Orator – an intelligent system supporting stuttering therapy[1]

Halina Kwaśnicka, Błażej Żak
Institute of Applied Informatics
Wrocław University of Technology

Abstract
The paper presents the *Orator* computer system – it is a system for detection and therapy of stuttering. The system comes out to the need of stuttering people. It consists of three modules: stuttering detection, stuttering therapy and automatic control of therapy process. The main idea of the system is based on the useful device – DAF (Delayed Auditory Feedback). The method drawn from DAF is implemented and the computer program can be used in manual mode as a DAF device. The main advantage of the Orator system is the automatic mode of therapy, and it is free. The paper gives introduction to the stuttering problem, presents the developed modules and used methods. The two techniques have been used for stuttering detection: SOM network and simple fuzzy rules. The latest proves to be considerably better. The paper ends with some remarks about the system.

1. Introduction

Computers and software are becoming necessity for people working in various professions. One of the factors pushing the development forward is rapid evolution of Artificial Intelligence (AI) methods – a field of computer science that was always widely explored by medical systems.

In the paper we use AI techniques to develop a system that would be helpful in a therapy of stuttering patients. We chose neurological together with psychological stuttering theory as a theoretical base for therapeutic model. Neurological theory implied very popular delayed auditory feedback (DAF) method as the basic mean of therapy, whereas psychological theory gave reasons to develop an intelligent system that automatically controls the auditory feedback parameters with help of artificial intelligence.

The paper describes system design, basic theoretical background and solutions to three basic problems faced in developing the application:

- detecting stuttering in human voice,
- intelligently controlling therapy parameters,
- providing real-time delayed auditory feedback.

Two methods of stuttering detection were implemented. First, based on Kohonen's Self-Organizing-Maps, that unfortunately did not perform good enough to provide quality data for therapy control module, and second based on fuzzy logic theory, that proven to be successful. Both stuttering detection methods are preceded with heavy sound preprocessing that includes sound filtering, Fourier transform, Mel scale transformation, dynamic spectrum normalization and feature extraction. Large inspiration for design of stuttering detection module were taken from applications of artificial intelligence in economy, mainly the problem of discovering stock price formations, that proven to be in a way similar to problem of detecting stutters in voice.

Therapy controller module was implemented as a fuzzy logic controller with fuzzy rules developed from Perkins' stuttering therapy program described in [6]. Controller was made to be auto-adaptive and insensitive to occasional false classifications by stuttering detection

[1] The shorter version of the paper, concerning the proposed fuzzy control system in the *Orator* system has been accepted and published in the ICASC'2006 Conference Proceedings: Kwaśnicka H., Zak B.: *Fuzzy logic in stuttering therapy*. Artificial Intelligence and Soft Computing – ICASC 206. Rutkowski L., Tadeusiewicz R., Zadeh L.A., Zurada J. (Eds), Lecture Notes in Artificial Intelligence, v. 4029, str. 925-930. Springer, 2006.

module. Sound filters, that build the therapy provider module, are described as well as simple single-window patient user interface. Finally, overall system performance results, patient and speech therapists opinions are quoted. The paper ends with further development map and ideas how it, as a whole, can be improved.

The *Orator* – intelligent system supporting stuttering therapy was developed up to production release. What is worth mentioning, *Orator* is a real-time system, what implied many necessary optimizations and resource effective implementation.

The paper has a following structure. Next section gives theoretical background of stuttering. It describes symptoms of stuttering and fluency training exercises and techniques. The Delayed Auditory Feedback (DAF) method together with a echo-correction device is described – they are fundamental for the developed system. Section 3 describes the *Orator* system supporting stuttering detection and therapy. The architecture of *Orator* and more details used in the developed system are also presented. The results are presented in Section 4. Last section summarizes the paper.

2. Stuttering – symptoms and therapy

Stuttering can be defined as a kind of speech fluency disorder, where by 'speech fluency' we understand the natural speech flow [6]. However speech has the appearance of continuity, it is highly discontinuous. Speaking we make pauses between words and phrases, e.g., to take a breath or to think what to say. Such pauses are natural, although soundless, usually carry part of the meaning. Pauses may emphasize meaning of certain words we want to underline, provide semantic and syntactic boundaries, etc. Actually "fifty percent of spontaneous speech is less than three words long, interrupted by pauses. Only about 10 percent of spontaneous speech is composed of phrases that are 10 words or longer" [7].

2.1. Symptoms

Having in mind that stuttering is a speech fluency disorder, let us define what does 'fluent speech' mean. There are four factors that make speech considered to be fluent:

1. *Continuity* – it tells if the pauses are in syntactically and semantically right places of the speech stream.
2. *Rate,* expressed in syllables per second, tells how fast do we speak.
3. "*Rhythm* of speech is the sense of the flow of speech one gets from the stress, duration and timing of syllables. In English, stressed syllables, which are longer, louder, and higher-pitched than unstressed syllables, occur at more or less equal intervals interspersed with one or more unstressed syllables." [5].
4. *Effort* is the physical and mental power that one uses to produce a speech.

Stuttering is a complex multi-dimensional speech fluency disorder [19]. In fact it involves disorders in all four factors mentioned above. *Continuity* and *Rhythm* are obviously disturbed by unattended pauses and blocks. *Rate* is usually much slower, however persons who stutter usually have tendency of trying to speak "too fast" but unattended blocks just make them unable to do so. *Effort* is significantly larger than average, there is also usually an emotional effort, that is absent with nonstutterers.

The three types of disfluencies are recognized [5]:

1. *Repetitions* (of sound, syllable, word and phrase), it means that sound or syllable or word or whole phrase is repeated, e.g., I t-t-talk like this; I ta-ta-talk like this; I talk talk like this; I talk I talk like this.
2. *Sound prolongation*, i.e., sustain a vowel, fricative, glide or liquid sound for a markedly longer duration than what would be normal. For example: I aaaaate my lunch; I ssssaid so.

3. *Blocks* – it is inability to produce audible speech at the beginning of an utterance or word (tense pause) or in the middle of a word (broken word). The resulting articulated posture might be tonic, e.g., I _____ don't know.

The most common observed type of stutters is sound and syllable repetition. Obviously, there are other features of stuttering, like uncontrolled facial grimaces, verbal mannerisms, respiratory abnormalities, performance anxieties and others, but since they are hardly audible, they are not vital for this research.

