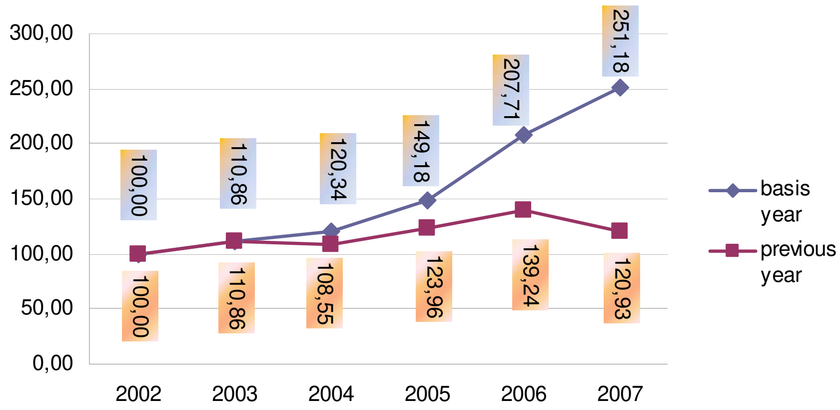


Figure 4

Extent of the increase on the earned premiums for life insurance

Source: Own edition based on MABISZ (2004, 2005, 2006, 2007, 2008)

The amount of the earned premiums on life insurance was only HUF 202,517 million. This value increased about two and a halffold in 2007, so the amount of the end of 2007 came close to HUF 508,680 million. Considering the increase compared to the value of the previous year, an outstanding ratio is with an increase of 40% in 2006. It was a year when the share of the life insurance sector exceeded the value of the non-life-insurance branch. Since 2006 the life insurance branch has been clearly superior within the insurance premium incomes.

An essential rearrangement has also occurred among the participants of the insurance market for the last six years. The biggest actor of the market was the ING, covering more than 28% of the market, which decreased essentially to 23% until 2007.

In 2002, AEGON was second (17%), and Generali-Providencia third (14%). In 2007 AVIVA Insurance was second (11.3), and Generali-Providencia remained third with a greatly reduced market share (10.9%). Further novelties of the last year are the headway of further actors of the market and further intensification of the market competition. Further participants of the top competitors include OTP Garancia Biztosító (10.3%) and AEGON Biztosító (8.5%) fallen back to fifth place. Despite the decrease in the market share no competitor in the market has suffered a decrease in the earned premium from the activities, which is a clear explanation of growing demand for the product.

3 The Methodology and Background of the Research

To survey the knowledge and opinion of the population in Hungary about savings habits and life assurance, a standard questionnaire based on questioning was made in October 2006 that comprised the whole population ensuring representation by gender, age group, residence and the type of residence. As a result, 1220 questionnaires were sent back that could be assessed by using SPSS 13.0 programme.

Among the results of this survey we wish to focus on the practical approach of segmentation possibilities highlighting the connection between life assurance and value system in the present paper.

In the course of the research the adult population of Hungary was regarded as basic multitude. The desired sampling composition was formed according to the 2005 yearbook of the Hungarian Central Statistical Office so that it might be representative of sex and age at the local level, i.e. the main averages of the sample and the basic multitude might coincide.

We compiled the quota form necessary for conscious sampling in compliance with the above-mentioned demographic criteria, whereby it became possible in a face-to-face interview to obtain a population sample representative of age and sex at the regional level. A complete publication of the survey results would extend the size of this paper, so we have to confine ourselves to present partial research results.

During the research the SPSS 11.5 software was used for a statistical analysis of the data.

The data analysis was carried out with single-variable (frequency distribution, average, dispersion) and with multivariable statistical methods (factor and cluster analysis). From among the multivariable methods, the aim of the factor analysis presented above in this paper was to investigate the relations based on the interrelation between the variables and the reduction of the number of the variables with the least possible loss of information. (Sváb, 1979) Thus it was possible to draw conclusions from the transformed multitude identical to the original multitude. (Lehota, 2001) With this procedure, the questions of what common components or factors were hidden behind a certain variable, and, also, what the relationship was like between the variable groups, were answered. The different factors influence the changes in the same variable with different weighting, and, as well, the same factor exerts an influence of different weight on the different variables, which defines the factor weight. In the course of the factor analysis, we determined the number of factors with orthogonal rotation, within this with the varimax procedure.

A major aim of the research programme was to become acquainted with the relations of the value system-based groups with life insurance. For this, the observation units were arranged into homogeneous groups with a cluster analysis. For the development of a systematized structure of value system-based attitude patterns the K-means clustering procedure was adopted. The aim of this procedure is to create groups, whose members are similar to each other by a particular criterion, and simultaneously, the differentiation between the groups is ensured. In the course of the K-means procedure, we determined the number of the clusters in advance, and the respondents were distributed among these groups by the algorithm according to the characteristics of their value system. On the basis of the variance analysis we examined whether a significant deviation could be found between at least two clusters in the case of the contributing value factors. We examined the criteria that influenced the respondents among the sociodemographic characteristics. As a result of this, the groups could also be characterized according to the primary segmentation criteria presented above.

4 The Results of the Research

4.1 The Esteem of Value Factors

The value system of the respondents in our survey was examined based on the Rokeach value structure successfully applied in sociologic research by using some of its elements. During our research we started from the assumption that common values or the value system were determinant for human behaviour, including consumer behaviour, to a great extent. The value system of the individual is a big influence on the way of selecting a certain service, i.e. on the character of the attitude towards a life insurance policy as well (Horváth – Fürediné Kovács – Fodor, 2005).

The sociological and antropological theories, which provide a basis for the exploration of the inherence between the factors determining consumer behaviour, suggest close relation between the value system and consumer behaviour (Boedekker, 1996).

During the comprehensive analysis of the value judgement of the individual we made use of the Rokeach value system (1973), one of the best known in marketing science from the methods measuring value adopted in psychology. In compliance with this, we compiled a value list, which was also tested during the test interview of the questionnaire. Based on the experiences of the test interview, a final list was made for assessment.

The ranking order of the factors presents well that the traditional values of safety, calmness, well-balanced life lead by pushing all hedonistic values in the background – as presented by the systematic results of Table 1.

Table 2
Mean values of ranking value factors

From: survey results, 2006 (N=1220)

Value factors	Sample mean
Traditional values	
Health	4,87
Family happiness	4,58
Happiness	4,5
Calm, balanced life	4,49
Inner harmony	4,36
Good human relationships	4,34
Peaceful world	4,2
True friendship	4,2
Financial well-being	4,16
Freedom	4,06
Moral acknowledgement	4,02
Taking care of others	4
Source of pleasure for others	3,96
Thriftiness	3,62
Hedonistic values	
Comfortable life	3,96
Affording something good	3,87
Plenty of leisure time	3,63
Enjoyable life	3,41
Career, professional success, self fulfilment	3,41
Hobby	3,15
Exciting life	2,73

4.2 Factor Analysis of Values

To find out which values make up correlating factor groups, factor analysis was carried out based on the value estimates of the respondents and, as a result, three well separable factors were outlined (Table 3):

- the group of „*traditional-classical”* values,
- the system of „*hedonistic”* values and
- the group connected to the values of „*maturity, financial well-being*”.

Table 3
Factor groups of values

From: survey results, 2006, N=1220 (KMO index: 0.78, explained cumulative variation: 72.03%; varimax rotation)

Values	Factor groups		
	Factor 1 factor weight	Factor 2 factor weight	Factor 3 factor weight
Harmonic life	0.767	0.300	0.212
Peaceful life	0.758	0.121	0.222
Taking care of others	0.756	0.114	0.239
Source of pleasure for others	0.745	0.170	0.248
Happiness	0.723	0.372	0.297
Good human relationships	0.721	0.365	0.106
Family happiness	0.695	0.079	0.382
True friendship	0.688	0.496	0.081
Calm, balanced life	0.620	0.062	0.513
Moral acknowledgement	0.600	0.465	0.237
Exciting life	0.034	0.827	0.036
Enjoyable life	0.095	0.807	0.281
Career	0.312	0.738	0.023
Hobby	0.284	0.714	0.053
Plenty of leisure time	0.254	0.602	0.466
Freedom	0.442	0.476	0.304
Affording something good	0.163	0.362	0.331
Comfortable life	0.211	0.326	0.747
Financial well-being	0.341	0.401	0.583
Thriftiness	0.115	0.050	0.422
Health	0.274	0-.057	0.379

In the group of „*traditional--classical*” values there are such classical values where rather the dominance of communal, social values is more decisive than that of individual objectives. The main point is the group of values emphasising the importance of families, friends and social life.

The group of „*hedonistic* values” can also be well characterised where there are the factors of exciting, pleasurable life dominating together with the experience-oriented ones.

Here the dominance of rather short-term objectives is discernible and we meet fresh, dynamic values that significantly differ from the „*traditional-classical*" values.

It is interesting to note that freedom was interpreted in two different ways by the members of the sample as it can either belong to both of the group of „*hedonistic*" and „*traditional-classical*" values at the same time regarding its factor weight. All this can be explained by the difference in interpretation, so it refers to the fact that it can both mean natural freedom as a „traditional" value and an attribute to human life as well as an experience-individual oriented one emphasising the freedom of the individual, in which case it can be classified as a „hedonistic" one in our mind.

The group of values connected with *„maturity, financial well-being"* outlines another dimension of „traditional" values in which we can find the system of values in connection with financial stability, calmness, being well-balanced, health and financial life dimension referring to security and long-term effects. Instead of experience-oriented values, we can find the ones highlighting break-out of the rat-race of everyday life, purity, calmness and stability.

4.3 Cluster Analysis Based on the Value-System Structure

To examine whether there is a group-forming dimension outlined based on the value system as a secondary segmentation criterion, cluster analysis was carried out among the members of the sample. As a result, three clusters could be defined in the examination of the value system as a result of the analysis corresponding to the three factors. The segments were termed as:

- „Family-centred, aiming at security"
- „Hedonistic"
- „Caring well-off.

Regarding the three clusters, well-distinguished characteristic value structures can be found that can be described by divergence from the main average of the sample mean with a number of specific differences and even from one another with regard to certain factors.

4.3.1 The Characterisation of the Segment Termed „Family-centred, Aiming at Security"

The first cluster consists of those family-centred aiming at security. They regarded comfortable, calm, balanced life, thriftiness and family happiness more important than the mean sample. At the same time, when compared to the sample, they underestimated exciting, enjoyable life, career, success and hobby. The group is rather characterised by a traditional, clear, classical, mature value system.

