CHALMERS

An algorithm based on the Dawes/Redman criteria
for automated fetal heart rate analysis
Master of Science Thesis

ERIKA LÄTT NYBOE

Department of Signals and Systems

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden, 2011

Report No. EX018/2011

Abstract

Each year there are about 35 children in Sweden who die during labor due to lack of oxygen and even more are being injured. The social and economic consequences are severe and justify vast efforts to identify fetuses at risk. One way to find those is to interpret the heart rate pattern, which however requires skills and long experience, and even between experts, the interpretation may differ significantly. Therefore, attempts have been made to automatize the interpretation and examples are versions of Huntleigh's System 8000®, which from gestational week 26 classifies the heart rate trace as normal or pathological based on the Dawes/Redman criteria. The system provides an indicator of the fetus health status and if further actions are needed in order to avoid damage. These criteria are well established in many countries.

The current master thesis investigates the possibility to implement analysis based on the Dawes/Redman criteria as a product in STAN S31, which is an apparatus used for fetal heart rate monitoring developed by Neoventa Medical AB in Mölndal. This includes legal issues as valid patents protecting the algorithms and practical issues as if relevant publications includes enough details to implement a version with claims on similarity.

To be able to validate the implemented algorithm against the original, both versions have been applied on the same data. To assess the results a comparison is made with how much the parameters change if we delay the start time ten seconds of the original algorithm. The results show that the output differences between Neoventa's and Huntleigh's versions are within the same range as the effect of a delayed start of the analysis using Huntleigh's version.

Keywords: Fetal heart rate, fetal monitoring, CTG, Dawes/Redman criteria

Table of Contents

1. Introduction

1.1 Pregnancy, labour and associated complications

All pregnancies would ideally be without complications but that is not always the case. Fortunately there are techniques nowadays to monitor the fetal heart rate so doctors and midwives are in many cases alarmed in time. Reasons why we would like to monitor the fetal heart rate could be: reduced fetal movements, uterine pain, suspected fetal anomalies, intrauterine growth restriction, hypertension or pre-eclampsia [1]. Antenatal analysis of the fetal heart rate may indicate if the fetus is acidaemic, hypoxic, anaemic or if it has an infection, an arrhythmia or if the fetal nervous system is damaged. The analysis can help us to be able to predict if a natural delivery is possible or if an intervention is necessary. Below are some complications explained that may occur during pregnancy or labour.

Lack of oxygen during pregnancy may occur due to an infection of the fetus, a malformation of organs of the fetus or a placenta abruption. During a placental abruption the placenta is partially or completely separated from the lining of the uterus before the child is born [2]. It is considered as a serious condition and may lead to, in the worst case, intrauterine death. One other reason for lack of oxygen is that the umbilical cord may be pinched off.

Another complication that may occur during pregnancy is pre-eclampsia, that is a condition where hypertension arises and great amounts of proteins is found in the urine [3]. Pre-eclampsia develops from week 21 and is one of the most common dangerous pregnancy complications because it may affect both the mother and fetus.

Hypoxemia is the first phase during lack of oxygen and is defined as decreased partial pressure of oxygen in the blood. It could be long-lasting and my result in reduced growth rate of the fetus. The fetus may control hypoxemia for days and even weeks.

Hypoxia means lack of oxygen and can result in neurological damage or prenatal death. This may be a consequence from a delayed birth or that the umbilical cord is pinched off.

Asphyxia is a condition during which the supply of oxygen to the body is being severely reduced for a prolonged time [4]. In the end of such a period, metabolic acidosis occurs in the central organs and the cardiovascular system collapses which leads to brain and heart failure. The fetus has to be delivered in a couple of minutes in order to avoid injuries.

Acidosis is the process during which the acidity of blood plasma increases, the arterial pH falls below 7.35 [5]. Acidemia defines the state of a low pH in the blood. Metabolic acidosis occurs during increased production of metabolic acids or reduced ability to remove waste products via the kidneys. Acidemia refers to a low PH in the blood and acidodsis to a low PH in the tissue. Very often acidosis occurs before acidemia.