There are many theories trying to explain the stuttering phenomenon:

Physiological theory says the primary reason is neurophysiological, and lies in improper coordination of respiratory muscles, and in phonetic and articulation apparatus. The psychological impact on patient is an effect of problems in socialization caused by speech disfluency.

Psychological theory states the opposite: stuttering is a psychosomatic disorder – the primary reasons are psychological, and stuttering is a secondary, derivative symptom. If we carry successful psychotherapy and patient will solve his emotional problems, the stuttering will go away.

Linguistic theory points the fact that 90% of stuttering cases began in the time when a child learns speaking (up to sixth year), and often stuttering disappears without any medical or psychological intervention. Hence they consider stuttering as just delayed development of speech fluency.

Neurological theory says that the problem lies in lack of synchronization, on a neural level, between speech production system (responsible for controlling speech muscles), and speech control system (correcting speech production system according to feedback it gets from auditory system). Problems of synchronization cause neural loops that we hear as repetition of sounds or syllables [8].

The stuttering phenomenon is explained in many different ways by many different theories, some even being contradictory to each other. It is worth mentioning that different approaches usually cover different real life cases, and since physiological theory might fit ideally to one patient, other case might be perfectly described with psychological theory. It is important that these theories bring vital knowledge to understanding stuttering.

2.2. Therapy

Stuttering therapy usually consists of some kind of psychotherapy or group therapy supported by linguistic training. The role of psychotherapy is to solve patient emotional problems either caused by stuttering (physiological theory) or causing stuttering (psychological theory). Linguistic training aims to show the patient techniques and exercises he can use to learn to speak fluently. Experienced speech therapists say that it is very important to merge those two kinds of therapy, otherwise the therapy is much more difficult and problem is very likely to recur. In our work we focus on one of the linguistic training exercises since psychotherapy seems to be a very vague area and probably impossible to support with software.

It is easy to find descriptions of many exercises of speech fluency, they help a stutterer to maintain a rate of speech by enforcing (even unnatural) rhythm of speech. As we mentioned earlier, most of those who stutter try to speak very fast "to say everything before a block occurs". Fluency training should change this strategy and teach them to speak slow, even much slower than other people, but do it fluently. Since the fluency is achieved they learn to speak faster and faster, finally reaching normal speech rate [5].

Speaking slowly can be achieved in many different ways, for instance:

- Speaking with metronome – patient says one syllable every tick of metronome
- Speaking like singing – patient unnaturally delay vowels "like if he was singing"

- Artificially prolonging syllables and inserting more pauses

Other very interesting technique making stutterers speak just slightly slower but more fluently is chorus speaking. The technique comes from observation that stutterers speaking in chorus stutter much less. The variation of this technique is called *Delayed Auditory Feedback* (*DAF*) and was first described in 1958 by B. Adamczyk [9]. *DAF* involves producing echo for a stuttering person – so he hears himself with a little delay. This echo-effect is very similar to speaking in chorus, but the chorus is just produced virtually, not by other person. The same effect can be achieved with reverberation effect.

Frequency Altered Feedback (*FAF*) is the other technique similar to *DAF*. It involves producing pitch shifted echo for the stutterer. Person using *FAF* hears himself speaking in higher or lower tone. Experiments have shown that the most effective is technique merging the two – *FAF* and *DAF*. There is still an open problem concerning the adjusting parameters of the *DAF* or *FAF* therapy like delay and amount of the frequency shift and echo volume – they seem to depend on both personal features of the patient and the state of the therapy. Generally in the beginning of the therapy we make the delay longer and with time – when the patient learns speak fluently – we decrease delay and volume, so at the end patient can speak without support of *DAF* or *FAF*.

A traditional echo-correction device was used as a basic model during building our computer therapeutic system. The one available at the clinic had only three knobs that could be used to adjust: volume, delay and intensity of reverb effect. A patient, talking to the microphone and wearing the headphones, hears himself with a little delay (adjusted with the knob) and reverberation effect (it must be carefully adjusted). This simple device, used by stuttering person, usually decrease the number of stutters immediately even by 70% [18]. With echo-correction device they speak slower but fluently, and the "speaking-slowly" effect is sometimes lasting for even few hours, making an impression that they got cured.

The main problem with *DAF* therapy is that it is touching just symptoms of stuttering, while leaving the reasons why people stutter behind. When such a temporarily cured person will face a stressful situation, in which he learned throughout the years to react with stuttering, *DAF* training will not help and the stuttering is likely to recur instantly. Hence *DAF* is usually used just as a kind of linguistic training, but very effective.

3. The *Orator* system

Our basic assumption was that the *Orator* system should be a PC software that can also act in the same way as a traditional *DAF* device. The important drawback of *DAF* training is that it makes the speech slower, but sometimes, when the delay is set too large, it makes it too slow than necessary. The *Orator* system will adaptively adjust the delay so it will reach minimum level for an individual person, hence making the speech slower but only to the level that is necessary to make it fluent. It is very important feature of the proposed system.

Orator contains sound filters necessary to carry on echo-therapy, i.e., echo effect, reverb effect and chorus effect. Unlike traditional *DAF* device, *Orator* can adjust all the therapy parameters automatically, and tune them adaptively to the patient. Those parameters are controlled adaptively in real time basing on information about training progress. The training progress is measured by the number of stutters in patient speech in a given time period.

The psychological theory and the observations reveal that many patients, when focused on their stuttering problem start stutter more. Training with *DAF* a patient himself must manually adjust therapy parameters what causes focusing on his stuttering and destruction of therapy effects (he asks the questions like: Do I stutter less? Should I decrease the echo delay now?).

When automatic control of therapy parameters is provided, patient can focus just on speaking or reading, and his attention can be drawn away from stuttering. This should highly

improve effects of therapeutic session. There is also a large group of patients that simply cannot judge how much do they stutter at the moment. The automatic control has also a psychological effect that can be achieved if a patient finishes therapeutic session without help of the system. If a patient starts to speak fluently, delay of echo is decreased and the volume is slowly faded out. At the end of session echo delay and volume are minimal or even zero. This means that we will use *DAF* to teach the patient speaking fluently without *DAF*, since we taking *DAF* away shortly after it does its job. Such effect has very serious psychological consequences – it shows the patient, that he can speak fluently without help of the machine, even though it is machine which help him to achieve this. This should make the therapeutic effect lasting much longer, and improve self-esteem of patients that is usually depreciated in stuttering people.

