PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 95 APRIL 2011 51–62

Several Aspects of Machine-Driven Phrasing in Text-to-Speech Systems

Jan Romportl, Jindřich Matoušek

Department of Cybernetics, Faculty of Applied Sciences, University of West Bohemia

Abstract

The article discusses differences between a priori and a posteriori phrasing and their importance in the task of automatic prosodic phrasing in text-to-speech systems. On several examples it illustrates shortcomings of common evaluation of a priori phrasing performance using a posteriori phrasing of referential corpus data. The paper also proposes and evaluates a method for a priori phrasing based on template matching of quasi-syntactical representations of sentences.

1. Introduction

A very important prosody processing task in text-to-speech (TTS) systems is proper suprasegmental symbolic description of input sentences. Such a symbolic description can be called *prosodic structure* of a sentence. Knowledge of a prosodic structure of a synthesised sentence is vital for both explicit and implicit prosody generation techniques (by "explicit" we mean those techniques which explicitly produce surface prosodic features such as F0 or intensity contours, and by "implicit" we mean techniques where suprasegmental surface features emerge from concatenated segmental features, which is the case, for example, in unit selection TTS systems without signal modifications or often in HMM-based systems).

Our theoretical framework of prosody description (Romportl and Matoušek, 2005) understands prosodic structure in terms of relations among prosodic words, prosodic phrases, prosodic clauses, prosodemes and semantic accents. Especially prosodic phrases play an important role in naturalness of synthesised speech (Romportl, 2010a), and therefore prosodic phrase boundary estimation based purely on textual represen-

Corresponding author: rompi@kky.zcu.cz
Cite as: Jan Romportl, Jindřich Matoušek. Several Aspects of Machine-Driven Phrasing in Text-to-Speech Systems. The Prague Bulletin of Mathematical Linguistics No. 95, 2011, pp. 51–62.
doi: 10.2478/v10108-011-0004-3.

tation of a synthesised sentence must be performed without major errors. It is one of the goals of this paper to propose and test a new algorithm which is able to designate prosodic phrase and clause boundaries in input TTS sentences so that the resulting phrasing is as much natural as possible (the question of prosodemes and semantic accents is left aside here).

Another goal, perhaps even more important, is to show that commonly used straightforward approaches to phrasing successfulness evaluation (i.e. comparison of automatically generated phrasing with referential testing data from a manually annotated corpus) are actually not very informative or fair because they ignore an essential fact about the nature of the prosodic phrasing problem.

2. Prosodic phrases

A prosodic clause is a continuous portion of speech between two pauses. It can comprise several prosodic phrases, and therefore prosodic phrases are often delimited by other prosodic features than a pause (e.g. intonation, segmental duration, etc.), thus their boundaries usually do not have special textual correlates such as punctuation marks.

A spoken utterance can usually be objectively segmented into prosodic phrases (Romportl, 2010a) because it already comprises relevant acoustic features actually produced by a particular speaker. However, in most cases it is not the only possible phrasing given the textual form of the utterance — the speaker could utter the text with different phrasing and it is also quite likely that if he utters the text once more, its phrasing will be different. This means that *a posteriori* phrasing of an utterance — i.e. the phrasing of an utterance already acoustically realised — is uniquely given, being a complex phenomenon determined by speaker's and listener's dispositions as well as by structural dispositions of the utterance itself. Both acoustic and syntactical features are important in the task of automatic *a posteriori* phrasing (Romportl, 2010b).

On the other hand, *a priori* phrasing of an utterance (or rather a sentence) is a process of purely *text-based* selection of one adequate phrasing from more potential variants which are allowed by the syntactical structure of the utterance. As a result of this, it is not correct to say that one particular *a priori* phrasing is correct whereas others are not: the sentence itself does not have enough causal potential to determine one particular phrasing

In the task of TTS synthesis we want to estimate the *a priori* phrasing of an input sentence, while this phrasing is then acoustically realised by the synthesis process itself. The question is, how to recognise whether the estimated phrasing is adequate for the given sentence or not. The immediate idea might be that we have a speech corpus of referential utterances with annotated phrases and we train and test the estimator using this corpus. However, this brings a serious problem: the *a priori* phrasing estimator is tested by the *a posteriori* phrasing annotations.