Table 4
The value structure of the clusters[1]

From: survey results, 2006

Value factors	Cluster 1	Cluster 2	Cluster 3	Sample mean
Freedom	3.65	**4.24**	4.11	4.06
Enjoyable life	***2.66***	**4.22**	***3***	*3.41*
Exciting life	***1.61***	**3.67**	***2.37***	*2.73*
Comfortable life	**4.37**	***4.16***	***3.54***	3.96
Calm, balanced life	**4.68**	***4.23***	4.65	4.49
Health	4.88	***4.58***	4.96	4.87
Good human relationships	3.82	4.26	**4.72**	4.34
Plenty of leisure time	3.28	**4.07**	3.39	3.63
Financial well-being	4.2	**4.32**	4	4.16
Moral acknowledgement	3.48	4.05	4.29	4.02
Affording something good	3.49	**4.25**	***3.66***	3.87
Thriftiness	**4.23**	3.47	3.55	3.62
Source of pleasure for others	3.81	***3.76***	**4.39**	3.96
Taking care of others	3.88	***3.74***	**4.46**	4
Family happiness	**4.74**	4.32	**4.89**	4.58
Career, professional success, self fulfilment	***2.11***	**4.01**	3.63	*3.41*
Hobby	***2.1***	**3.69**	3.3	*3.15*
Peaceful world	4.12	3.92	4.7	4.2
Happiness	4.25	4.46	**4.82**	4.5
Inner harmony	4.06	4.22	**4.82**	4.36
True friendship	3.55	4.3	**4.63**	4.21

4.3.2 The Characteristics of the Value Judgement of the "Hedonist" Segment

The members of the second cluster follow hedonistic values. They prefer enjoyable life rich in adventures and for them financial well-being is more important than average to afford themselves something good. Career, plenty of leisure time and hobby are all the most overestimated factors in this group when compared to the mean of the sample. At the same time, the somehow contradictory things to the instant enjoyment of life, namely, comfortable, calm and balanced life were judged to be less important than average. Those who prefer

[1] Cluster 1 = The value structure of the clusters „Family-centred, aiming at security" (N = 1220)
Cluster 2 = Value structure in the segment of „Hedonists" (N = 478)
Cluster 3 = Value structure in the group of „Caring well-off" (N=488)

self- remuneration and individual features belong to this group. It is proved by the fact that taking care of others and giving a source of happiness for others are less important than average for this group.

4.3.3 Value System Specifications in the Case of the Segment" Caring Well-Off"

In the „caring well-off" cluster we can see those aiming at security and who are socially sensitive. The members of the sample have a lot in common with those in the first cluster with a mature value system but in this group values like the source of enjoyment for others, peaceful, harmonic family life and friendship are significantly overestimated. This is the most socially sensitive cluster. Self-remuneration and making individual lifestyle exciting and enjoyable are pushed in the background and communal values are emphasised instead of the individual ones.

4.4 The Demographic Characteristics of the Groups Formed on the Basis of Thevalue System

The group "Family-centred, aiming at security" is typically (51.2%) made up by families living with a child or children older than 19 years old so adults. Mainly the middle-aged group and the representatives of the older generation over 60 (49%) make up this second segment.

Half of the family-centred group (50.4%) possess secondary school degree who typically (42.5%) live on a monthly net income of 60-100 thousand Ft.

These demographic characteristics predestine the value system that characterise this sample group. We think this group is the potential target market of life assurance. This is also supported by the fact that the proportion of those with life assurance is the highest here (60%).

The value system preferred by the "Hedonist" cluster is in accord with the demographic features of the cluster: typically, the members of the group represent young people. It is also proved by the proportion of 59.8% of the age group between 18-39. This situation in life can also explain the fact that this cluster is primarily the group of young people seeking adventure and self-remuneration who live for themselves rather than for the others.

In the „Hedonists" while examining the proportion of gender we can state that men are overrepresented (55.6%). A unique criterion in the cluster is the fact that one-fifth of its members do not possess self-supporting income. This can be explained by the age of the members as almost 40% is made up by young males between 18 and 29. That is why we think this cluster cannot be regarded as the potential base of life assurance. We can find the highest female dominance (57.8%) in the cluster "Caring well-off".

Another criterion is that a significant part of the group is city dweller and member of the group of the middle aged who typically raise their child (children) older than 19. The members of this segment live under better than average financial circumstances as it is proved by the fact that one-third of the group live on a monthly net income of 100-150 thousand Ft and a further one-third on 150-200 thousand Ft.

The need for taking care of others can be the main *motivating basis* for taking out life assurance. We think it so because in this sample *the proportion of those with life assurance is outstandingly high* (68%).

4.5 Differences, Discrepancies between the Clusters

In our esteem the *Hedonist* group is the sharpest contrast to the followers of traditional, secure values of those who think of tomorrow like the “Caring well-off” and “Family-centred, aiming at security” groups.

In contrast with the more mature age groups they *do not wish to invest*, save, rather they live for today and prefer instant presents so it is not surprising that *the proportion of those with life assurance is the lowest here* (48%).

However, the members of the *“Caring well-off” and “Family-centred, aiming at security”* groups have a lot in common. Both groups *judged the classical, traditional values important* but, on the other hand, in the case of the third group the priority of social sensitivity and giving a source of enjoyment to others are exceptionally high so they also can be regarded socially sensitive to the greatest extent in relation to the two emphatic groups.

In our opinion these two groups are open to the service of life assurance on offer to the greatest extent as knowing themselves as well as the others in safe appears as the greatest criterion at these groups.

We think the clusters outlined on the basis of this value system have special criteria that correlate with their attitude to life assurance, namely, their openness or reservation to it. Furthermore, in our mind, *on the market of life assurance characterised by a more and more intense competition monitoring the groups that can be well characterised by the value system can be an important objective in marketing as the value structure of these segments predestine the range of marketing activities ranked to be effective in their case.*

In our research we wanted to find out what factors influence consumers when taking out life assurance as well as what the motives are behind their decision. In our mind the value ranking of the individual is one of the most important determining factors so we want to utilise the opportunity given by this value-system-based classification in the segmentation procedure. The consumer groups outlined by this secondary segmentation criterion can be described by a peculiar value structure that highlights the motivation system of taking out life assurance and thus can help get to know the range of marketing activities suiting the single group's demands and expectations the best. In our opinion the segmentation based on value system and lifestyle as a secondary segmentation criterion very significant in target market marketing can be well matched to characterise consumer market, too and, as a consequence, to outline certain client groups in the peculiar market of life assurance. To find out which values make up correlating factor groups, factor analysis was carried out based on the value estimates of the respondents and, as a result, three well separable factors were outlined: the group of „traditional-classical" values, the system of „hedonistic" values and the group connected to the values of „maturity, financial well-being".

After the factor analysis a cluster analysis was carried out to make groupings based on the value system of the respondents and, as a result, well-structured groups showing similarities regarding value system preferences were obtained and three clusters could be defined matching the three factors: the cluster of "the hedonists", the segment of "family-centred, aiming at security" and the group of "caring well-offs".

As a result of our research these latter two groups are the most open to services offered by taking out life assurance as knowing themselves as well as the others in safe appears as the main motivating criterion in these segments. As a matter of fact, the motivation for safety is the most influential factor when taking out life assurance.

Anyway, safety motivation is the most influential and decisive factor when taking out life assurance as it is also supported by further results of our research.

Conclusions

In our present paper we examine the role of the value system and lifestyle as reflected in our primary research carried out among a representative consumer sample.

In our opinion the differentiating effect of the individual value system as secondary segmentation criterion can be reflected in the market of life assurance, the basis of our research. By means of factor analysis we created value system structures and with the help of cluster analysis the main criteria of value system-based segments were outlined.

As a result of the research we concluded that the individual value system was one of the most determining factors in taking out life assurance. The consumer segments formed alongside the value system have more differentiated attitudes than those of life assurance. Getting to know these groups more thoroughly and deeply offers opportunity for the „suppliers" of the life assurance market to create an effective marketing mix

References

[1] Association of Hungarian Insurance Companies (MABISZ) (2008): Yearbook of Hungarian Insurers 2007, http://www.mabisz.hu/publikaciok_f.html, downloaded: 2008. 05. 20. (Hungarian)

[2] Association of Hungarian Insurance Companies (MABISZ) (2007): Yearbook of Hungarian Insurers 2006, http://www.mabisz.hu/publikaciok_f.html, downloaded: 2008. 05. 20. (Hungarian)

[3] Association of Hungarian Insurance Companies (MABISZ) (2006): Yearbook of Hungarian Insurers 2005, http://www.mabisz.hu/publikaciok_f.html, downloaded: 2008. 05. 20. (Hungarian)

[4] Association of Hungarian Insurance Companies (MABISZ) (2005): Yearbook of Hungarian Insurers 2004, http://www.mabisz.hu/publikaciok_f.html, downloaded: 2008. 05. 20. (Hungarian)

[5] Association of Hungarian Insurance Companies (MABISZ) (2004): Yearbook of Hungarian Insurers 2003, http://www.mabisz.hu/publikaciok_f.html, downloaded: 2008. 05. 20. (Hungarian)

[6] Boedekker, Mika (1996): Recreational Shopping. An Environmental Psychology Approach, Competitive Paper, EMAC, Budapest

[7] Dögei, S. (2003): Insurance politics, BOI, Budapest, (Hungarian)

[8] Ágnes Hofmeister-Tóth (2003): Consumer Behaviour, Aula, Budapest (Hungarian)

[9] Ágnes Hofmeister-Tóth, Mária Törőcsik (1997): Lifestyle Research in Hungary, Marketing Management, No. 1-2, (Hungarian)

[10] Á. Horváth, A. Füredi-Kovács, M. Fodor (2005): The Effect of the Value System on Nutrition, Food, Nutrition, Marketing, Year II, No. 1-2, (Hungarian)

[11] J. Lehota (ed.) (2001): Marketing Research in Agriculture

[12] Maslow, A. (1943): A Theory of Human Motivation, Psychological Review, 50, Vol. 4

[13] Philip Kotler (1998): Marketing Management, Technical Publischer, Budapest, (Hungarian)

[14] Rokeach, M.: 1973: The Nature of Human Values, Free Press, New York

[15] J. Sváb (1979): Multivariable Approaches in Biometry, Agricultural Publischer, Budapest (Hungarian)

[16] L. I. Szabó, L. Viharos (2001): Fundamentals of Life Insurance, Polygon, Szeged (Hungarian)

Case Studies for Improving FMS Scheduling by Lot Streaming in Flow-Shop Systems

Ezedeen Kodeekha

Budapest University of Technology and Economics
Műegyetem rkp. 3, H-1111 Budapest, Hungary
Fax: +36-1-463-3271; Tel.: +36-1-463-2945
ezo12@yahoo.com

Abstract: This paper deals with scheduling problems of the Flexible manufacturing systems (FMS). The objective is to improve the utilization of FMS. Lot Streaming (LS) is used to meet this objective.