Proposed system consists of three basic components:

1. *Therapy Provider* – it provides Delayed Auditory Feedback (*DAF*) – delay, reverb and chorus effects.
2. *Stuttering Detector* – it analyses sound and detects whether stuttering occurred or not.
3. *Therapy Controller* – it controls the parameters of therapy according to therapy progress.

The four components imply the basic problems connected with building the whole system:

- How to detect stuttering?
- How to adaptively control the therapy?
- How to implement high speed and quality *DAF* on a standard PC?
- How to make the whole system work in the real-time?

Next subsections answer to these questions.

3.1. Therapy Provider

Let us start from presentation of Therapy Provider. It is a computer program working as traditional echo-correction device – it provides necessary sound effects to the user with ability to adjust them manually without use of any artificial intelligence. Without this module, whole effort for the *Orator* goes to waste.

Therapy Provider must contain:

- A buffer storing sound for specified amount of time, and then reproducing it (*Delay* module)
- A mixer for mixing sound from various sources (*Sum* module)
- An element providing a low frequency sine wave used as a delay value in chorus filter (*LFO* – Low Frequency Oscillator module)

Above modules are used in three filters required for *DAF* device modeling. Fig. 1-3 show the filter designs.

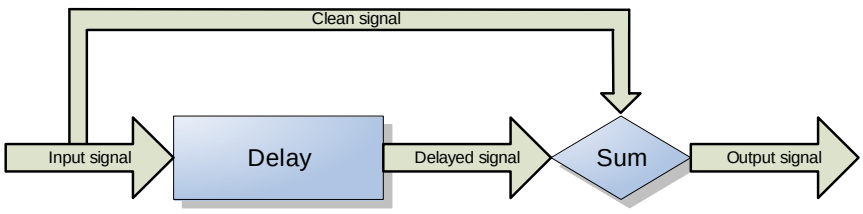


Figure 1. Echo filter design

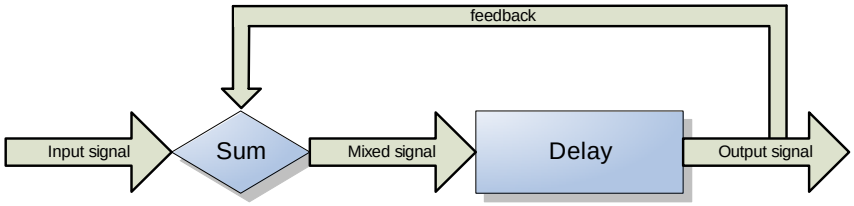


Figure 2. Basic reverb filter design

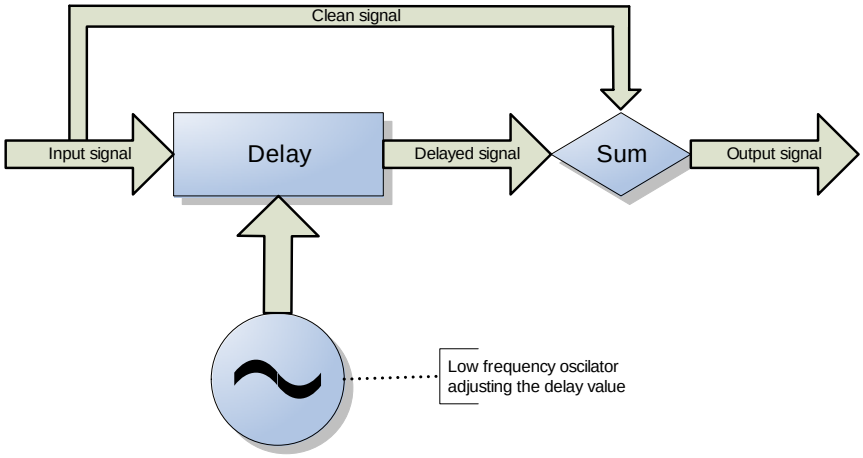


Figure 3. Basic chorus filter design

The *Orator* system has interface in Polish, but in can be used for stuttering therapy in other languages – the method is independent of nationality and speaking language. It is able to work in two modes of interaction with a patient:

1. **Manual mode** – only *Therapy Provider* module is used and the patient adjusts all the therapy parameters.

The system acts virtually the same as a traditional *DAF* device; the only difference is that users use sliders instead of knobs to adjust therapy parameters. A user uses sliders to adjust four basic parameters:

- Master volume – the global feedback volume
- Echo delay – the amount of delay
- Reverb volume – the amount of reverberation effect
- Chorus volume – the amount of chorus effect

2. **Automatic mode** – the parameters are adjusted by the therapy controller module

Interface of the *Orator* system used in automatic mode differs from the manual mode. There are no sliders since the therapeutic session is fully automatically controlled, but the system has to be calibrated before using it. Before starting a session, a patient has to go through three simple steps to setup and calibrate the system:

- Choosing initial volume – so the system will not hurt patient's ears.
- Recording noise sample for noise filtering component – noise filtering is unfortunately necessary if we do not use high-end microphone and sound card. To perform FFT noise filtering, a noise sample has to be recorded beforehand.
- Reading three sentences of text provided to calibrate the stuttering detection component.

After those three steps the system is ready to use, and does not really interact with the patient through other than auditory channel. The patient task is just to talk or read things laud; the system will adjust all the parameters itself.

3.2. Stuttering Detector

The process of detecting stutters in voice is crucial for automation of therapy. To control the therapy the system has to detect and evaluate a patient's stuttering. The process is complex; the flowchart in Fig. 4 presents the whole stuttering detection process.

Signal preprocessing

Recorded sound undergoes the preprocessing [10, 11, 15]. The first step is sampling – an analogous signal is converted into the digital one [3]. This signal is transformed using Fast Fourier Transform [1, 2]. C++ library *Fastest Fourier Transform in the West* [1, 12] was used as implementation of FFT. Axis X on the spectrogram represents time, and axis *Y* frequency (in Hertz) of a composite wave, the color of a given point shows the amplitude of a wave of given frequency in some given point in time. The darker the color is the higher amplitude. Some characteristic features of human voice are easily seen on the spectrogram, e.g.:
For instance you can easily notice on the spectrogram:

- Vowels – with characteristic spectrum showing all the harmonics [4];
- Consonants – which spectrum is "blurred" and for which it is difficult to figure out the dominant frequencies;
- One of the stutters – prolonged vowels.