We can illustrate the situation by the following example. Lets suppose the speech corpus includes these two Czech utterances with annotated phrase boundaries (designated by "/"):

1a) Z mohutného kopce porostlého nízkými keři / se vine pěšina / do blízkého městečka.
(From mighty-Gen hill-Gen overgrown-Gen (by) low-Ins bushes-Ins / Refl winds footpath-Nom / to near-Gen town-Gen(-Diminutive).)

2a) Do sešlého hradu / zbořeného dlouhými věky / se vkrádá temnota / ze starého podzemí.
(To shabby-Gen castle-Gen / destroyed-Gen (by) long-Ins ages-Ins / Refl creeps in darkness-Nom / from old-Gen dungeons-Gen.

These two utterances have exactly the same syntactic structures, lexical words at the same positions bear identical morphological and syntactical categories (parts of speech, grammatical cases, syntactical functions), prosodic words at the same positions contain the same number of syllables, and still these two utterances have different *a posteriori* prosodic structures because 1a has three prosodic phrases whereas 2a has four. This means that there is not enough information in the textual form of an utterance to determine unambiguously its *a posteriori* phrasing. As a result, if a text-based phrasing estimator of a TTS system produces the *a priori* phrasing 2b, we really cannot say it is an error because there is no information available for the estimator to let it know that the "correct" phrasing form is 2a, not 1a.

2b) Do sešlého hradu zbořeného dlouhými roky / se vkrádá temnota / ze starého podzemí.

On the other hand, the *a priori* phrasing 2c can be considered as erroneous because it is in contradiction with the syntactic structure of the sentence (a tight syntactic relation between a noun "hradu/castle-Gen" and its attribute "sešlého/shabby-Gen" is disrupted by a phrase boundary):

2c) Do sešlého / hradu zbořeného dlouhými roky / se vkrádá temnota / ze starého podzemí.

Therefore, it is reasonable to impose requirements on a text-based *a priori* phrasing estimator so as the estimator avoids errors like 2c as much as possible while differences similar to the one between 1a and 2a (or 2a and 2b) do not matter.

It might seem that we are somehow trying to say what has been known for long: the placement of prosodic boundaries helps the listener parsing the sentence, hence they are highly correlated with syntactic boundaries, but to a large degree optional; however, at some places they would be rather confusing and this is considered wrong.

Such a statement is definitely true and well known, but this is not what we are aiming at here — instead, we are explicitly articulating the differences between *a posteriori* and *a priori* phrasing due to their influence on machine-learning and classification performance evaluation in the process of automatic *a priori* phrasing estimation.

A common machine-learning scheme would unnecessarily penalise the estimator's response 2b because the referential variant 2a is in the training/testing database. It would force the estimator to try to find some cues in the text of the sentence indicating that 2a is "correct" whereas 2b is not. But there are no such cues inherently present in the text — these cues might be found in speaker's dispositions, not in the sentence itself. And as the estimator does not have any access to what the speaker's dispositions can be, it will either continue to make these "false errors" (formally decreasing its nominal performance), or it will discover "false cues", which leads to overtraining.

A solution can be that there are all possible (or at least more) *a posteriori* phrasing variants of every sentence present in the corpus-based testing/held-out data for machine learning, allowing the machine learning algorithm to decide whether its output for a given feature-described sentence is correct (i.e. is one of the phrasing variants) or not. However, this is infeasible in normal situations when only one variant of each sentence is available, such as common speech corpora for TTS voices. Another solution, presented further in this paper, is more radical: it does not choose the approach of classical machine learning techniques or structurally driven construction of new prosodic structures for processed textual sentences; instead of this, it considers the whole TTS corpus (which is usually large) as the universe of all possible prosodic structures, and by a very simple algorithm it finds the most similar sentence to the processed one and reuses its phrasing.

By a machine learning technique we mean a process of automatic optimisation (usually iterative) of internal parameters of a classifier on the basis of training data (and possibly held-out data). The simple method proposed in this paper is a classifier, but its internal parameters are not optimised in any way, therefore no machine-learning technique is used.

Structurally driven construction of new prosodic structures refers to a process of building whole prosodic structures from smaller parts on the basis of various structural rules, such as those in grammar-based deterministic or stochastic parsing techniques. The proposed method does not use this approach as well — instead it takes prosodic structures already created in the corpus and does not consider any structural rules standing behind them.