In this paper a comparative study is performed between the applications of new methods: Brute Force method (BFM) and Joinable Schedule Approach (JSA). Case studies for Flow Shop Systems (FSS) are performed. Attached independent sequence setup times are considered. It is concluded that these methods can be used effectively to solve LS problems. In the paper a general optimization mathematical model of LS for FMS scheduling problems of FSS is developed and presented.

Keywords: FMS Scheduling, Scheduling Priority Rules, Lot Streaming, Global Minimum of Production Time, Excess Time Coefficient, Brute Force Method, Joinable Schedule Approach

Abbreviation: CIM: Computer Integrated Manufacturing. CIF: Computer Integrated Factory. FMS: Flexible Manufacturing System. FSS: Flow Shop System. SPR: Scheduling Priority Rules. LS: Lot Streaming. BFM: Brute Force Method. JSA: Joinable Schedule Approach

1 Introduction

Nowadays, in modern manufacturing, Computer Integrated Manufacturing (CIM) is directing the technology of manufacturing towards Computer Integrated Factory (CIF) which is a fully automated factory.

Because CIF would involve a high capital investment, especially in its Flexible Manufacturing System (FMS), efficient machine utilization is extremely essential; machines must not stand idle. Consequently, proper FMS scheduling is required. Furthermore, for the industrialized nation, FMS must be able to meet critical challenge: to react quickly to current competitive market conditions. There are two challenges: maximize utilization and minimize production time. Of course, these quantities are interconnected and highly depend on the quality of scheduling. So, appropriate FMS scheduling must be analyzed accurately.

FMS Scheduling is a manufacturing function to schedule different machines to different jobs which may have different quantities, different processes, different setups, different process sequences, etc. organized according to a certain priority rule subject to certain constraints in order to meet one or multi-criteria.

This paper deals with FMS scheduling problem of Flow Shop System (FSS) in where all the jobs to be produced follow the same process sequence (path or route).

In this paper, FSS with attached independent sequence setup time is considered.

The objective of this paper, like the earlier ones [10, 11, 12, 13, 14], is to minimize maximum production time (makespan) close to the global minimum of production time in order to improve machine utilization.

The classic methods such as Scheduling Priority Rules (SPR) usually produce schedules with low system utilization.

In this paper, to achieve this objective Lot Streaming (LS) technique is used in which the jobs (batches) are broken and the processes are overlapped concurrently.

Many researchers studied the LS problems. For FSS, there is a lot of literature: two machines/one job (2/1) Lot Streaming with setup time was given in [1], 2-machines/ multi-jobs (2/J) with setup time was presented in [22, 2, 8], (3/1) was presented in [4], (3/J) in [21], multi-machine group, multi-job (M/J) without setup time in [7, 9], M/J with setup time using Dynamic Programming algorithm in [16], M/J with setup time using Mixed Integer Linear Programming (MILP) in [23], M/J with setup time using Genetic Algorithm (GA) was proposed in [19]. The analysis of batch splitting in an assembly scheduling environment was presented in [18]. Tabu Search (TS) and Simulated Annealing (SA) were proposed in [20]. Comprehensive review of Lot Streaming is presented in [6].

In [10, 11, 12, 13, 14], new methods to solve the LS problems for FMS scheduling were developed. These methods were named as Brute Force Method [BFM] and Joinable Schedule Approach [JSA]. These two methods have different basic ideas and different procedure. BFM is a search method and JSA is an analytical method.

In this paper a comparative study is performed between these methods through case studies for FSS.

1.1 The Content of the Present Paper

This paper begins with an introduction in Section 1 and continues in Section 2 with a problem definition. Case studies are formulated and their engineering database is given in Section 3. In Section 4 applications of Brute Force Method and Joinable Schedule Approach are given and the comparison of the results of the methods is presented. Conclusions can be read in Section 5.

2 Problem Definition

2.1 Problem Statement

The problem considered in this paper is FMS scheduling problem. The FMS consists of **different** machine groups m ($m = 1, 2 \ldots M$) to process **different** jobs (bathes) j ($j=1, 2 \ldots J$) in **different** volumes (number of parts) $n_1, n_2 \ldots n_j$ with **different** processing time of one part of job j on machine group m, this time is indicated by $\tau_{j,m}$.

Rather detailed information about the above model can be found in [13].

2.2 Global Minimum of Production Time and the Excess Time Coefficient

Let $\tau_{j,m}$ be the processing time of one part of job j on machine group m as introduced above. Then, the total load time of machine group m is indicated as L_m

$$L_m = \sum_{j=1}^{J} \tau_{j,m} n_j \tag{1}$$

The load time of the bottleneck machine group, $\boldsymbol{L_b}$ is the maximum load among all loads of the machine groups.

$$L_b = Max\, L_m = Max \sum_{j=1}^{J} \tau_{j,b} n_j \tag{2}$$

Let $\delta_{j,m}$ be the attached independent sequence setup time of machine group m to process job j. Then, the overall setup time for a machine group, when the manufacturing sequences are known, is the sum of set up times of machine group m to process all jobs.

$$S_m = \sum_{j=1}^{J} \delta_{j,m} \tag{3}$$

We suppose that the bottleneck machine group has the maximum summation of setup times. So, the total setup times of bottleneck machine group is

$$S_b = Max\ S_m = Max \sum_{j=1}^{J} \delta_{j,b} \tag{4}$$

The fulfillment of this condition is by far not trivial. But we suppose that it is valid in a number of practical cases and here we deal with these cases.

It is remarkable that the L_m *(m = 1, 2 ...M)* values for a given order of production do not depend on the order of production sequences. But the $\boldsymbol{S_m}$ *(m = 1, 2 ...M)* values depend on that. In the present paper we consider known feasible schedules for which the sequences are known, too. So, the setup times sum can be estimated, furthermore, in the case studies, for simplicity, everywhere the same setup time value was used for all parts and for the machine groups but this did not restrict the validity of the results. The number of setups is known and is the same for all of the machine groups (FSS case). In the FMS type production the setup times have small values. So, it seems to us that different relaxing assumptions concerning setup times do not affect too much the quality of system performance.

Returning to the above, **the global minimum of production time** is

$$t_g = Min\ t_{pr} = L_b + S_b \tag{5}$$

In this paper, like in the earlier ones [10, 11, 12, 13, 14], a new quantity called **Excess time coefficient** C_r is introduced to measure the goodness of FMS scheduling system.

Let t_{pr} be the makespan of the system which is the time length of completion time of the last job to leave the system. It can be defined as the maximum of the production time. The makespan is usually indicated in the literature as C_{max} (see: [17, 3]). Here we use for that $\boldsymbol{t_{pr}}$. Of course, $\boldsymbol{t_{pr} = C_{max}}$.

The excess time coefficient is defined as the ratio of makespan to the global minimum of production time:

$$C_r = \frac{t_{pr}}{t_g} \tag{6}$$

High values of C_r mean low utilization of the system. C_r never has a lower value than 1. To decrease C_r, we will use a lot streaming technique.

We remark that for job shop scheduling problems cases may exist where a value close to 1 (with the closeness determined by the setup times) may be realized. However, the given schedule may not be very easy to find.

For flow shop problems the global minimum of production time is different from the above. It may be determined as outlined in paper [10]. Nevertheless, C_r is a good quantity for comparisons.

2.3 Lot Streaming Technique

According to the lot streaming technique proposed in [10, 11, 12, 13, 14], the production batches are divided into a number of equal sub-batches, *N*. Then, the sub-batches can be processed in overlapping manner in order to achieve one or more objectives. At that makespan will decrease due to overlapping process but, at the same time, the sum of set up times will increase. For that reason, the problem of lot streaming to be solved is: What is the optimal number of sub-batches? It is a trade-off optimization problem. In this paper, two methods applied in [10, 11, 12, 13, 14] are used:

a) **Brute Force Method, BFM**

b) **Joinable Schedule Approach, JSA**

For the investigations of the features of these approaches we will use simulation methods.

2.4 Simulation Method

The objectives of using simulation technique based on Scheduling Priority Rules (SPR) for solving the given problem can be outlined as follows:

a) To select the best feasible initial schedule giving a suitable makespan value.

b) To represent Gantt charts.

c) To specify the global minimum of production time.

d) To determine the excess time coefficient.

e) To determine the utilization of the system.

2.5 Brute Force Method, BFM

BFM is a break and test method in which the initial feasible schedule of production batches is broken many times into sub-batches at certain setup time and tested until finding the suitable number of sub-batches. BFM is a search, enumeration, and optimization method.

In this paper we used a simulation computer program as described in [5]. At certain setup time we divided all batches into many possible sub-batches and then testing was made to compare the new number of sub-batches with the previous number until finding the optimum number of sub-batches in which the excess time coefficient is minimum and system utilization is maximum.

2.6 Joinable Schedule Approach, JSA

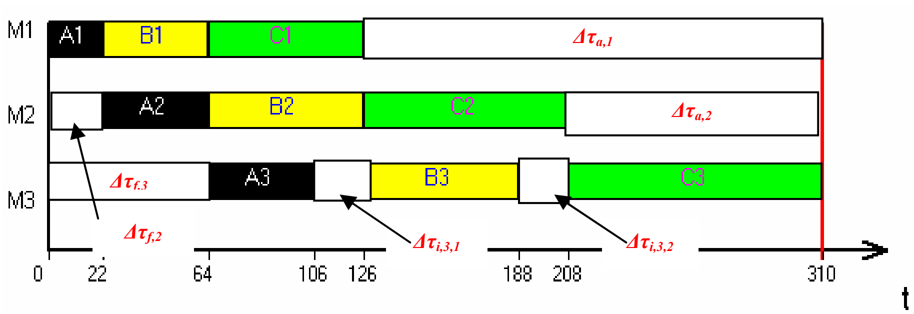


Figure 1

Gantt chart of 3/ 3 flow shop scheduling problem with idle times

Let us demonstrate the given approach for FSS cases. For demonstration we introduce an example to clarify the idle times of the system and the method how to schedule a flexible manufacturing system (FSS case). FMS consists of three machine groups (M1, M2, M3) to process three jobs (A, B, C) by different processing times. It is a 3/3 scheduling problem. We suppose that the FIFO schedule is the best feasible schedule. The Gantt chart is illustrated in Figure 1.