It is possible to extract from FFT even more information in further processing that is going to be helpful in stuttering detection. The spectrogram not only shows some characteristic features of sound and voice, but also together with *Reverse Fourier Transform* it lets us design sound filters extremely easily. The Reverse Fourier Transform transforms spectrogram (signal decomposed into composite sinusoidal waves) back into original single wave that can be played through the speakers or headphones. Roughly speaking it is just reverse operation to Fourier Transform.

The filtering is very important, it proves that an ordinary microphone connected to an ordinary sound card in ordinary PC, records except patient voice also a lot of different noises. These are characteristic for the computers, i.e., noises of processor, and the frequency of electricity in a power socket – strong 50Hz signal. All this effects can be eliminated either by buying high quality, expensive equipment or by filtering the signal from unnecessary noises. Obviously, the second solution is more convenient for the users.

With FFT it is very easy to build highly effective noise filter that can even be adaptive and respond to specific noise pattern of a concrete hardware configuration and microphone type. The recipe is simple:

- Record noise sample – the signal recorded when there is a silence in the room.
- Calculate spectrum of the silence with FFT – this is going to be a characteristic of our filter.

Having the filter characteristic we can now filter out noises from any recorded sound, to do it we simply subtract the spectrum of noise from the recorded signal, hence receiving spectrum of the clean signal. If we want to play it through speakers or headphones we must then apply Reverse Fourier Transform to reproduce the signal.

Developed filter is an adaptive one because we can change its characteristic easily, and tune it to the specific hardware environment. This is very important feature because, as tested on many different microphones and PCs, they produce different kind of noises in recorded signal, and the filter designed for one device would not work well with another.

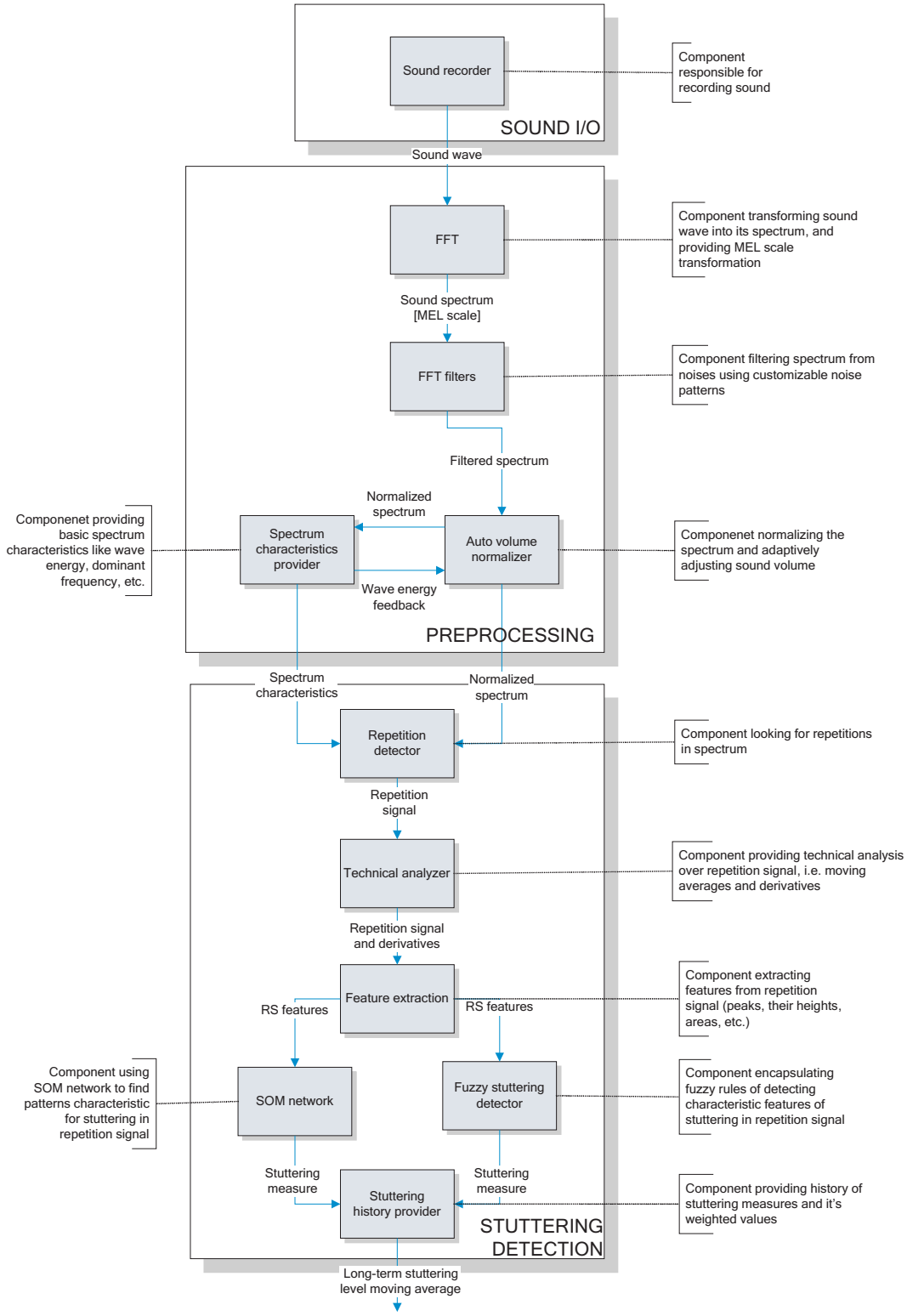


Figure 4. Stuttering detection process

Filters having constant characteristics are implemented in similar way. All used filters cut off the frequencies that are not audible for human ear and hence play insignificant role in stuttering detection. Filtering them out lets us decrease the amount of data for further processing.