Anaemia is a decrease of red blood cells or a reduction of hemoglobin in the blood and it will lead to lack of oxygen in organs of the fetus, hypoxia [6].

1.2 Cardiotocography

Cardiotocography, CTG, is a technique to record fetal heart rates during the third trimester of the pregnancy and in addition to uterine contractions during the delivery phase [6]. The apparatus used for this purpose is called a cardiotocograph or an electric fetal monitor and is nowadays being used worldwide as standard equipment for hospitals in developed countries. By interpreting the recorded data in real time the experienced user is able to detect fetal distress. Both the uterine contractions and the fetal heart rate may be recorded externally or internally and the CTG traces are printed out or/and stored at a computer.

To obtain the fetal heart rate externally, an ultrasound transducer is placed on the abdominal wall of the mother. For internal measurements a scalp electrode is attached to the head of the fetus after rupture of the membranes. The latter of these techniques is more precise.

The tocodynamometer, also called toco, measures the uterine contractions externally and is strapped around the abdominal wall. It then measures pressure changes which are correlated with the uterine contractions. For more precise measurements, a pressure catheter is instead inserted into the uterine cavity. This requires that the membranes are ruptured. In Sweden, the external version is the clearly dominating technique.

CTG interpretation can be very complex and the user has to consider levels, patterns and relations between the traces. Patterns are mainly characterized during delivery by four parameters and their relation to the contractions: accelerations, decelerations, heart rate variability and basal heart rate. An acceleration is defined by a transient increase in heart rate and a deceleration is a transient decrease. To be able to detect a decrease or increase of the fetal heart rate a baseline is used. The baseline follow slow variations of the fetal heart rate, see Figure 4. The heart rate variability is a measurement of how much the heart rate varies. There are two different types of variability called long-term variation and short-term variation.

A primary indicator for a healthy fetus is that accelerations and fetal movements are present in association with a good variability [1]. The basal heart rate is the average heart rate when accelerations and decelerations are excluded. With knowledge about these parameters and the data of the contractions, the status of the fetus may be categorized into three different types: normal, indeterminate and abnormal [6].

A sinusoidal pattern is a rare fetal heart rate pattern and is estimated by investigating the ratio between short-term variation and long-term variation. A low-frequency sinusoidal pattern superimposed on a flat trace is associated with disease and poor fetal outcome. A high-frequency component can be an indication of fetal anemia, fetal or maternal hemorrhage or fetal intracranial hemorrhage [7].

Below is an example, Figure 1, of a trace from a fetus obtained by ultrasound. There are three major accelerations, indicated with arrows, in the trace and it has good heart rate variability. The fetal heart rate is both measured in beats per minute and in pulse intervals in milliseconds. For example corresponds 120 bpm to 500 milliseconds (60 000/120).
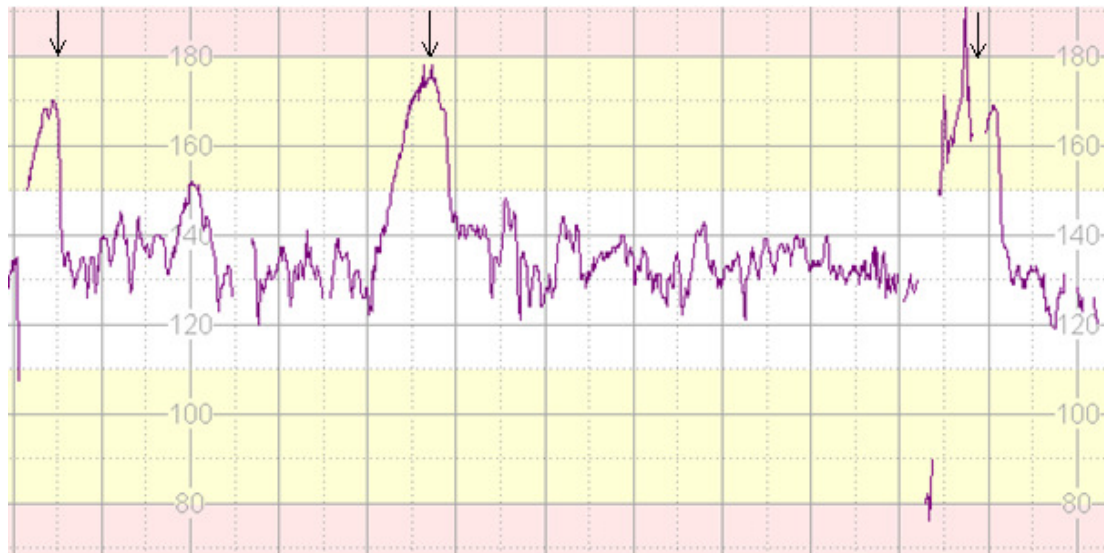
Analysis of antenatal, before the delivery phase has started, recordings are slightly less complex in that there are no contractions to consider. From about week 28 (gestational age) the fetus cycles between active and quiet sleep [1]. During active sleep the trace of a healthy fetus should contain accelerations, increased heart rate variation and fetal movements. However, it is not possible to assess fetal wellbeing during quiet sleep since it is associated with decreased variation and reduced fetal movements [1].