In Figure 1, $\Delta\tau_{f,m}$, $\Delta\tau_{a,m}$ are the front and after idle time of machine group *m*, respectively.

$\boldsymbol{\Delta\tau_{i,m}}$ is the sum of all of the inside (in-between) idle times. The total idle times of the machine groups $\Delta\tau_m$ is

$$\Delta\tau_m = \Delta\tau_{f,m} + \Delta\tau_{i,m} + \Delta\tau_{a,m} \qquad (7)$$

In FSS, there is no idle time in front of the first machine group and behind (after) the last machine group, $\Delta\tau_{f,1} = \Delta\tau_{a,M} = 0$

The idle time of the bottleneck machine group is

$$\Delta\tau_b = \Delta\tau_{f,b} + \Delta\tau_{i,b} + \Delta\tau_{a,b} \qquad (8)$$

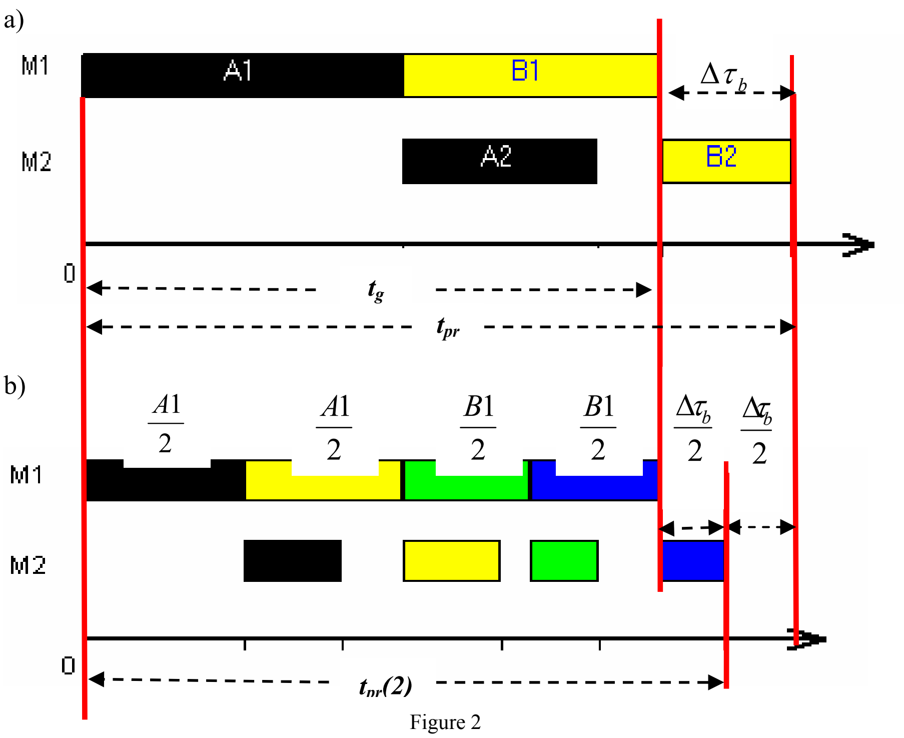


Figure 2

Gantt chart of 2/2 flow shop schedule

a) Without lot streaming b) With Lot Streaming, $N=2$

To simplify the model, let us introduce an example illustrated in Figure 2a and b. In Figure 2 Gantt chart of initial feasible schedule without lot streaming for 2/2 flow shop scheduling problem is given. We assume that the bottleneck machine group is index 1. The idle time of the bottleneck is $\Delta\tau_b$, where $\Delta\tau_b = \Delta\tau_{a,b}$, of course, $\Delta\tau_{f,b} = \Delta\tau_{i,b} = 0$. As was given, the global minimum of production time is

$$t_g = L_b + S_b$$

The makespan t_{pr} is

$$t_{pr} = t_g + \Delta\tau_b \tag{9}$$

From equation (5)

$$t_{pr} = L_b + S_b + \Delta\tau_b \tag{10}$$

Now, as proposed in [10, 11, 12, 13, 14], we divide the schedule lengths by integer number N. Then, we move the sub-batches together until they touch each other. Clearly, at the given formulation of the problem this is always possible. We remark that for job shop problems this is quite different, and the "Joinable Schedule Approach" can only be used for special schedules which are not very easy to find (see: [12]).

Let us divide the batches into 2 equal-size sub-batches (see Figures 1a and b).

The bottleneck load time value is constant, $\frac{A1}{2}+\frac{A1}{2}+\frac{B1}{2}+\frac{B1}{2}=A1+B1=L_b$

The setup times of bottleneck machine group becomes $2*n_b*\delta = 2\,S_b$

The idle time of bottleneck machine group becomes $\frac{\Delta\tau_b}{2}$

So, the makespan function becomes

$$t_{pr}(2)= L_b+ 2\,S_b+ \frac{\Delta\tau_b}{2} \tag{11}$$

If the batches are divided into N equal-size sub-batches, the makespan function will change as follows:

$$t_{pr}(N) = L_b+ N\,S_b+ \frac{\Delta\tau_b}{N} \tag{12}$$

Equation (12) needs some comment, in fact, when dividing the batches the setup times appear not only in the bottleneck section but in others too, which are forming the $\Delta\tau_b$ part. But this has a very little effect on the system performance, and so it can be neglected, as reflected in equations (11) and (12).

Dividing equation (12) by t_g, we obtain the following coefficients:

$$C_r = \frac{t_{pr}}{t_g}\ , \qquad \Psi_r = \frac{L_b}{t_g}, \qquad \theta_r = \frac{S_b}{t_g}, \qquad \Phi_r = \frac{\Delta\tau_b}{t_g} \tag{13}$$

Where C_r, Ψ_r, θ_r and Φ_r are called excess time coefficient, bottleneck global coefficient, setup relation coefficient and bottleneck idle time coefficient, respectively.

Equation (12) becomes

$$C_r = \Psi_r + \theta_r N + \Phi_r \frac{1}{N} \tag{14}$$

To minimize C_r we can differentiate C_r with respect to $\boldsymbol{N}$ and equalize to zero.

$$\frac{\partial C_r}{\partial N} = \theta_r - \Phi_r \frac{1}{N^2} = 0 \tag{15}$$

The optimum number of sub-batches is

$$N^* = \sqrt{\frac{\Phi_r}{\theta_r}} = \sqrt{\frac{\Delta\tau_b}{S_b}} \tag{16}$$

The optimum excess time coefficient is

$$C_r^* = \Psi_r + 2\sqrt{\Phi_r \theta_r} \tag{17}$$

The minimum makespan is

$$t_{pr}^* = L_b + 2\sqrt{\Delta \tau_b S_b} \tag{18}$$

The optimum excess time coefficient can be determined as

$$C_r^* = \frac{t_{pr}^*}{t_g} \tag{19}$$

2.7 Utilization and Makespan

One of the most important means to improve productivity of any system is the efficient utilization of the available resources. As mentioned above, the objective of this paper is to improve the system utilization through FMS scheduling system.

A low value of makespan implies high utilization of the machines. Utilization and makespan are interconnected quantities.

Let *U* be the initial utilization of the system; it can be computed by the following formula:

$$U = \frac{L}{M * t_{pr}} \tag{20}$$

Where ***L*** is the total load time of the system. It is determined by summation of all the processing times required to process all jobs.

U^* is the optimum utilization of the system achieved using BFM or JSA to solve lot streaming problem, and can be computed as follow:

$$U^* = \frac{L}{M * t_{pr}^*} \tag{21}$$

To evaluate the improvement of the schedule quality, we use **the productivity improvement rate *η***

$$\eta = \frac{U^* - U}{U} * 100 \tag{22}$$

3 Case Studies Characterization

In this paper, we analyze 7 different cases of LS problems of FMS scheduling for FSS, each case is characterized as a category ***S/M/J/m$_b$/O/δ,*** where ***S*** is the type of the system, ***M*** is the number of machine groups, ***J*** is the number of jobs, m_b is the bottleneck machine group index, ***O*** is the objective or criterion to measure the performance of the system, ***δ*** is the setup time.

The case studies data are introduced in Table 1: To demonstrate the content of the table we give an example which is the first case: *FSS/2/2/1/U/2*: The flexible manufacturing system is a Flow Shop System consists of two machine groups (*M=2*) to be processed two jobs (*J=2*), and the bottleneck machine group index is 1 ($m_b=1$), The objective is to obtain higher utilization *U*, the setup time (*δ=2h*).

By using LS technique and applying the two new methods, BFM and JSA, for two Scheduling Priority Rules (SPR), First In First Out (FIFO) and Minimum Slack (MS), we can find out the optimal quantities of: number of sub-batches, makespan, Excess time coefficient, utilization and Productivity improvement rate.

We can recognize from Table 1 that cases 1, 2 have same *M/J, δ* and *L* but different m_b and L_b. Cases 3, 4 have same *M/J, δ, L* and L_b but different m_b.

Case	*M/J*	m_b	L_b	*δ*	*L*
1	2/2	1	180	2	280
2	2/2	2	160	2	280
3	3/3	1	200	2	500
4	3/3	2	200	2	500
5	3/4	2	320	3	840
6	4/4	3	380	3	1180
7	5/4	5	360	4	1360

Table 1
Seven case studies of FSS with different machine group index

3.1 Engineering Database of Case Studies

Case No 1: FSS/2/2/1/U/2

Job j	n_j	***Machine group m***						T_i
		1			**2**			
		τ	**t**	**k**	**τ**	**t**	**k**	
1(A)	150	0.67	100	1	0.40	60	2	160
2(B)	200	0.40	80	1	0.20	40	2	120
L_j			***180***			100		***280***

Table 2
Database case No 1

Case No 2: FSS/2/2/2/U/2

Job j	n_j	Machine group m						T_i
		1			2			
		τ	t	k	τ	t	k	
1(A)	200	0.40	80	1	0.50	100	2	180
2(B)	150	0.27	40	1	0.40	60	2	100
L_j			**120**			160		**280**

Table 3
Database of case No 2

Case No 3: FSS/3/3/1/U/2

Job j	n_j	Machine group m									T_i
		1			2			3			
		τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.40	40	1	0.40	40	2	0.20	20	3	100
2(B)	150	0.40	60	1	0.40	60	2	0.27	40	3	160
3(C)	150	0.67	100	1	0.53	80	2	0.40	60	3	240
L_j			**200**			180			120		**500**

Table 4
Database of case No 3

Case No 4: FSS/3/3/2/U/2

Job j	n_j	Machine group m									T_i
		1			2			3			
		τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.40	40	1	0.40	40	2	0.20	20	3	100
2(B)	150	0.40	60	1	0.40	60	2	0.27	40	3	160
3(C)	150	0.53	80	1	0.67	100	2	0.40	60	3	240
L_j			180			**200**			120		**500**