It is worth underlying that from FFT we obtain a set of amplitudes at given frequencies. These frequencies are not scaled in convenient way. We can distribute the data more evenly

over a *Mel scale*, a scale that make each point of the spectral buffer carry more or less the same amount of information. This scale was created empirically and tries to map the perceived frequencies into a linear scale. This mapping is expressed by the formula [11]:

$$f[mel] = 2595 \cdot \log_{10}\left(1 + \frac{f[Hz]}{700}\right)$$

Because we must fill the Mel buffer, we transform it into:

$$f[Hz] = 700 \cdot \left(10^{f[mel]/2595} - 1\right)$$

Since in FFT buffer the frequencies does not correspond to buffer indexes so third transformation is necessary:

$$f[pos] = 700 \cdot \left(10^{f[mel] \cdot 1.3 / buffer\_size} - 1\right) \cdot \frac{buffer\_size}{sample\_rate}$$

This final transformation will give us an index in FFT buffer where we should look up for value when transforming it into Mel scale. The last step is to apply a windowing function when copying from buffer to buffer to make the result more accurate, than when performing a simple lookup.

For further processing we also need to calculate *wave energy*. A variety of possible methods are presented in [15], we chose the one of them – calculating the average peak height in Mel spectrum. Wave energy is an important feature of recorded sound; it is very closely bound to sound volume.

The problem is that a person when speaking to the microphone often moves from and towards it, hence causing volume changes. To have accurate data we must handle such situations. Component *Auto-Volume Normalizer* is responsible for this task – it scales the height of spectrum peaks by a given *amplification factor*. If rapid volume rise is detected it decreases the amplification factor, and oppositely – if it detects rapid volume fall amplification factor is increased. Such a simple system has adaptive features and keeps the volume in a given range quite well.

Stuttering detection

The idea of how to detect stutters comes from its definition and observation that they are in most cases repetitions of sounds or syllables, or sound prolongations (that in more general case can be also seen as kind of repetition). The other types of stutters are far less common and far more difficult to detect, hence we will first focus on what is vital for the system.

The repetition kind of stutters have quite characteristic spectrum. The spectrum patterns are simply repeated with a little change between repetitions. Across many repetitions of the same syllable pitch usually does not change, and overall shape stays quite similar. The other vital observation is that between the repetitions there are usually a small period of silence caused by a block of muscles in phonetic and articulation apparatus of a stuttering person. This block is clearly visible on the spectrum as a clean space and can be sharply expressed by a sound energy or simply its amplitude.

Developed stuttering detection module basis on two measures scattered in time: *wave energy vector* (described above, it is a history of wave energies in a recent past) and *spectral distance vector* (it shows how similar the current spectral pattern is to those seen short time ago). Spectral distance is a Cartesian distance between two vectors of amplitudes in two given points of time:

$$spectral\_distance = \frac{\sqrt{\sum_{i=0}^{n} ampl_{t1}[i] - ampl_{t2}[i]}}{n}$$

The aim of stuttering detection is to find repetition in analyzed signals. We model another measure that directly describes repetitions:

$$Repetition\_signal[i] = energyVector[0] \cdot energyVector[1] \cdot (MaxDistance - distanceVector[i])$$

The following assumptions lie on the basis of above formula:

1. We need to detect repetition of sound, not of silence, therefore we multiply energy vector in time *zero* by this in time *I*
2. We represent similarity of two spectral patterns as inverse of spectral distance.

Calculated vector we call *repetition signal*. This vector represents repetitions in recorded sound quite well. Now me have to find patterns characteristic for stuttering in the repetition signal.

SOM approach

Our first approach was to use unsupervised learning – SOM neural networks. Two premises motivated our choice:

1. Difficulties in preparing data for supervised learning, and
2. Having a repetition signal we might suppose that some certain repetition patterns will match to certain stutter types. Further, we might look for some shapes characteristic for stuttering. But a problem of finding shapes on a graph is very similar to problem of looking for formations in stock prices; hence we can use the knowledge of economists and apply it to stuttering detection. Moreover stuttering detection should be an easier problem, because we don't need to predict a formation, but just recognize it. The idea of using SOM was borrowed from formations prediction in stock prices.

This approach requires data preparation: calculation of moving averages of repetition signal and feature extraction: extraction of peaks and plateaus from a given graph. Next step is extracting features form each single peak or plateau. After a number of experiments with different sets of features, the feature set was limited to just one – normalized peak height, because other features have proven to either insignificant or highly cross-dependant. Standard Kohonen algorithm was used to train the SOM network.

Pattern discovery with SOM network seems to be quite a simple task when we have two classes of signal, i.e., stutters and fluent speech. To find out which parts of SOM got trained to recognize every class we follow a simple procedure.

- Preparing two sound files, one containing stuttering together with fluent speech, and second one containing just fluent speech.
- Training the network with both files
- Calculation the coverage for every file separately. As coverage we understand the average excitement of each specific neuron during the file playback, when the trained SOM is working in classifying mode. The excitement is calculated with a *diameter* parameter, that specifies if neighbors of excited neuron should also get excited and so participate in coverage. The higher the diameter is the more we look for *areas* that will probably detect given type of signal than a single neurons that were acting as *centroids* in training process.
- Find areas probably represent stutters (their coverage). We use a simple formula and applying it to every neuron:

$$stutters\_coverage = mixed\_speech\_coverage - C \cdot fluent\_speech\_coverage$$

where *C* is a constant that specifies what features the calculated coverage will have. If *C* is large that it is very unlikely that later on in detection process we will have false alarms. Small C will let us detect subtler stutters, but will yield with higher rate of false alarms.

Obtained results were far from what was expected. SOM rather learns how to detect pauses in speech. Careful analysis leads us to the probable reason: stutters are usually surrounded by blocks that are simply silence, but unfortunately SOM network did not distinguish them from

normal pauses. It was usually yielding stutter at ends of words where pause was starting. After this conclusion we changed our approach.

Fuzzy logic approach

It was quite amazing that SOM sometimes could not learn to recognize even basic stuttering patterns since they can be even easily noticed when graphed. Many observations clearly show that in many cases during a simple syllable or sound repetition, the repetition signal looks like in Fig. 5.