3. Automatic a priori phrasing

As it was just mentioned, the idea behind our approach is following: if we have a suitable referential speech corpus (such as the one used as the source corpus for a given voice in a unit selection TTS system), we can understand all its utterances as templates and the phrasing estimation process is conceived as template matching

— an input TTS sentence receives the *a priori* prosodic structure (phrasing) which *a posteriori* belongs to the matched template sentence. This ensures that the selected assumed phrasing fits well with the syntactic structure of the given input sentence, and errors such as 2c are far less likely to occur than with other methods artificially constructing new phrasings which often might have not occurred in the corpus at all — e.g. HMMs, prosodic parsing, neural networks, etc., cf. (Romportl, 2010b; van Santen et al., 2008; Dutoit, 1997; Fitzpatrick and Bachenko, 1989).

3.1. Speech corpus

The template matching algorithm utilises a large collection of recorded utterances, which is usually not a problem in unit selection TTS systems where such data are necessary for speech segment database creation as well. It is even advisable to use the same corpus for both these tasks, because the unit selection algorithm will then process *a priori* phrasings originating from the same data as the concatenated segments.

For our experiments, we have used the corpus of 9,596 Czech declarative sentences recorded by a male speaker and used in the Czech TTS system ARTIC (Matoušek and Romportl, 2007). Prosodic phrases were automatically annotated in the whole corpus by a method based on artificial neural networks (Romportl, 2010b) trained on 250 manually inter-subjectively annotated sentences (Romportl, 2010a).

3.2. Syntactic features

A syntactic structure is a very important aspect in determining prosodic phrases. However, rather than the whole non-linear structure, it is more important for prosodic phrase boundaries to consider local syntactic relations between adjacent words, such as subject–attribute or predicate–object syntagmas (Palková, 1974). We proposed two sets of features for lexical word representation which proved suitable for automatic *a posteriori* phrasing (Romportl, 2010b):

- **Analytical functors** (`AFUN`). Analytical functors represent *syntactical functions* of lexical words. The inventory of functors we used originates from Prague Dependency Treebank 2.0. It has been slightly modified and it is listed in Table 1. Our whole corpus was syntactically parsed using the `TectoMT` application (Žabokrtský et al., 2008) with the McDonald's dependency parser yielding accuracy 85 % for Czech text. The parser assigns each lexical word an analytical functor, and since `AFUN` is a categorical feature, this functor is coded as a vector of 0's with a 1 in the dimension corresponding to the functor's order in Table 1 (e.g. *Obj* is coded as $[0, 0, 1, 0, 0, \ldots]$).
- **Apriori estimation of analytical functors** (`AFUNap`). Each lexical word form can be parameterised by a vector of *a priori* probabilities of analytical functions that this word form can appear in (e.g. $p(w = \mathrm{Pred}) = 0$, $p(w = \mathrm{Sb}) = 0.2$, $p(w = \mathrm{Obj}) = 0.5, \ldots$). The advantage of such a parameterisation is that no syntactical

abbrev.	description
Pred	Predicate
Sb	Subject
Obj	Object
Adv	Adverbial
Atv	Complement
Atr	Attribute
Pnom	Nominal predicate
AuxV	Auxiliary verb "be"
Coord	Coordination
Apos	Apposition
AuxTR	Reflexive tantum
AuxP	Preposition
AuxC	Conjunction
AuxOZ	Redundant or emotional item
AuxY	Adverbs and particles

Table 1. List of analytical functors.

parsing is needed — only a lexicon with word forms and probabilities which were derived from the data of Prague Dependency Treebank 2.0 in our case.

3.3. Template matching algorithm

1. Every sentence in the corpus is parameterised using the analytical functors of lexical words:
   (a) Each lexical word w_i of the sentence

   $$S_k : w_1, w_2, \ldots, w_p$$

   with p words is represented by a 15-dimensional feature vector $\mathbf{a}_i$ of `AFUN` or `AFUNap` (the choice between `AFUN` and `AFUNap` depends on the experiment; see the next section).
   (b) The parameterisation of the whole sentence S_k is given by the vector

   $$\mathbf{s}_k = \left[\mathbf{a}_1^T, \mathbf{a}_2^T, \ldots, \mathbf{a}_p^T\right]^T. \quad (1)$$

   The vector $\mathbf{s}_k$ is thus an element of a 15p-dimensional space. The whole corpus creates as many spaces as there are different sentence lengths.
2. A sentence to be synthesised (further denoted as *input sentence*) has l lexical word tokens.