Table 5
Database of case 4

Case No 5: FSS/3/4/2/U/3

Job j	n_j	Machine group m									T_i
		1			2			3			
		τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.40	40	1	0.80	80	2	0.60	60	3	180
2(B)	150	0.27	40	1	0.40	60	2	0.40	60	3	160
3(C)	150	0.40	60	1	0.53	80	2	0.53	80	3	220
4(D)	200	0.40	80	1	0.5	100	2	0.5	100	3	280
L_j			220			**320**			300		**840**

Table 6
Database of case 5

Case No 6: FSS/4/4/3/U/3

Job j	n_j	Machine group m												T_i
		1			2			3			4			
		τ	t	k	τ	t	k	τ	t	k	τ	t	k	
1(A)	200	0.50	100	1	0.40	80	2	0.50	100	3	0.40	80	4	360
2(B)	250	0.24	60	1	0.32	80	2	0.40	100	3	0.24	60	4	300
3(C)	300	0.27	80	1	0.20	60	2	0.27	80	3	0.13	40	4	260
4(D)	250	0.24	60	1	0.24	60	2	0.40	100	3	0.16	40	4	260
L_j			300			280			**380**			220		**1180**

Table 7
Database of case No 6

Case No 7: FSS/5/4/5/U/4

Job j	n_j	Machine group m															T_i
		1			2			3			4			5			
		τ	t	k	τ	t	k	τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.60	60	1	1	100	2	0.80	80	3	0.80	80	4	1	100	5	420
2(B)	150	0.40	60	1	0.53	80	2	0.40	60	3	0.40	60	4	0.53	80	5	340
3(C)	150	0.27	40	1	0.40	60	2	0.53	80	3	0.40	60	4	0.67	100	5	340
4(D)	200	0.20	40	1	0.30	60	2	0.20	40	3	0.20	40	4	0.40	80	5	260
L_j			200			300			260			240			**360**		**1360**

Table 8
Database of case No 7

4 Case Studies for BFM and JSA Applications

4.1 Application of BFM

Using **LEKIN** computer program [17] and applying BFM for the given case studies using another computer program of lot streaming given in [5, 15] we represent the Gantt charts such as in Figures 3 a, b, c of case 7. The values of C_r and U are presented in Tables 9-14.

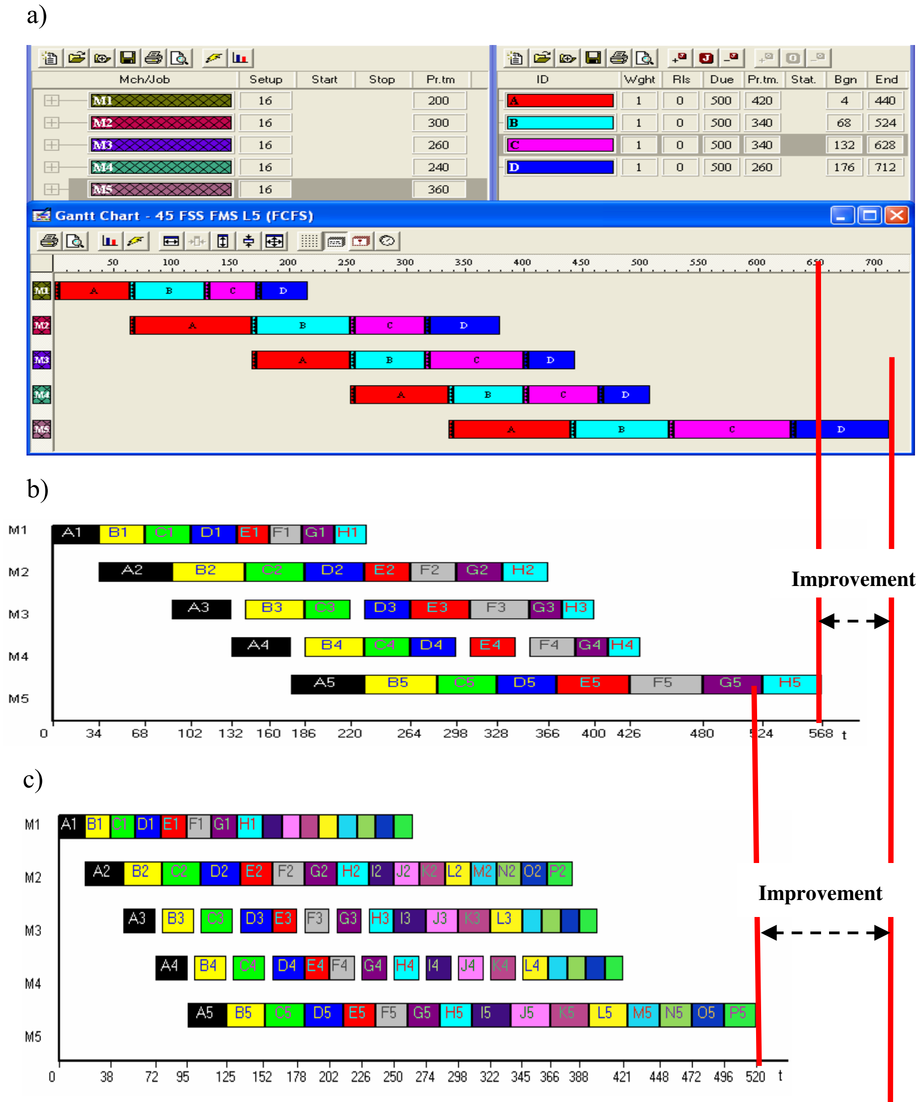


Figure 3

Gantt charts of case No 7

a)Without lot streaming b) With lot streaming, *N* =*2* c) With lot streaming, *N* =*4*

4.2 Results of BFM Applications

N	FIFO,MS		
	t_{pr}	C_r	U
1	**226**	**1,228**	**61,95**
2	210	1,141	66,67
3	**207**	**1,125**	67,63
4	208	1,130	67,31
5	210	1,141	66,67
6	213	1,158	65,73
7	211	1,147	66,35
8	215	1,168	65,12

Table 9
Case No 1

N	FIFO,MS		
	t_{pr}	C_r	U
1	**246**	**1,500**	**56,91**
2	210	1,280	66,67
3	200	1,220	70,00
4	**198**	1,207	70,71
5	**198**	1,207	70,71
6	201	1,226	69,65
7	202	1,232	69,31
8	204	1,244	68,63

Table 10
Case No 2

N	FIFO			MS		
	t_{pr}	C_r	U	t_{pr}	C_r	U
1	**350**	**1,699**	**47,62**	**310**	**1,505**	**53,76**
2	286	1,388	58,28	266	1,291	62,66
3	267	1,296	62,42	254	1,233	65,62
4	263	1,277	63,37	253	**1,228**	65,88
5	262	**1,272**	63,61	254	1,233	65,62
6	267	1,296	62,42	260	1,262	64,10
7	269	1,306	61,96	263	1,277	63,37
8	270	1,311	61,73	264	1,282	63,13

Table 11
Cases No 3, 4

N	FIFO			MS		
	t_{pr}	C_r	U	t_{pr}	C_r	U
1	**478**	**1,440**	**58,58**	**498**	**1,500**	**56,22**
2	420	1,265	66,67	420	1,265	66,67
3	409	**1,232**	68,46	410	1,235	68,29
4	409	**1,232**	68,46	409	1,232	68,46
5	414	1,247	67,63	414	1,247	67,63
6	420	1,265	66,67	419	1,262	66,83
7	425	1,280	65,88	425	1,280	65,88
8	439	1,322	63,78	440	1,325	63,64

Table 12
Case No 5

N	*FIFO,MS*		
	t_{pr}	C_r	*U*
1	***621***	***1,584***	***47,50***
2	*523*	*1,334*	*56,41*
3	*496*	*1,265*	*59,48*
4	*492*	***1,255***	*59,96*
5	*493*	*1,258*	*59,84*
6	*502*	*1,281*	*58,76*
7	*495*	*1,263*	*59,60*
8	*500*	*1,276*	*59,00*

Table 13
Case No 6

N	*FIFO,MS*		
	t_{pr}	C_r	*U*
1	***712***	***1,894***	***38,20***
2	*568*	*1,511*	*47,89*
3	*531*	*1,412*	*51,22*
4	*520*	***1,383***	*52,31*
5	*520*	***1,383***	*52,31*
6	*525*	*1,396*	*51,81*
7	*523*	*1,391*	*52,01*
8	*536*	*1,426*	*50,75*

Table 14
Case No 7

N	$C_{r\,1}$	$C_{r\,2}$	$C_{r3,4}$		$C_{r\,5}$		$C_{r\,6}$	C_{r7}
			FIFO	***MS***	***FIFO***	***MS***		
1	*1,228*	*1,5*	*1,699*	*1,505*	*1,44*	*1,5*	*1,584*	*1,894*
2	*1,141*	*1,28*	*1,388*	*1,291*	*1,265*	*1,265*	*1,334*	*1,511*
3	***1,125***	*1,22*	*1,296*	*1,233*	***1,232***	*1,235*	*1,265*	*1,412*
4	*1,13*	***1,207***	*1,277*	***1,228***	***1,232***	***1,232***	***1,255***	***1,383***
5	*1,141*	***1,207***	***1,272***	*1,233*	*1,247*	*1,247*	*1,258*	***1,383***
6	*1,158*	*1,226*	*1,296*	*1,262*	*1,265*	*1,262*	*1,281*	*1,396*
7	*1,147*	*1,232*	*1,306*	*1,277*	*1,28*	*1,28*	*1,263*	*1,391*
8	*1,168*	*1,244*	*1,311*	*1,282*	*1,322*	*1,325*	*1,276*	*1,426*

Table 15
Values of excess time coefficient of application BFM for all cases

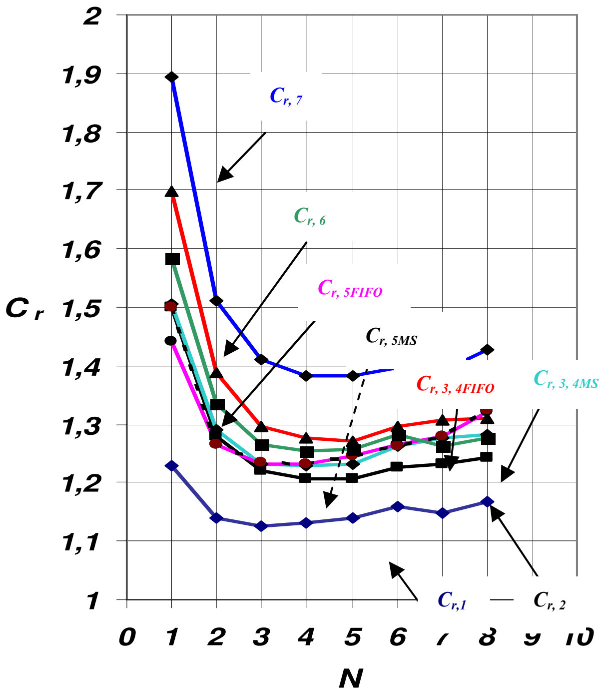


Figure 4

Excess time coefficient curves of all cases: $C_{r,\ case\ number,\ rule}$

4.3 Application of JSA and its Results

By the substitution of the given values into the equations (5, 9, 16, 18, 21, 22) we can get the results as given in Table 16. From the results given in Table 16 it can be concluded that the productivity improvement rate, for some cases, is high reached to 40.52% and low reached to 9.75%.