1.3 Dawes and Redman

In 1977 professors Dawes and Redman at Oxford University in the UK started to investigate the connection between outcomes and traces of the fetal heart rate [1]. It lead to a development of a computerized system that analyses antenatal data for gestational of week 26-42, and the analysis uses a set of criteria called the Dawes/Redman criteria to assess a recording. The system is today named Sonicaid Fetalcare, owned by Huntleigh, and is based on a database with more than 73 500 traces [7].

The Dawes/Redman criteria in Huntleigh's system is the foundation of an automated analysis of the fetal heart rate which alerts after 60 minutes if not all the criteria are fulfilled [1]. The first analysis is made after 10 minutes and if all the criteria are met, the system indicates that the fetus is healthy. However, if not all criteria are met it will continue the recording and evaluate every 2 minutes until either all criteria are met or the recording has been going on for 60 minutes, whichever comes first. The Huntleigh system evaluates accelerations, decelerations, basal heart rate, long-term variation and short-term variation. The baseline will be recalculated for every new evaluation.

The Dawes/Redman criteria may help to predict the outcome. However, even if the criteria are met it does not guarantee that the fetus is born healthy [7]. A placental abruption may happen very suddenly, without any warning and with a devastating result. However, with the Dawes/Redman analysis the risks can be reduced and enhance the probability of a successful delivery [1].

The structure of the Dawes/Redman algorithm is illustrated below in Figure 2. The program starts with the pre-processing of the signal and ends with storing the results of the analysis. The

Dawes/Redman criteria are evaluated in the end of the process. To check against the criteria the program detects accelerations and decelerations. It also estimates the baseline, short-term variation (STV), long-term variation (LTV) and the basal heart rate. The program also evaluates if there is a possible sinusoidal pattern.
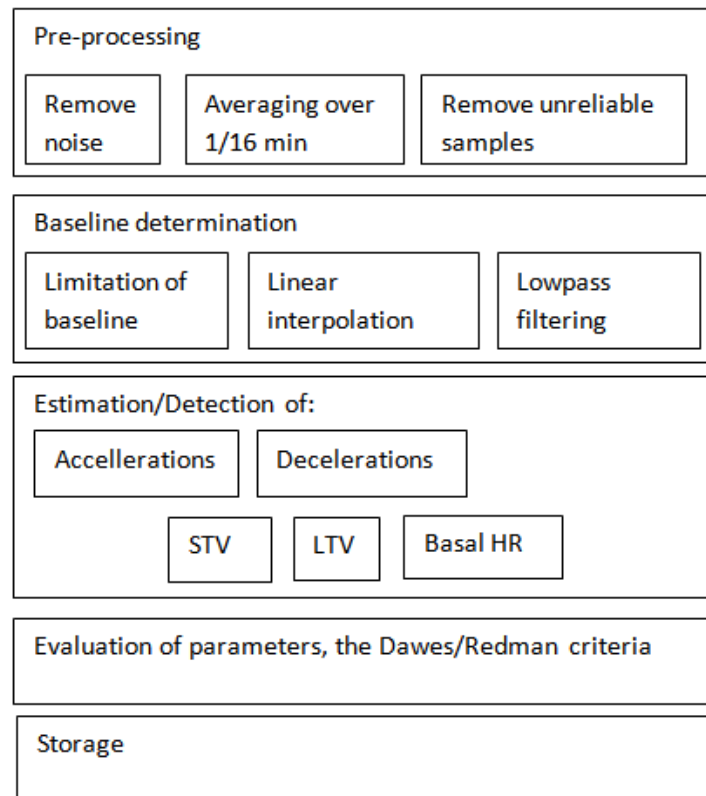

Figure 2. A schematic block diagram over the Dawes/Redman algorithm.