3. If $l < 5$, then the input sentence consists of a single prosodic phrase (and prosodic clause as well) and the algorithm ends. This is justified by the fact that there are only 31 sentences shorter than 5 words in the corpus, hence their phrasing variability can be omitted (there are only 6 phrasing variants for them anyway).
4. If $l > 9$, then prosodic clauses (and thus pauses) are determined in the input sentence in such a manner that each prosodic clause is a continuous part of the sentence between two adjacent punctuation marks (commas, hyphens, brackets, etc.). If a prosodic clause boundary is to be placed on a comma, the clause must be at least 4 lexical words long, otherwise the comma is inside the phrase and does not end it. The condition of 9 words is based on the fact that no prosodic phrase in the corpus was longer than 9 words.
5. If $l \leq 9$, then the input sentence is considered to be a single prosodic clause for now.
6. The whole input sentence is processed clause by clause. Each prosodic clause is further processed separately as if it were a standalone sentence.
7. The actually processed clause is l_C words long and is parameterised by a $15l_C$-dimensional vector $\mathbf{x}$ determined analogically to the steps (1a) and (1b) with the only difference that now it is a clause, not the whole sentence.
8. The sentence S_{k*} (the matched template) is found such that k^* is determined as

$$k^* = \arg\min_{k \in \mathcal{S}_{lC}} \|\mathbf{s}_k - \mathbf{x}\|, \tag{2}$$

   where $\mathcal{S}_{lC}$ is a set of indexes of those sentences from the corpus whose length equals to l_C.
9. Prosodic phrase boundaries in the actually processed prosodic clause are placed exactly as they are in S_{k*}.
10. If any of the phrase boundaries placed in (9) coincides with a punctuation mark not tagged as a clause boundary in (4), then this punctuation mark is newly considered to be a clause boundary (i.e. the actually processed clause can further be split into smaller clauses).
11. After processing all the clauses determined in (4) and (5), the phrasing of the whole input sentence is finished: the prosodic clause placement (and thus pause placement) is given by (4) and (10), the prosodic phrase placement inside of these clauses is given by (9).

Even though the syntactic and prosodic structures have a many-to-many mapping in the universe given by the corpus, the rule (8) ensures that only one prosodic structure is selected for the given input syntactic structure — the one belonging to the real corpus utterance with the closest syntactic structure to the input sentence.

The rules (3), (4) and (5) are clearly specific for this particular corpus, determined by the phrase length distribution in it. Sentences shorter that 5 words are omitted due to their low phrasing variability in the Czech language, sentences longer than 9 words are processed heuristically because there was no phrase longer than 9 words in the

corpus and we need a reasonable upper limit for the feature vector dimension. These values are presented here because they are probably more generally valid within the Czech language or at least the particular speaker, but there is technically no problem changing them in dependence on a corpus actually used.

4. Experimental evaluation

The algorithm described in the previous section is able to estimate *a priori* phrasing of any textual sentence so that this phrasing is consistent with the speaking style of the speaker who recorded the corpus. The key role is played by the formula 2 which expresses our hypothesis that the best *a priori* phrasing estimation of a so far unobserved sentence (or its part) is the *a posteriori* phrasing of an observed (in the corpus) sentence of the same length which is (quasi-)syntactically most similar to the unobserved sentence. This hypothesis can be justified by the following experiment using the collection of 4,824 sentences from the corpus whose length was 5–9 lexical words (this experiment excludes sentences longer than 9 words because the step (4) of the described algorithm is really just a heuristic rule technically allowing processing of longer sentences):

- The experiment is performed with the same number of iterations as the number of sentences in the collection (i.e. 4,824).
- In each iteration a tested sentence S_t is removed from the collection. From the rest of the collection, a sentence S_{k*} is selected according to the formula 2 for the sentence S_t. This is iteratively performed for all S_t from the collection.
- If in the particular iteration the referential phrasing of S_t is identical to the phrasing of S_{k*}, the counter of absolute agreement is increased by one.
- If the referential phrasing of S_t is not identical to the phrasing of S_{k*}, the difference is quantified as $\varepsilon = \|\mathbf{f}_t - \mathbf{f}_{k*}\|$. As S_t and S_{k*} are sentences p words long, the p-dimensional vector $\mathbf{f}_t$ represents the phrasing of S_t so that there are 1's in the vector at the positions corresponding to the indexes of the words at the phrase boundaries, and 0's elsewhere (e.g. for S_t: "word1 word2 / word3" $\mathbf{f}_t = [0, 1, 0]^T$. The vector $\mathbf{f}_{k*}$ analogically represents the phrasing of S_{k*}.