CASE	RULE	L_b	S_b	t_g	t_{pr}	$\Delta\tau_b$	N^*	t^*_{pr}	C^*_r	$U^*\%$	$\eta\%$
1	FIFO,MS	180	4	184	226	42	3.2	205.9	1.119	67.99	9,75
2	FIFO,MS	160	4	164	246	82	4.5	196.22	1.196	71.34	25,36
3	FIFO	200	6	206	350	144	4.9	258.78	1.246	64.40	35,24
	MS	200	6	206	310	104	4.1	249.95	1.213	66.68	24,03
4	FIFO	200	6	206	350	144	4.9	258.78	1.246	64.40	35,24
	MS	200	6	206	310	104	4.1	249.95	1.213	66.68	24,03
5	FIFO	320	12	332	478	146	3.5	403.71	1.215	69.35	18,39
	MS	320	12	332	498	166	3.7	409.26	1.232	68.41	21,68
6	FIFO,MS	380	12	392	621	229	4.3	484.84	1.236	60.84	28,08
7	FIFO,MS	360	16	376	712	336	4.5	506.64	1.355	53.68	40,52

Table 16

Results of application of JSA

Case	*Rule*	N^**(JSA)*		N^**(BFM)*	C^*_r*(JSA)*	C^*_r*(BFM)*
1	*FIFO,MS*	*3.2 ≈*	*3*	*3*	*1,119*	*1,125*
2	*FIFO,MS*	*4.5 ≈*	*5*	*4-5*	*1,196*	*1,207*
3	*FIFO*	*4.9 ≈*	*5*	*5*	*1,246*	*1,272*
	MS	*4.1 ≈*	*4*	*4*	*1,213*	*1,228*
4	*FIFO*	*4.9 ≈*	*5*	*5*	*1,246*	*1,272*
	MS	*4.1≈*	*4*	*4*	*1,213*	*1,228*
5	*FIFO*	*3.5 ≈*	*4*	*3-4*	*1.215*	*1,232*
	MS	*3.7 ≈*	*4*	*4*	*1,232*	*1,232*
6	*FIFO,MS*	*4.3 ≈*	*4*	*4*	*1,236*	*1,255*
7	*FIFO,MS*	*4.5 ≈*	*5*	*4-5*	*1,355*	*1,383*

Table 17

Optimal excess time coefficient values of BFM and JSA for all cases

4.4 Comparing the BFM and JSA Results

In Table 17, the values of C_r for seven cases for both rules FIFO and MS applying both methods BFM and JSA are presented.

The values of optimum number of sub-batches N^* for JSA are rounded to closed integer value.

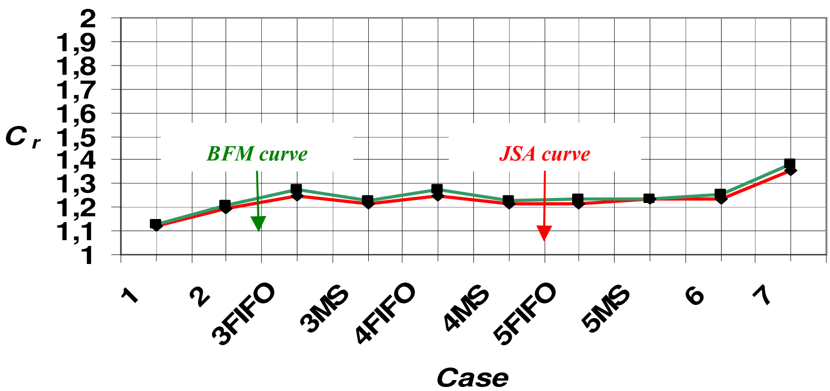


Figure 5

Optimal excess time coefficient curves BFM and JSA

Conclusions

From Table 17 and Figure 5 we can conclude that BFM and JSA can be used effectively to solve lot streaming problems of FSS. The application of both methods BFM and JSA gives almost the same results.

JSA can be used for FSS without modifying the initial feasible schedule, and there is no need for joinability test.

The optimization mathematical model of JSA developed can be used as a general optimization model of Lot Streaming used for FMS scheduling problem of Flow Shop System with an attached independent sequence setup time.

The data applied in the case studies examples are quite general. So, it may be supposed that the results are widely applicable. Namely, the analytical results obtained by JSA can be easily obtained for extended applications.

References

[1] Baker K. R: *Lot Streaming in Two-Machine Flow Shop with Set up Time,* Annual of Operations Research 57, 1995, pp. 1-11

[2] Cetinkaya, F. C.: *Lot Streaming in a Two-Stage Flow-Shop with Set up Processing and Removal Times Separated*, Journal of the Operational Research Society 45(12), 1994, pp. 1445-1455

[3] French S., B. A., M. A. and D. Phil.: *Sequencing and Scheduling: An Introduction to the Mathematics of the Job-shop*, Ellis Horwood Ltd. England, 1982

[4] Glass C. A., Gupta J. N. D. and Potts C. N.: "*Lot Streaming in Three-Stage Production Processes*" European Journal of Operations Research, 75, 1994, pp. 378-394

[5] J. Somló, A. V. Savkin, A. Anufriev, T. Koncz: *Pragmatic Aspects of the Solution of FMS Scheduling Problems Using Hybrid Dynamical Approach*, Robotics and Computer-Integrated Manufacturing 20, 2004, pp. 35- 47

[6] J. H. Chang, H. N. Chiu: *A Comprehensive Review of Lot Streaming*" International Journal of Production Research 43, 2005, pp. 1515-1536

[7] Kalir A., Sarin C.: *Evaluation of the Potential Benefits of Lot Streaming in Flow Shop Systems*, International Journal of Production Economics 66, 2000, pp. 131-142

[8] Kalir A., Sarin C.: *Constructing Near Optimal Schedules for the Flow Shop Lot Streaming Problem with Sublot-Attached Setups*, Journal of Combinatorial Optimization 7(1), 2003, pp. 23-44

[9] Kalir A. A., Sarin S. C.: *A Near-Optimal Heuristic for the Sequencing Problem in Multiple-Batch Flow-Shops with Small Equal Sublots*, Omega 29, 2001, pp. 577-584

[10] Kodeekha Ezedeen, János Somló: *Improvement of FMS Scheduling Efficiency by Optimal Lot Streaming for Flow Shop Problems*, in proceedings of 6th National Conference, BUTE, Budapest, Hungary, May 29-30, 2008

[11] Kodeekha Ezedeen and János Somló: *Improving FMS Scheduling by Lot Streaming*, Journal of Periodica Polytechnica Series, BUTE*,* 51.1, 2007, pp. 3-14

[12] Kodeekha Ezedeen, János Somló: *Optimal Lot Streaming for a Class of FMS Scheduling Problems-Joinable Schedule Approach*, in Proceedings of 11th IFAC International Symposium on Computational Economics & Financial and Industrial Systems, Istanbul, Turkey, 2007, pp. 363-368

[13] Kodeekha Ezedeen, János Somló J.: *Optimal Lot Streaming for FMS Scheduling of Flow Shop Systems*, in proceedings of 12th International Conference on Intelligent Engineering Systems, Miami, USA, February 25-29, 2008, pp. 53-85

[14] Kodeekha Ezedeen: *Brute Force Method for Lot Streaming in FMS Scheduling Problems*, in proceedings of 11th International Conference on Intelligent Engineering Systems, Budapest, Hungary, 29 June-1 July, 2007, pp. 179-184

[15] Koncz T.: *Scheduling of Production of Flexible Manufacturing Systems Using Hybrid Dynamical Method*, Thesis of Master of Science, Budapest University of Technology, Hungary, 2002

[16] N. G. Hall, G. Laporte, E. Selvarajah, C. Sriskandarajah: *Scheduling and Lot Streaming in Flow Shops with No-wait Process*" Journal of Scheduling, 6,2003, pp. 339-354

[17] Pinedo M.: *Planning and Scheduling in Manufacturing and Services*, Springer Series in Operations Research, 2005

[18] S. G. Dastidar, R. Nagi: *Batch Splitting in an Assembly Scheduling Environment*, International Journal of Production Economics 105, 2007, pp. 372-384

[19] Subodha Kumar, T. P. Bagchi, C. Sriskandarajah: *Lot Streaming and Scheduling Heuristics for m-machine No-wait Flow shops*, Computers and Industrial Engineering 38, 2000, pp. 149-172

[20] V. Ganapathy, S. Marimuthu, S. G. Ponnambalam: *Tabu Search and Simulated Annealing Algorithms for Lot Streaming in Two-machine Flow-shop*, IEEE International Conference on Systems, Man and Cybernetics, 2004

[21] Vickson R., B. Alfredsson: *Two and Three Machine Flow Shop Scheduling Problems with Equal Sized Transfer Batches*, International Journal Production Research 30, 1992, pp. 1551-1574

[22] Vickson R. G.: *Optimal Lot Streaming for Multiple Products in a Two-Machine Flow Shop*, European Journal of Operational Research 85, 1993, pp. 556-575

[23] W. Zhang, C. Yin, J. Liu, R. J. Linn: *Multi-job Lot Streaming to Minimize the Mean Completion Time in m-Hybrid Flow Shops*, International Journal of Production Economics 96, 2005, pp. 189-200

The Concept to Measure and Compare Students Knowledge Level in Computer Science in Germany and in Hungary

Gábor Kiss

Department of Machine and System Engineering, Budapest Tech
Népszínház u. 8, H-1081 Budapest, Hungary
e-mail: kiss.gabor@bgk.bmf.hu

Abstract: While education is based on a national basic curriculum in Hungary, Germany's 16 regions each with it's own Ministry of Education determine curricula independently resulting in difficulties for school switching students. Standards of computer science education varying in different regions is discussed along with efforts to unify the curricula. Differences in CS education of the two countries are pointed out. In order to make the two education system comparable a web-based on-line questionnaire was prepared by the author. Students are to fill in from all regions and in all grades in both Germany and Hungary. The questionnaire and evaluation methods are discussed together with first results.