1.4 STAN S31

STAN S31 is a CTG system with the addition of a computerized real time analysis of the fetal ECG trace. The system examines the T-wave amplitude and alerts at changes related to hypoxia, lack of oxygen [5]. This analysis is the feature that discriminates STAN S31 from the other CTG systems [5]. STAN S31 is developed and owned by the company Neoventa Medical AB, the head office is located in Mölndal, Sweden.

2. Aim and objectives

The aim of this project is to implement an algorithm based on the Dawes/Redman criteria that can be included in STAN S31 for antenatal analysis of the fetal heart rate. These criteria are described in several published journal papers, but it is uncertain if the publications are detailed enough to allow an implementation of the original algorithm. Furthermore, it is not clarified if there are active patents protecting the algorithms.

Objectives of the project were to:

- Investigate if there are any currently active patents regarding the Dawes/Redman criteria which could hinder an implementation in STAN S31

- Determine if publications contain enough details to enable an implementation with claims on similarity
- Implement an algorithm as similar as possible to the Dawes/Redman criteria in Matlab
- Generate simulated signals with known heart rate patterns
- Collect real data traces
- Run the data through the Huntleigh's and Neoventa's version
- Create an architecture for analysis based on the Dawes/Redman criteria in the language C++ and implement it
- Validate the implemented algorithm

3. Methods

The first task was to find patents concerning the Dawes/Redman criteria in Europe and in the US. For this purpose three different search engines were utilized over the internet [8-10]. The search criteria were: Dawes AND/OR Redman, Dawes/Redman, fetal heart rate analysis. For the literature study Chalmers database, the library of Gothenburg University and scirus.com were used to search for relevant articles and publications treating the Dawes/Redman algorithm [11-12].

An implementation of the Dawes/Redman algorithm in Matlab was then possible after enough information about the original analysis had been collected. Matlab was selected because it is a high-level technical computing language with interactive environment for algorithm development, data visualization and data analysis. The algorithm was tested iteratively both with simulated signals and with real fetal heart rate data. With knowledge about the examined parameters and how the analysis should work I was able to detect bugs and determine the functionality of the implemented program. The parameters are available after each analysis of a trace is performed. The simulated signals were created in Matlab and played on a speaker unit which was connected to the ultrasound transducers of STAN S31. The real data traces were obtained from the hospitals in Varberg and in Mölndal, where I assisted the midwives with some registrations.

The final validation of the implemented algorithm was performed in Oslo at Akershus hospital. The hospital had a program containing the original Dawes/Redman algorithm installed at a computer. The traces were analyzed by the original algorithm, starting with the newest and continuing backwards in time in their database. Each trace was also processed by my implemented algorithm. Then a comparison was performed between the two systems by examining the obtained parameters from each analysis. The evaluated parameters were: signal loss(%), Basal heart rate(bpm), number of Accelerations and Decelerations, STV(ms), High episodes(minutes), Baseline maximum and minimum during the first 10 min(bpm) and Criteria met(yes/no).

The Huntleigh's version is time invariant, meaning that by just moving the start of the analysis some seconds resulted in different outcomes even though the examined data did not change. Therefore a similar validation between Huntleigh's original program and a 10 seconds delayed analysis was performed. Then I was able to compare the variation between Neoventa's and Huntleigh's version and the detected osäkerheten within the original algorithm. The two evaluations were realized in the same way, investigating the same parameters with identical methods. The validation was time consuming and the number of traces for each evaluation was set to 25.