The experiment was performed separately for both `AFUN` and `AFUNap` parameterisations and the results are summarised in Table 2. It is clear that `AFUN` "outperforms" `AFUNap` in terms of the absolute agreement: in 26.1 % of the tested cases the *a priori* phrasing of the tested sentence was estimated identically to the referential *a posteriori* phrasing (the tested cases are whole sentences, not words). It might seem that this rate of absolute agreement is not high enough — but such a judgement would be a misinterpretation: we must bear in mind that we still test the *a priori* phrasing against the *a posteriori* phrasing. This value thus must not be simply interpreted as the *accuracy* in terms of a classification performance evaluation. It does not tell us much about the classifier we used (which is, anyway, trivial) — instead it tells us something more important about the data: only 26.1 % of the sentences in the corpus have their

	absolute agreement	$\mathrm{E}\{\varepsilon\}$
AFUN	1259 (26.1 %)	1.4111
AFUNap	888 (18.4 %)	1.4511

Table 2. Results of the experimental evaluation.

prosodic structures fully determined by their AFUN (quasi-)syntactic representations (and their linear distances).

Even though there are differences between referential and estimated phrasings in the remaining 73.9 % sentences from the collection, we can still assert that in spite of being different, an estimated phrasing is always a phrasing of a real utterance with a very similar syntactic structure (this is an analytical assertion), and therefore most likely fitting to the tested sentence (this assertion, though, should be corroborated by formal listening tests).

Moreover, the average value of ε shows that in those cases where the estimated *a priori* phrasing was not identical to the referential *a posteriori* phrasing, the average differences lie only in shifting one phrase boundary in each sentence. This interpretation of the average value $\mathrm{E}\{\varepsilon\}$ is based on the fact that $1{,}4111 \approx \sqrt{2}$ and if the vectors $\mathbf{f}_t$ and $\mathbf{f}_{k*}$ differ only in the placement of one element with the value 1 (e.g. $\mathbf{f}_t = [1,0,0,1,0]^T$ and $\mathbf{f}_{k*} = [0,1,0,1,0]^T$), then $\|\mathbf{f}_t - \mathbf{f}_{k*}\| = \sqrt{2}$. Of course this could also mean that there were 2 phrase boundaries added or deleted in every sentence, but after manual inspection of 100 randomly chosen tested sentences we verified that the most frequent difference really is a boundary shift, and most importantly, that the estimated *a priori* phrasing was always adequate for the given sentence, even though one boundary was shifted against the referential *a posteriori* phrasing — i.e. there were no errors similar to the example 2c, except for the cases where the real speaker recorded such inappropriate phrasing to the corpus (however, having the assumption that "the corpus is always right", these cases should not be considered as erroneous here — after all, the system tries to duplicate the speaking style of the original speaker as much as possible).

If we recalculate the values of the sentence absolute agreement and ε so that we consider the numbers of words in the sentences (i.e. the length distribution of the evaluated sentences, measured in lexical words), we get approximately 80 % accuracy of phrase boundary placement on words (including insertion and deletion errors). This accuracy value is fairly comparable with reports on English phrasing; no similar results allowing direct comparison have been reported for Czech. However, in our opinion it is not vital to further increase the word-level accuracy at any cost because our approach should guarantee that all the estimated phrase structures are appropriate in spite of possible phrase boundary insertions or deletions.

From the comparison of `AFUN` and `AFUNap` it is clear that it is better to have a syntactic parser as a part of the TTS system. However, if this is not possible for some reason, complete syntactic parsing can be replaced by the `AFUNap` approximation to some extent.