Keywords: Computer Science Education, German, Hungarian, Comparison Method

1 Introduction

Germany is a Federal Republic with 16 members. Each member has it's own Ministry of Education and an own School system with own curricula. Therefore differences in computer science education are significant in different regions [1]. Computer science is obligatory in Bavaria and in Saxony but not even selectable in Hessen and Schleswig-Holstein. In some region the curriculum consists of just the office packet, in Mecklenburg-Vorpommern students learn the newest methods in Cryptography too, in Bavaria word processors are taught from object oriented viewpoint in the 5^{th} grade. Except Lower Saxony the basic knowledge of informationtechnology (Informationstechnologische Grundlagen; ITG) is introduced in all schooltypes from grade 5-10: somewhere as an independent subject, somewhere as part of natural science generally taught in 1-2 hour per weeks. Efforts are made to make computer science obligatory in all regions teaching it along uniform standards.

2 Computer Science Education in Germany and in Hungary

2.1 Basic Knowledge of Informationtechnology (ITG)

The german subject „*Basic knowledge of informationtechnology*” doesn’t exist in Hungary. According to the National Basic Curriculum (NBC) of Hungary the use of computer science is to be demonstrated in the first four school grades since 2003 (e.g. search on the Internet, painting with computers, etc.). In Germany this begins in the Basic knowledge of informationtechnology from the 5th grade, but there are differences in the different regions. In most cases this is taught as part of natural science, but in Bavaria, Berlin and in Saxony Anhalt it is an independent subject. Table 1 shows the weekly CS grades taught in the various school types of the german regions.

There are three different schooltypes: lower secondary school (Hauptschule), intermediate secondary school (Realschule) and high school (Gymnasium). The counterpart of the german lower secondary and intermediate secondary school is the primary school in Hungary. In Germany after the lower secondary school students may decide to study in vocational school where they can absolve the final exam too (Table 1).

Table 1
Basic knowledge of informationtechnology in the German regions

Region	Lower sec. sch.						Intermediate sec.						High school					
Grade	5	6	7	8	9	10	5	6	7	8	9	10	5	6	7	8	9	10
B.-W.																		
Bavaria														56				
Berlin			28						28						28			
Brand.									80						80			
Bremen																		
Hamb.	24																	
Hessen	Min. 16						Min. 16											
M.W.P.																		
L. Sax.																		
N.R.W.			60						60						60			
Rh.-P.																		
Saarland							24						40					
Saxony							26						26					
S.-An.									30		28				56			
Schl.-H.																		
Thur.							84						84					

One can see the number of grades is region specific and it's on the teacher to decide how long time he/she takes to teach basic knowledge of informationtechnology integrated in natural science in some regions.

2.2 Computer Science in High School

In Germany computer science is taught in some form in all regions. Difference show up in being obligatory or not. Int he best part of regions there is a chance to learn it in higher level which means 4-5 hours per week while in basic level it is just 2-3 hours per week. This chance is at hand in the 11-13th grades. Some regions make studying CS obligatory for those who want absolve a final exam in 10th grade. Final exam from computer science can be taken in all regions but in Baden-Württemberg and Saxony-Anhalt the exam is just verbal.

In Hungary CS is just selectable in the 11-12 grades. On basic level it is taught in 2 hour per week, on a higher level in 3 hour per week (Table 2). A final exam can be taken as in Germany [2].

Table 2

Computer science in high school

Grade	**10**	**11**	**12**	**13**
Baden-Württemberg	2 hours	2 hours		
Bavaria	2 hours	3 hours		
Berlin	OS, 3 h.	Basic, 3 h. / higher, 5 h.		
Brandenburg	OS, 2-3 h.	Basic, 3 h. / higher, 5 h.		
Bremen	OS, 3 h	Basic, 3 h. / higher, 5 h.		
Hamburg	OS, 2 h.	Basic, 3 h. / higher, 5 h.		
Hessen	Basic, 3 h.	Basic, 3 h. / higher, 5 h.		
Mecklenburg Western Pomenaria	2 h.	Obl. 2 h. / higher 4 h		
Lower Saxony		3 hours		
Nord RhineWestphalen		Basic, 3 h.	Bas., 3 h./ higher, 5 h.	
Rhineland-Palatinate		Bacis, 3 hours / higher, 5 hours		
Saarland	2 h.	Basic, 3 h. / higher, 5 h.		
Saxony	2 hours			
Saxony-Anhalt	OS, 2 hours			
Schleswig-Holstein		2-3 hours	Bacis, 2-3 hours	
Thuringen	2 h.	Basic, 3 h. / higher, 6 h.		
Hungary	2 h.	Basic, 2 h. / higher, 3 h.		

3 The Curriculum of Computer Science in the High Schools of Germany and Hungary

The Computer Science curriculum in the high schools of Hungary consists of:

- *Word processing*
- *Spreadsheet calculation*
- *Presentation*
- *Algorithm and programming*
- *Database management*

First let's see the differences between the high school curricula of the two countries.

In Hungary generally the Microsoft Office packet is taught, while in Germany the Open Office and other freeware softwares. The reason is of financial kind: the goal is to spare the price of expensive softwares which otherwise should be bought by the parents.

3.1 Word Processing

Table 3 shows when word processing is entering the curriculum in the German regions and in Hungary, respectively. In some regions this isn't part of the Computer Science curriculum, because it is taught within the literature subject.

Table 3

Word processing

Region	**Grade**								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg		■		■					
Bavaria		■							
Berlin			■						
Brandenburg			■						
Bremen	■	■							
Hamburg				■					
Hessen	■								
Meckl. Western Pom.					■				
Lower Saxony							■	■	
Nord Rhine Westph.					■	■			
Rhineland-Palatinate									
Saarland	■								
Saxony			■						
Saxony-Anhalt			■	■					
Schleswig-Holstein									
Thuringen	■								
Hungary		■	■	■	■				

It can be seen that teaching this subject draws 4 years in Hungary while only 1-2 years in Germany. Computer Science education begins somewhat later (in the 11-12 grade) in Lower Saxony but in this two years a lot of other knowledge is taught too.

3.2 Spreadsheet Calculation

Spreadsheet calculation is missing from the curriculum of Computer Science in 5 regions (Table 4) while word processing is missing in just 2 regions. The reason is similar to those above: spreadsheet calculation is integrated in the Math subject. This situation has advantages and disadvantages as well.

The advantage is that in Computer Science education it leaves more time to teach other subject matters. The disadvantage is that most teachers have no degree in Computer Science.

Table 4

Spreadsheet calculation

Region	**Grade**								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg									
Bavaria									
Berlin									
Brandenburg									
Bremen									
Hamburg									
Hessen									
Meckl. Western Pomenaria									
Lower Saxony									
Nord RhineWestphalen									
Rhineland-Palatinate									
Saarland									
Saxony									
Saxony-Anhalt									
Schleswig-Holstein									
Thuringen									
Hungary									

3.3 Presentation

Presentation classes are missing in regions (Table 5) where word processing or the spreadsheet calculation are not. The reason is not as clear as in the earlier case. In some regions it is missing completely from the education while sometimes we find it as part of the art subject. If it is taught at all, then generally much earlier than in Hungary.

Table 5

Presentation

Region	Grade								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg									
Bavaria									
Berlin									
Brandenburg									
Bremen									
Hamburg									
Hessen									
Mecklenburg Western Pomenaria									
Lower Saxony									
Nord RhineWestphalen									
Rhineland-Palatinate									
Saarland									
Saxony									
Saxony-Anhalt									
Schleswig-Holstein									
Thuringen									
Hungary									

3.4 Algorithm and Programming

Basic algorithms or rather programming appears in Computer Science sooner in Hungary (Table 6). But then as opposite to the used methods in Hungary, teaching the subject is approached from object oriented aspects in most of the regions in Germany. In some provinces JAVA is taught too, while in other regions algorithms are taught together with DELPHI. In Bavaria even word processing is taught from object oriented aspect in grade 6. In provinces where programming is taught more than two years, students learn more than in Hungary, because recursion, list and tree data structure are part of curriculum, while in Hungary they are just selectable. Another advantage of the German CS education is group work.The students in high school learn to work together: 2-3 persons work on bigger projects learning the advantage and disadvantage of project work too, what they will utilize in their later work in the area of Computer Science.

The LEGO Mindstorm kit which is very useful in learning programming is used from the lower level (grade 5-6) to university in almost all of the provinces [3].

Table 6

Algorithm and programming

Region	Grade								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg									

Bavaria									
Berlin									
Brandenburg									
Bremen									
Hamburg									
Hessen									
Meckl. Western Pom.									
Lower Saxony									
Nord RhineWestphalen									
Rhineland-Palatinate									
Saarland									
Saxony									
Saxony-Anhalt									
Schleswig-Holstein									
Thuringen									
Hungary									

3.5 Database Management

Database management is a subject taught in most regions for two years while in Hungary just one year (Table 7). This means that not just database management, but also SQL commands are taught. This part of the Computer Science curriculum begins in the 9th grade like in Hungary.

Table 7

Database management

Region	**Grade**								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg									
Bavaria									
Berlin									
Brandenburg									
Bremen									
Hamburg									
Hessen									
Meckl. Western Pom.									
Lower Saxony									
Nord Rhine Westph.									
Rhineland-Palatinate									
Saarland									
Saxony									
Saxony-Anhalt									
Schleswig-Holstein									
Thuringen									
Hungary									

3.6 Other Fields of Computer Science not Taught in Hungary

Two fields are to be mentioned here which are taught only in the German high schools though not in all regions.

3.6.1 Formal Languages and Automats

Teaching a formal language and automats or rather the Turing machine is a subject typically part of higher education in Hungary but in some regions of Germany this areas are parts of the Computer Science curriculum in high school (Table 8). Because of the difficulty of these subjects they appear first in the 11th grade or after. The knowledge of these subjects plays an important role later in Computer Science or in Mechatronics [4].