In order to enhance the distribution of the results from the validation a method called "bootstrapping" was used [13]. The traces with matching analysis were all numbered between 1 and 25. By randomly picking a number between 1 and 25, which corresponds to a specific trace, I could create 1000 test suites with 25 traces in each. Then we may examine the median and percentiles which now will give us a better and more reliable distribution of the obtained results. The percentiles were set to the 10th and 90th. Meaning that if the compared results are not overlapping with their percentiles conclusions can be drawn that the median differs with 99% certainty.

A future product would be programmed in C++ since it is most commonly used for STAN S31. Therefore it is of interest to transform the Matlab program into a C++ project. The basics of the implemented Matlab code was implemented and integrated in already used projects at Neoventa.

4. Currently active patents

There are currently no patents, neither in Europe nor in the US, which protects the Dawes/Redman algorithm [8-10]. I found two patents that were of any relevance for my project but they were however not protecting the algorithm [14-15]. One patent was about improving the currently utilized monitoring technique for the fetus and mother during labor, also used in the Huntleigh system, by applying a localized group of electrodes to a patient's skin. The other patent concerned a fetal monitoring system and outcome predictor which included a system for automatically assessing fetal health and predicting fetal outcomes. The ideas were founded and inspired by the work of Dawes and Redman but aimed to build a system that would adapt and learn about new situations in order to asses fetal health from noisy data.

5. The Sonicaid system

In this chapter the Dawes/Redman algorithm is described. One of the objectives of the project was to determine if enough information can be found in publications about the algorithm to allow an implementation with claims on similarity. Therefore, the details concerning the algorithm are important results.

5.1 Pre-processing

In the first step of the analysis a filtration is made. This filter removes the samples that are not within the range 65%-175% of the average of the two last valid samples. If a sample is removed the two following are also considered not valid [16]. Though, the two last samples will be used as valid samples while computing the average. Otherwise, too big parts of the original signal would be lost.

The signal is then averaged over consecutive 1/16 minute periods (3.75 seconds) [7]. These averaged values are later used in the whole analysis, for example during the computation of LTV (long-term variation) and STV (short-term variation). Thus, every 3.75 seconds a new average is being computed, but if there is no valid sample during the period the value is instead later estimated by interpolation. The signal loss is estimated as the proportion of 3.75 seconds epochs for which no valid pulse interval was found [17].

After the averaging, changes in the heart rate intervals greater than 75 milliseconds from the average of the signal are detected [18]. Then, if the values have an instantaneous return to the average value, it is considered not valid.
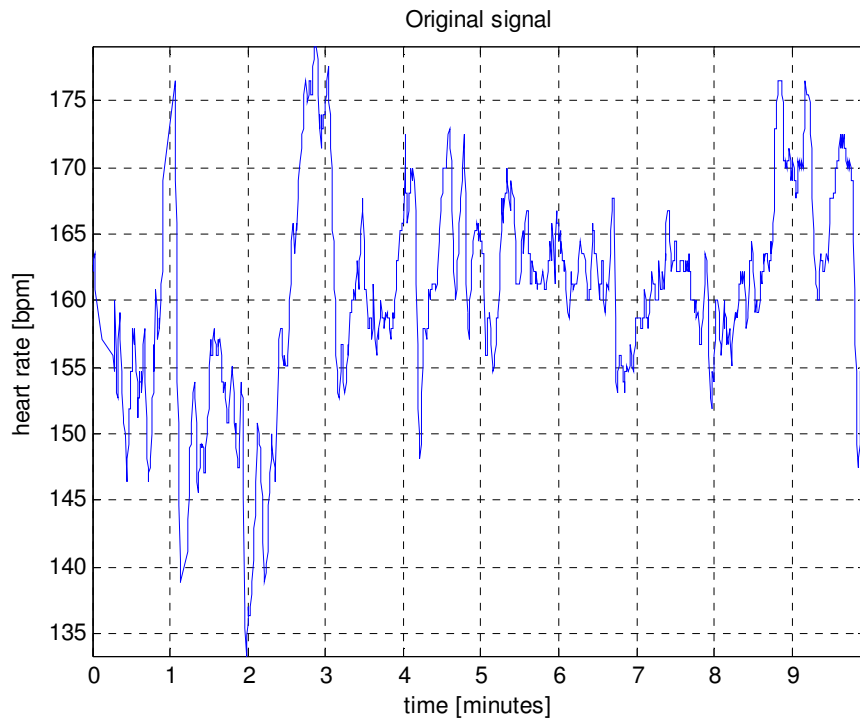
Figure 3. An example of a fetal trace before pre-processing is applied, the x-axis is the time [min] and the y-axis is the heart rate [bpm].