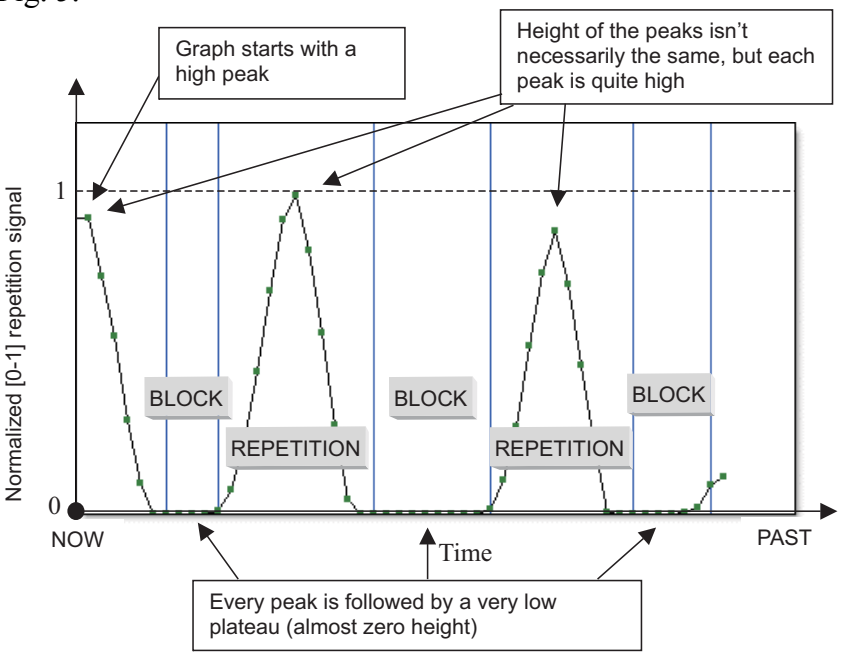


Figure 5. Syllable repetition stutter characteristic features

Those observations have also a theoretical base. The graph above represents double stutter represented by the peaks that is interleaved by speech blocks. The graph represents a typical stuttering pattern, which is: block – repetition – block – repetition. Examples of this pattern in stutterers speech can be for instance "I t-t-talk like this" or "I ta-ta-talk like this". Such repetition stutters are the most common across many stuttering people, and hence can be used as a base for the stuttering classifier.

Fuzzy logic seems to be a right tool for solving quite well but still not sharply defined problems, such as finding mentioned above pattern [13, 14]. For finding the pattern just two rules were necessary.

SINGLE_STUTTER is detected IF successive three peaks are: HIGH (peak 1), LOW (peak 2), HIGH (peak 3)
DOUBLE_STUTTER is detected IF: SINGLE_STUTTER is detected AND successive two peaks are LOW (peak 4), HIGH (peak 5)

Single stutter is single repetition of sound and double stutter is repetition that occurred two times in turn. Observations show that patterns similar to single stutter occur also in fluent speech so relying on them will not give the best discrimination results. However, patterns described by double stutter rule can be rarely observed in fluent speech whereas they are very common in stuttering. Hence when evaluating numerical value of stuttering level we give double stutter ten times bigger stuttering discrimination value than when single stutter is detected.

We define our fuzzy sets by simple ramp and triangular membership functions defined on log value of relative peak height. The actual numerical values on the graph were found after numerous experiments.

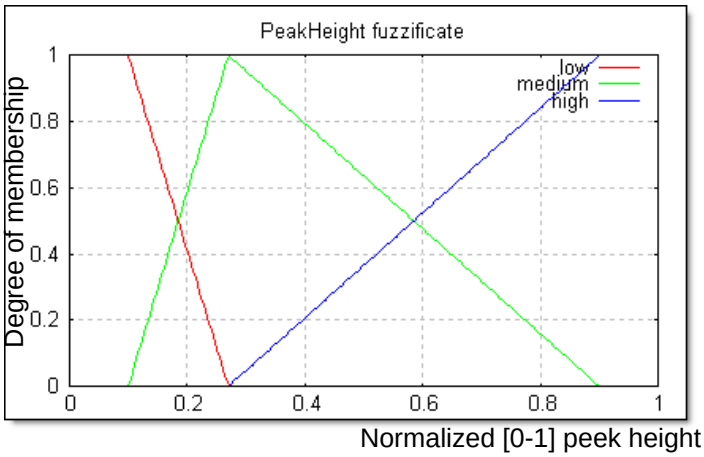


Figure 6. Peak height fuzzy set

Fuzzy classifier build in a way mentioned above works well enough to provide quality data for control component. However, since there are no objective stuttering measures, we cannot tell the exact accuracy of classification it provides. But we should keep in mind that the therapy control component does not need the classifier to be 100% accurate. We feed control component with long term moving average of stuttering level, so single false alarms or not detected stutters does not play much role in the quality of long term moving average.
Moreover the controller is self-tuning so the stuttering level does not have to denote any objective measure, but just should be able to tell if a person is stuttering less or more than some time ago. Having in mind these small requirements for stuttering detector we can tell it performs very well for the task it is given.

3.2. Therapy Controller

As a base for the automatic therapy control component we used a part of Perkins' stuttering therapy program, described in [6]. We consulted the method of single therapeutic session control with the experienced speech therapist.[2] We have learned that the session should start with delayed echo equal to 250 ms, and it should be decreased always when the significant progress is observed. The patients should have impression that at the end of session they are speaking without help of the system, and so they can carry on with fluent speech when they walk away from the computer, therefore we should also fade out volume by the end of the therapeutic session.

The *Therapy Controller* module was implemented as a fuzzy logic controller. As an input value we provide long term moving average of stuttering measure provided by the *Stuttering Detector*, the output values are just delay and volume of the echo (see Fig. 7).

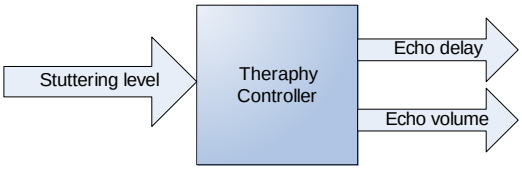


Figure 7. Therapy control component outer view

[2] M.Sc. Katarzyna Marszewska. Psychological and Pedagogical Clinic No.1 in Wroclaw.