Table 8

Formal languages and automats

Region	**Grade**								
	5	6	7	8	9	10	11	12	13
Baden-Württemberg									
Bavaria									
Berlin									
Brandenburg									
Bremen									
Hamburg									
Hessen									
Mecklenburg Western Pom.									
Lower Saxony									
Nord RhineWestphalen									
Rhineland-Palatinate									
Saarland									
Saxony									
Saxony-Anhalt									
Schleswig-Holstein									
Thuringen									
Hungary									

3.6.2 Cryptography and Data Protection

Cryptography and the data protection are taught in Hungary similarly to formal languages and automats at the universities. In Germany it is taught in 7 regions in the high school (Table 9). The topic is especially important today being useful to show the students how easy is to break short or easy passwords especially when using the Internet where safe passwords are a must. In the table below it can be seen in which regions are taught the newest methods of cryptography. The grades are certainly varying with the regions.

Table 9

Cryptography and data protection

<table>
<tr><th>Region</th><th colspan="9">Grade</th></tr>
<tr><td></td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td></tr>
<tr><td>Hamburg</td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td></td><td style="background:yellow"></td><td></td><td></td></tr>
<tr><td>Mecklenburg Western Pomenaria</td><td></td><td></td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td style="background:yellow"></td><td></td></tr>
<tr><td>Lower Saxony</td><td></td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td style="background:yellow"></td><td></td><td></td></tr>
<tr><td>Rhineland-Palatinate</td><td></td><td></td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td style="background:yellow"></td><td style="background:yellow"></td></tr>
<tr><td>Saarland</td><td></td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td></td><td style="background:yellow"></td><td></td></tr>
<tr><td>Saxony-Anhalt</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td style="background:yellow"></td><td></td></tr>
<tr><td>Thuringen</td><td></td><td></td><td></td><td style="background:yellow"></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

4 The Basic Idea and Standpoint for Making a Comparison

The basic idea is to prepare a web-based on-line questionnaire with several test questions for students from all regions and all grades in Germany and in Hungary [5]. The answers will help to show how the two education systems differ from each other. In order to reach the best results the author used unified questions of computer science based on the national/regional curricula of Hungary and Germany.

Why was the web-based method chosen? This method was superior because in paper form the students would have got the questions in the same order and they could have answered them just in lesson time. The web-based format meant they could face the questions in a different order so they cannot help each other. A further benefit of a web-based questionnaire is that students can complete the test at home too, not only in school where time is limited. Moreover it is more convenient for the teachers and lecturers because the test does not shorten the duration of the lesson. Preparing the test in a one-hour class would probably not give enough time because the large amount of questions. This is due to the unification of the two countries' different curricula of computer science. In order to find the best results the appropriate questions have to be tailor-made for every theme and for every grade. The questions are composed to suit the grade and the level of the different education systems. For instance, because of these differences, German students in grade 6 face the same questions as Hungarian students in grade 8.

4.1 The Test

First of all students have to give details about their actual grade and other qualifying data. (Figure 1)

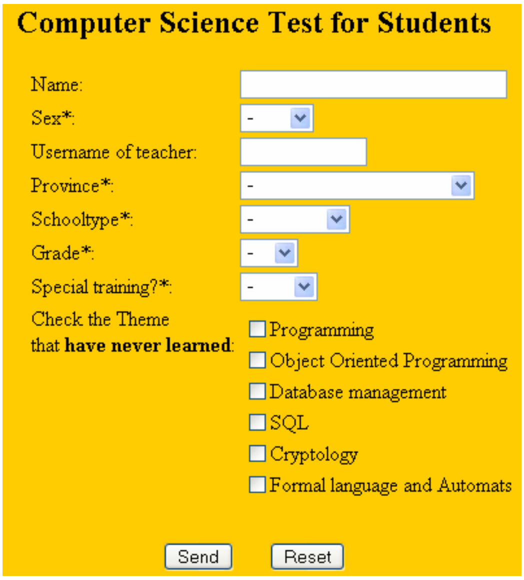


Figure 1
Students details

The respondent's name is not required, which provides anonymity, however gender is important so that it is known how many girls and boys have completed the test. In cases where a student gives the username of his/her teacher, the teacher can see how the students performed on the test and can control the students' work. A further requirement is to choose the province, this data helps us to find out the underlying perspective of the study, namely, to compare the regions. The following question asks for the type of school, potential answers are secondary school, high school or university. The grade is important, because this will decide what kind of question sheet the student will get that suits the level of his or her grade. The same questions are put in a different order so that the students writing the same test in the same time and in the same room aren't able to help each other.

The numbers of questions in grades are:

1 5th Grade: 11 questions

2 6th Grade: 26 questions

3 7th Grade: 70 questions

4 8th Grade: 113 questions

5 9th Grade: 136 questions

6 10th Grade: 137 questions

7 11th Grade: 151 questions

The first impression when considering the numbers of questions in the higher levels is that there may be too many but the average solving time in the following table (Table 10) shows this is fear is not realised.

Table 10
The average solving time

Grade	Time (min)
5	12
6	14
7	36
8	49
9	72
10	59
11	67
12	57

Special training means that the student has achieved a higher level in computer science. The themes covered in the test are as follows:

1 Basic computer science
2 Office packages
3 Programming
4 Object oriented programming
5 Database management
6 SQL
7 Cryptography
8 Formal languages and automats

Students can tick some of the themes (except for basic computer science and office packages, because these themes appear in both countries in every region), which they have never learned. In this case the system does not ask questions in connection with those topics but registers and saves them as the answer, ”I've never come across that”. With this system in place students get fewer questions and the examiner receives lots of answers quickly. After completing the first administration page students can begin the test.

Every test question has 6 possible answers; one of them is correct, 3 of them are false, the 5th possible choice is: "I've never come across that", the 6th one is: "I've forgotten it". The last two potential answers "I've never come across that" and "I've forgotten it" can provide the author with crucial information as to whether the student has learned certain topics (of the national / regional curriculum) and whether he or she remembers them.

Every question has two time limits in seconds. The first one is the minimum time needed to read, understand and answer the question, the second limit is the maximum answering time. The software saves the total time used by the student, furthermore, time limits are not seen or known by the student.

During the evaluation process various statistical methods are used (F-test, t-test, statistical analysis of the post-test, only randomized experimental design, etc.). With the help of these tools the examiner can show the potential real differences between the two countries from grade to grade in the analysed topic. Furthermore, using the same answers, this test will provide the opportunity to compare the computer science education in various regions of Germany as well.

4.2 What does the Test Provide for Teachers?

Teachers can register on the following site (Figure 2). They have to give several details at registration. The first notable difference is the username with which they can log on. The system is protected by a registration code so as to ensure that only the teacher which filled in this form can use the system.

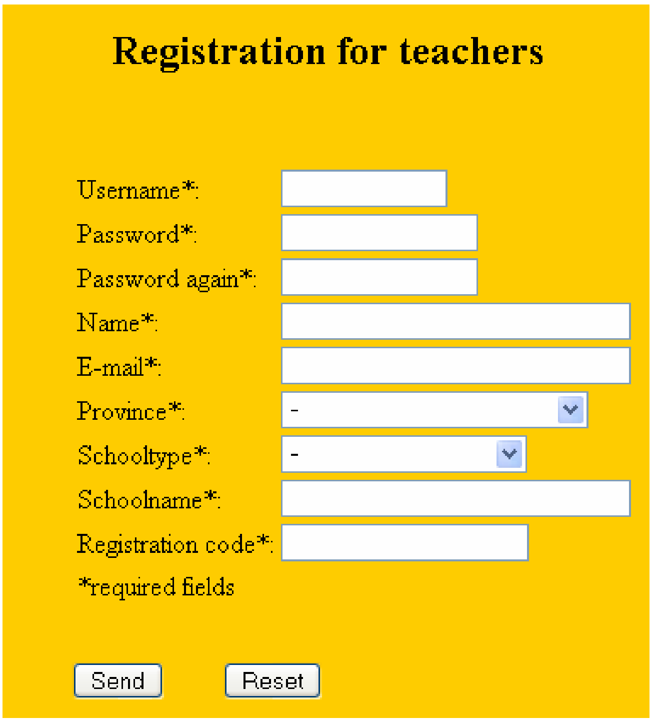


Figure 2
Registration for teachers

If the students give the username of their teacher, the teacher can see their results and how they performed in the test. Some reports are really useful in assisting the work of the teacher. The registered teachers can see:

1 The distribution of students in provinces
2 The distribution of students in various types of schools
3 The distribution of students in a Class in Secondary school
4 The distribution of students in a Class in High school
5 The distribution of students in a Class in University
6 The average time taken to complete the test in Secondary school
7 The average time taken to complete the test in High school
8 The average time taken to complete the test in Universities
9 The distribution of their students in Classes
10 The list of their students in Classes
11 The results of their students (including the "I've never come across that" answers, too)
12 The results of their students (including just the answered questions)
13 The summarized results of their students by subject matters (including just the answered questions)

Summary

Comparison produces evidence that in the significant part of Germany's regions appears computer science with more hours per weeks than in Hungarian education. The German schools have more financial means to buy and use newest hardware and software. Computer laboratories are well equipped and numerous. Sofware used comes generally from the public domain. Characteristically computers are found and connected to the internet in every family.

In Hungary the SULINET program helped to narrow the initial gap but there is still significant difference in the everyday use of computers in the two countries. German students use efficiently the advantages given by computer devices after finishing their schools while most hungarian students still have problems using word processors after their final exam.

After comparing Computer Science education in the two countries we can conclude that students learn CS longer in most regions of Germany than in Hungary and there are some topics which are not part of the curriculum in Hungary at all. Analyzing the data we found that though in some regions word processing and spreadsheet calculation isn't part of the Computer Science education, but they are taught in the frame of literature or math subjects leaving more time to learn other important areas in Computer Science.

Up to the present 1470 students have completed the test in Hungary. Most of them are from High schools. This suggests High School teachers used the test to validate their work.

Next, test questions will be translated to German so testing can start soon in Germany too. The Hungarian version is running on the computer of the author at Budapest Tech, the German counterpart will run in the University of Paderborn on the server of the Department of Computer Science.

References

[1] Gábor Kiss: Computer Science Education in Germany, 7th International Conference on Applied Informatics, Eger, 2007, pp. 45-55

[2] Gábor Kiss: Differences between the Hungarian and the German Computer Science Education, International Conference on Agricultural Economics, Debrecen, 2007, pp. 147-152

[3] Kiss Gábor: A LEGO Mindstorm németországi alkalmazása az informatikaoktatásban (6 old.), Matematika, fizika, számítástechnika főiskolai oktatók XXX. konferenciája, Pécs, 2006

[4] Kiss Gábor: A német és a magyar informatikaoktatás tematikájának összevetése a mechatronika oktatás tükrében (3 old.), 7. Nemzetközi Mechatronika Szimpózium, Budapest, 2007

[5] http://nero.banki.hu