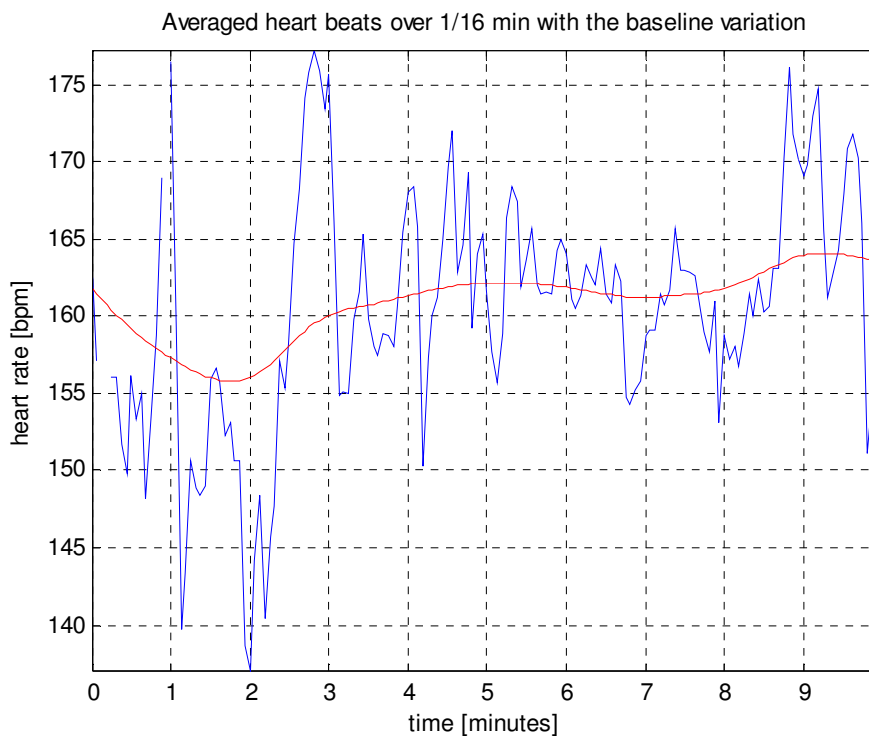Figure 4. A filtered and average trace with a baseline marked as red in the graph, In the graph the x-axis is the time [min] and the y-axis is the heart rate [bpm].

In Figure 3 we can see a heart rate trace prior to signal pre-processing. In Figure 4 the signal has been run through the error algorithm during which unreliable samples are deleted and then averaged over

1/16 minute epochs. The trace obviously gets smoother after the averaging, which is shown in Figure 4.

5.2 Baseline

In order to detect accelerations and decelerations, and estimate LTV and basal heart rate, a baseline needs to be determined. The estimation of the baseline is performed in steps. First, a value is calculated which determines the limits of the baseline. To find this peak value a histogram of the entire signal is used for the averaged pulse intervals. The resolution was not mentioned but I chose 1.0 ms because some examples used the same and I was also able, with this resolution, to detect all frequencies in Hz [18] [19]. Then the frequency resolution was smoothen out [19]. Since it was not described which filter to use, I chose the following smoothing filter with following coefficients:

[0.5 1 0.5]/2.

The algorithm then goes through the frequency distribution from right to left (high to low frequencies). The value that first satisfies the following criteria is then chosen:

- At least 12.5% of the frequency distribution shall lie to the right of the peak
- The peak value must exceed the five next values to the left of it in the frequency distribution
- The sample shall contain at least 0.5% of the total frequency distribution or not differ more than 30 ms from the largest sample [7].

If no sample is found that satisfies these requirements, the peak value of the frequency distribution is used. In order to delimit the baseline, the values that are not within the range +/- 60 ms from the calculated peak value are not considered valid for the baseline estimation [19].