Session progress is a vital measure determining the how the component should act. To calculate the progress we should store the history of stuttering level inside the controller. The progress is calculated as the difference between current stuttering level and last checkpoint stuttering level expressed in percent. Checkpoints are points in therapeutic session when we consider *therapy stage* to change. Five basic *therapy stages* were highlighted, namely: 1. starting stage, 2. early stage, 3. middle, 4. advanced, 5. final stage.
Significance of progress is another measure playing important part in fuzzy rules of the controller. This is the relation between change in stuttering level and maximum stuttering level observed in the current session. This measure makes the controller relative changes sensitive, whereas the absolute values of stuttering level are insignificant. Such design was necessary since the absolute stuttering measure provided by stuttering detector might vary among different people even if they stutter more or less the same (what is very subjective to say anyway). Quite simple fuzzy rules describe the way the controller works:

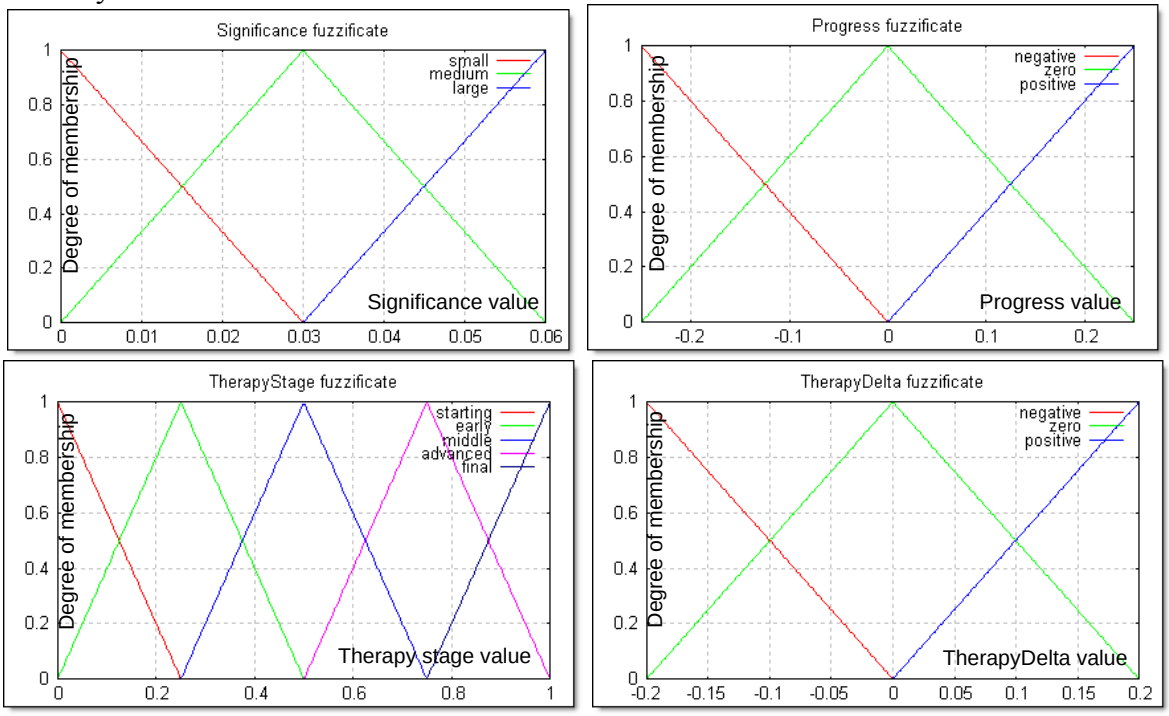
```
if Significance is small or medium                then TherapyDelta is zero;
if Significance is large and Progress is negative then TherapyDelta is negative;
if Significance is large and Progress is zero     then TherapyDelta is zero;
if Significance is large and Progress is positive then TherapyDelta is positive;

TherapyStage = TherapyStage + TherapyDelta;

if TherapyStage is starting  then Delay is very_large;
if TherapyStage is early     then Delay is large;
if TherapyStage is middle    then Delay is medium;
if TherapyStage is advanced  then Delay is small;
if TherapyStage is final     then Delay is very_small;

if TherapyStage is starting  then Volume is very_large;
if TherapyStage is early     then Volume is very_large;
if TherapyStage is middle    then Volume is very_large;
if TherapyStage is advanced  then Volume is large;
if TherapyStage is final     then Volume is small;
```

The fuzzy sets are defined as follows:



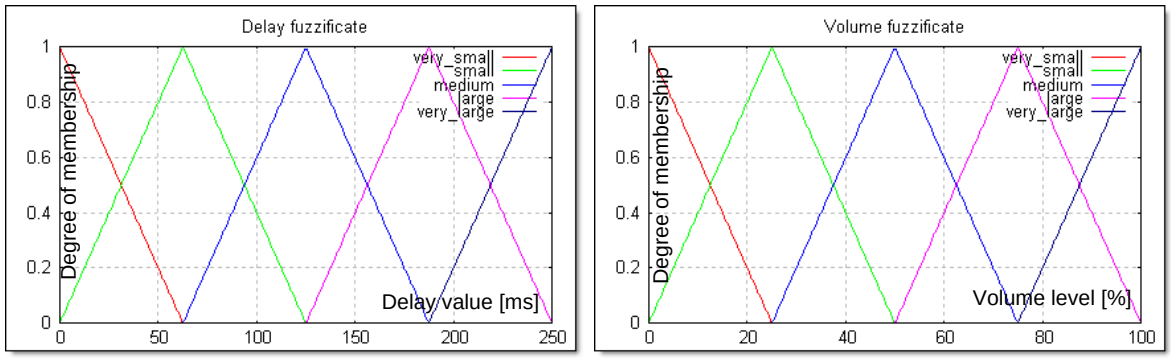


Figure 8. Therapy controller fuzzy sets

The changes of controlled variables (delay and volume) are smoothed by short term moving average. It is done to prevent rapid changes that would be hearable to the user. This makes the delay and volume adjustments smooth enough so they cannot disturb a patient nor grab his attention.

The developed fuzzy controller seemed to follow well the therapeutic session scenario provided by Perkins, and what is important, the controller does not need very high quality input data (the stuttering level) to perform well. This makes the whole system still effective even if stuttering detector is not doing its best, or if it does not detect all the different types of stutters

4. Results

The last stage of system development was to put everything together (*Stuttering Detection* module, *Therapy Controller*, *Therapy Provider*, and user interface), perform integration testing and to tune the whole system.

The *Orator* system performs its task of providing automated therapeutic session very well, and it is considered to be very helpful in everyday stuttering therapy. It solves an important problem of patient homework. Many speech therapists point a problem that a therapy cannot be effective if it is provided just once or twice a week during a visit at a clinic. Patient should work everyday at home to achieve good and stable results. The *Orator* system gives patient such possibility, providing quality echo-based therapeutic session. Nevertheless we should again emphasize that echo-correction is just one dimension of stuttering therapy and other methods should be applied as well to make it effective.