5. Conclusions

Our main goal was not to create a sophisticated algorithm for prosodic phrasing; rather we wanted to evoke more discussions on justness of many complicated machine-learning methods for prosodic phrasing by showing that even a very simple algorithm can efficiently fulfil this task once the apparent difference between *a priori* and *a posteriori* phrasing is considered as really constitutive for the view on the classification performance evaluation. Many common methods struggle for achieving higher accuracy in phrase boundary placement, forgetting that this often is — with a little hyperbole — rather a phantom chase. The most important thing is to clarify what we want: is it natural phrasing of synthetic speech, or is it the ability of the estimator to blindly follow its training/testing data? We have just wanted to point out that the former can be achieved by a simple algorithm based on the understanding that the corpus is all we know about prosodic phrasing and that if a new sentence comes, its *a priori* phrasing is same as the *a posteriori* phrasing of some sentence from the corpus. In our case, we have deliberately abandoned attempts to measure the phrasing successfulness in terms of the classification accuracy — instead we rely on a hypothesis that reusing of the phrasing of a real utterance syntactically similar to the processed one delivers an appropriate phrasing as well. The next step is to corroborate this hypothesis by large-scale formal listening tests following the scheme already used in the inter-subjective *a posteriori* phrasing annotation process of our corpus (Romportl, 2010a).

The algorithm proved well in the evaluation experiments and it can be easily implemented in a real TTS system. Its main advantages lie in its straightforward structure and its ability to generate adequate phrasing in almost all cases. The analytical functors used for parameterisation of words and sentences seem to be suitable as well. Still there are various aspects remaining unexplored: it might be interesting to see whether some optimisation of the algorithm parameters can improve its performance in terms of the absolute agreement — these parameters comprise mainly weights of particular functors in the formula for minimal distance of the sentence parameterisations. Since syntactic parsing is employed for analytical functor estimation anyway, it might also be helpful to utilise mutual syntactic relations of words in addition to their analytical functors, which would lead to more complex comparison and distance measuring. And finally the most important issue: the sentence template matching, as it is performed now, does not take into account rhythmical structure on the level of prosodic words; therefore features such as number of syllables or their distribution shall be added.

Acknowledgements

Support for this work was provided by the Ministry of Education of the Czech Republic, project LC536, and by the Grant Agency of the Czech Republic, project GAČR 102/09/0989. The access to the MetaCentrum computing facilities was supported by the research intent MSM6383917201.

Bibliography

Dutoit, Thierry. *An Introduction to Text-to-Speech Synthesis*. Kluwer Academic Publishers, Dordrecht–Boston–London, 1997.

Fitzpatrick, Eileen and Joan Bachenko. Parsing for prosody: what a text-to-speech system needs from syntax. In *In Proceedings of the Annual AI Systems in Government Conference*, pages 188 –194, Mar. 1989. doi: 10.1109/AISIG.1989.47324.

Matoušek, Jindřich and Jan Romportl. Recording and annotation of speech corpus for Czech unit selection speech synthesis. In *Proceedings of TSD 2007*, Lecture Notes in Artificial Intelligence, vol. 4629, pages 326–333. Springer, Berlin–Heidelberg, 2007.

Palková, Zdena. *Rytmická výstavba prozaického textu (Rhythmical Potential of Prose)*. Academia, Praha, 1974.

Romportl, Jan. On the objectivity of prosodic phrases. *The Phonetician*, 96:7–19, 2010a.

Romportl, Jan. Automatic prosodic phrase annotation in a corpus for speech synthesis. In *Proceedings of Speech Prosody 2010*, Chicago, IL, USA, 2010b.

Romportl, Jan and Jindřich Matoušek. Formal prosodic structures and their application in NLP. In *Proceedings of TSD 2005*, Lecture Notes in Artificial Intelligence, vol. 3658, pages 371–378. Springer, Berlin–Heidelberg, 2005.

van Santen, Jan, Taniya Mishra, and Esther Klabbers. Prosodic processing. In Benesty, Jacob, M. Mohan Sondhi, and Yiteng Huang, editors, *Springer Handbook of Speech Processing*, chapter 23, pages 471–487. Springer, Berlin, 2008.

Žabokrtský, Zdeněk, Jan Ptáček, and Petr Pajas. TectoMT: Highly modular MT system with tectogrammatics used as transfer layer. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 167–170, Columbus, OH, USA, 2008.

Address for correspondence:
Jan Romportl
rompi@kky.zcu.cz
Department of Cybernetics
Faculty of Applied Sciences
University of West Bohemia
Univerzitní 8
306 14 Plzeň, Czech Republic