One example of how a frequency distribution can look like is shown below in Figure 5. The green circle spots the peak that is being selected.
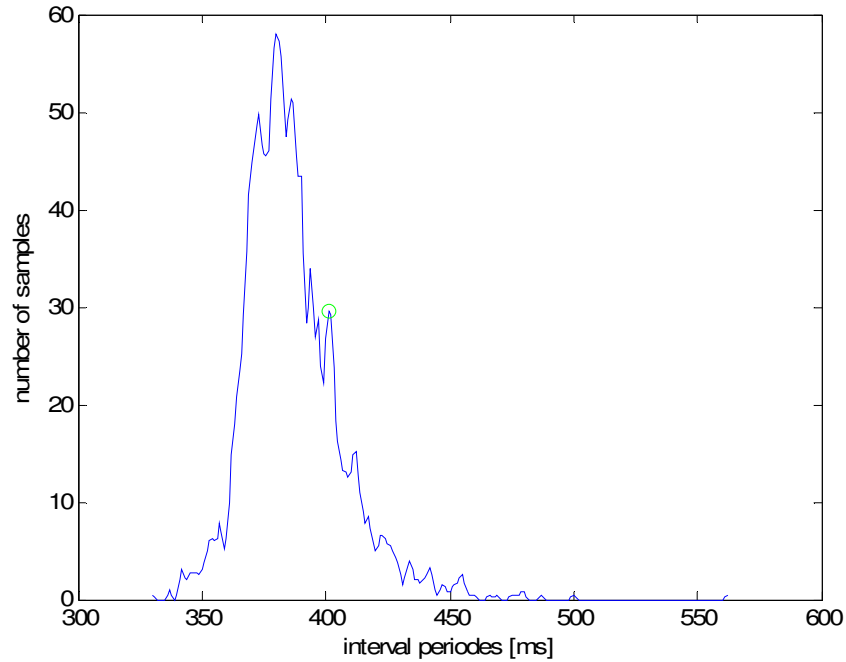
Figure 5. An example of a histogram or the fetal heart rate where the y-axis is the number of samples and x-axis is the interval periods [ms], within one trace. Out from this a peak value may be calculated to be used during the baseline filtration.

Continuing the estimation of the baseline, all not valid values are linearly interpolated. The corrupted values are interpolated between the two surrounding valid samples [19]. The entire signal is then processed through a low-pass filter with properties according to . An example of an estimated baseline is shown in Figure 4 where it is marked with red color.

Table 1. Values of how the low-pass filter attenuates different frequencies in the original Dawes/Redman algorithm.

Frequency (cycles/minute)	Attenuation (%)
2	0.2
1	2.7
0.5	9.8
0.25	28.7
0.125	60
0.0625	82
0.0312	93

5.3 Accelerations and decelerations

After the baseline is determined, accelerations and decelerations are identified. Accelerations are defined, according to the Dawes/Redman algorithm, to be an increase in heart rate at least 10 bpm above the baseline for more than 15 seconds [1]. The algorithm makes a difference between accelerations above 15 bpm and accelerations between 10-15 bpm.