It is difficult to provide objective measures of how the software really influences the therapy process, since no objective stuttering measures were yet discovered. However subjective speech therapists opinions can be easily collected and they do provide a feeling how the system performs. Bellow few of such opinion are quoted.

„Orator application is highly useful in therapy of stuttering children, particularly the fact that a child can use it at home. Everyday work is very important for the therapy results and traditional echo-correction devices are often unavailable for our patients because they are simply too expensive. The option of automatic echo delay control seems to be interesting and promising. Therapeutic session carried out with it seems to give much better results than one carried just with a traditional device. I recommend the *Orator* system to every speech therapists using delayed auditory feedback in therapy of his patients.”[3]

M.Sc. Katarzyna Marszewska

Speech therapist from Psychological and Pedagogical Clinic No. 1 in Wroclaw

[3] It is a translation of the original opinion given in Polish.

5. Summary

Very important lesson was learned during the system design and implementation process. That is a simple rule saying that we should incorporate as much knowledge as we have about the task the system is given. Even though a trained SOM network looks mysterious and impressive, it acts far less effectively than two simple fuzzy rules that look actually trivial.

The resulting system user interface (described in Section 2) was simplified to minimum, and since the therapy is controlled automatically, patient task is just to talk. Such minimalism may introduce at first little confusion to users since it is usually expected from software to interact with human also through other than auditory channel. Hence it might be a good idea to incorporate some reading material or multimedia exercises into the program in the future.

The system was developed and proven to be useful in therapy of real patients in Psychological and Pedagogical Clinic No. 1 in Wroclaw. System made to be production stable, easy to use and install by an average computer user. Moreover a simple end-user webpage (http://orator.inside.net) for speech therapists and patients was developed to distribute the software to broader public.

Even though stuttering detection was good enough to carry the therapy it is still far from perfect when we analyze it separate from the whole system. This component can be definitely improved either by fine-tuning the fuzzy sets or introducing some more sophisticated rules into fuzzy stuttering detector, or by using completely different approach.

SOM network approach proven to be not efficient for the task given, but it might be that simple back-propagation network would perform very well. This is quite probable, but would yield a problem of providing learning examples for the network, and it has proven to be difficult and very time-consuming task to mark stutters in a sound file.

In fact only a subset of possible therapeutic methods was implemented, there is still an important method – frequency altered feedback – that could be incorporated into the system. Other methods could include metronome effect or sounding somewhat nicer chorus effect. Implementing all this features could improve effectiveness of the therapy but would create a need of controlling all of them, and unfortunately, no literature describing therapy scenario of such a complex system was found. This means that such scenario would have to be created by the authors, what would require in depth psychological and linguistic knowledge.

Therapy control component is currently using only stuttering level measure to carry on the therapeutic session. It is very likely that the therapy can be controlled more effectively having more information about the patient provided by the patient himself or by speech therapist. The problem is how to use this knowledge to personalize therapy, and again strong speech therapist's support would be required to develop rules for such a personalized system.

Other important research topic that is closely related to the subject and seems to be not covered by literature is neural model of speech production system. Creating such a theoretical neural model would be a huge step in understanding and explaining stuttering phenomenon, and probably create various new therapeutic techniques helpful in speech therapists everyday work. Having such model would make controlling therapy much easier and much more effective, but speech production system is for sure very complex, and today's neurological knowledge and modeling techniques might be not sufficient to develop such model, although it is definitely worth a try.

References

[1] Fastest Fourier Transform in the West. Massachusetts Institute of Technology. 2003. http://www.fftw.org

[2] Fourier transforms: signals processing (in Polish). 2003. http://student.uci.agh.edu.pl/~markrol/

[3] Sound. Great Internet Multimedia Encyclopedia (in Polish). 2003. http://wiem.onet.pl/wiem/00f6a8.html
[4] Harmonic. Great Encyclopedy PWN (in Polish). 2003. http://encyklopedia.pwn.pl/27437_1.html
[5] Fluency Tutorials. Horabail Venkatagiri. Iowa State University. 2003. http://www.public.iastate.edu/~cmdis476/tutorials/fluency_tutorials/definition_of_stuttering.html
[6] Tarkowski Z.: Stuttering (in Polish). Wydawnictwo Naukowe PWN. Warszawa 1999.
[7] Goldman-Eisler F.: Psycholinguistics. London, 1968.
[8] Roland-Mieszkowski M.: DSA (Digital Speech Aid) – a New Device to Decrease or Eliminate Stuttering. 1st World Congress on Fluency Disorders. 1994. (Accessed in 2003). http://www.digital-recordings.com/publ/pubdsa1.html
[9] Delayed Audio Feedback System for Treatment of Stuttering. Luminaud, Inc. 2004. http://www.luminaud.com/DAF.htm
[10] Smith Steven W.: The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Publishing. 2004. http://www.dspguide.com/
[11] Picone J.: Signal Modeling Techniques In Speech Recognition. Texas Instruments Systems and Information Sciences Laboratory Tsukuba Research and development Center Tsukuba, Japan. 2004. http://www.isip.msstate.edu/publications/journals/ieee_proceedings/1993/signal_modeling/paper_v2.pdf
[12] Wood G.: Simple utility classes. http://sucs.sourceforge.net/
[13] Mortensen Jan E.; JFS – fuzzy logic tool. 2003. http://inet.uni2.dk/~jemor/jfs.htm
[14] Kaehler Steven D.: Fuzzy Logic Tutorial. The Boeing Company. http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html
[15] DSP Algorithms - ISIP Foundation Classes Documentation. Institute for Signal and Information Processing, Mississippi State University. http://www.isip.msstate.edu/projects/speech/software/documentation/class/algo/
[16] Kaski S.: Data analysis with the SOM. 2003. Helsinki University Of Technology. http://www.cis.hut.fi/research/som-research/
[17] Bass audio library 2.0. Un4seen developments. 2004. http://www.un4seen.com/
[18] Stuttering Therapy Devices. Casa Futura Technologies. 2003. http://www.casafuturatech.com/
[19] Sheehan Stuttering Center. 2003. http://www.stutterssc.org