Examples of accelerations are given in Figure 6. Accelerations greater than 15 bpm from the baseline are marked light grey and smaller accelerations between 10 and 15 bpm are coloured dark grey. The figure is obtained from my implemented algorithm. Both the number of detected accelerations and decelerations depend very much on how the baseline varies.
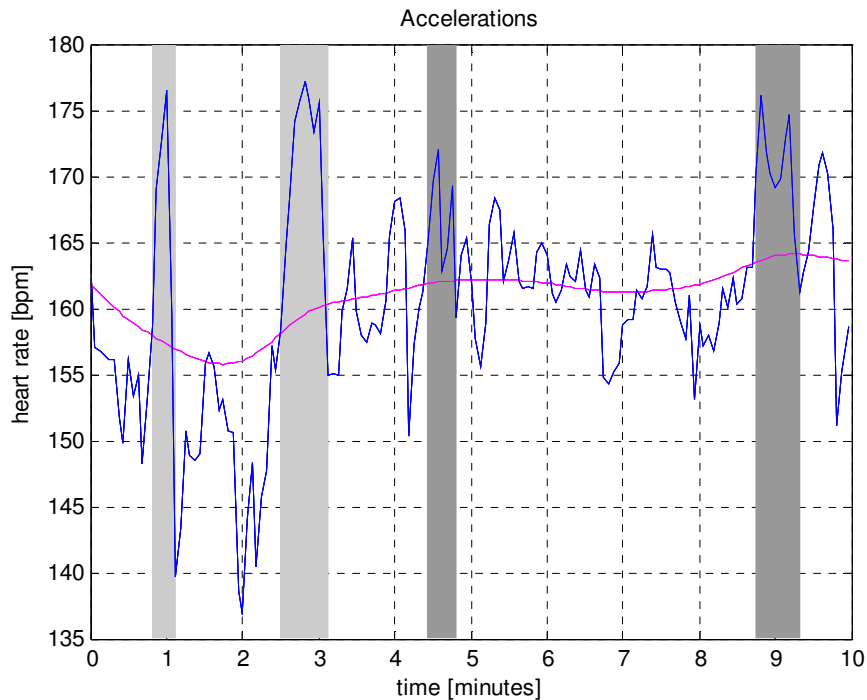
Figure 6. The graph indicates with light grey big accelerations and darks grey small accelerations which were found during the analysis.

A deceleration is a decrease in heart rate below the baseline, either 10 bpm for at least 60 or 20 bpm for at least 30 seconds. The decelerations are quantified by "lost beats", which is the expected heartbeats (the baseline) subtracted by the true value of how many heartbeats really occurred during the deceleration [20].
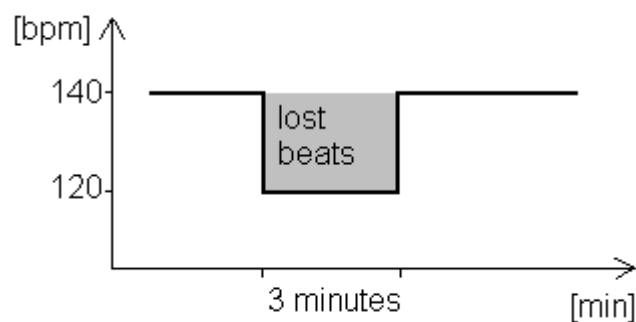
Figure 7. A schematic picture of how a deceleration is calculated and measured in the Dawes/Redman algorithm.

Figure 7 is an example of how lost beats are estimated. If there are no decelerations we would have had 140*3=420 heartbeats during this three minute period. But due to the deceleration only 120*3=360 heartbeats were detected. This means that we "lost" 420-360=60 heartbeats during this deceleration.

5.4 Long-term variation

In Huntleigh's algorithm a parameter called long-term variation (LTV) is estimated and evaluated. LTV is measured by first discarding samples that contain more than 50% signal loss [6]. Then each minute

of the whole signal will be examined starting with time equals zero seconds. To estimate LTV, the maximum and minimum excursion from the baseline for each minute are detected and the difference is called the minute range. If there is no excursion below the baseline, which is normal during accelerations, the minimum value of the baseline is used.

With this information the algorithm can identify high and low variation in the trace. An episode of low variation is when 5 out of 6 minute ranges are below 30 milliseconds. For high variation, 5 out of 6 minute ranges must exceed 32 ms and the total episode must be above the first centile for the corresponding gestational age [20]. If the period is in the beginning or in the end it is sufficient with 4 out of 5 minutes for both high and low variation [16].

The first, third and tenth centile for LTV are specified below in Table 2 and they are all used in the Dawes/Redman criteria [16]. These centiles are calculated from a database with more than 75000 traces and they are used to decide whether or not a trace belongs to a healthy child.

Figure 8 is an example of how LTV varies for an analysis during 15 minutes. We can conclude that the analysis has an episode of low variation between the 8th and 14th minute. That is because five out of six consecutive minute ranges are below 32 milliseconds.

Table 2. The table holds the first, third and tenth centiles of long-term variation obtained from a database with 75000 traces.

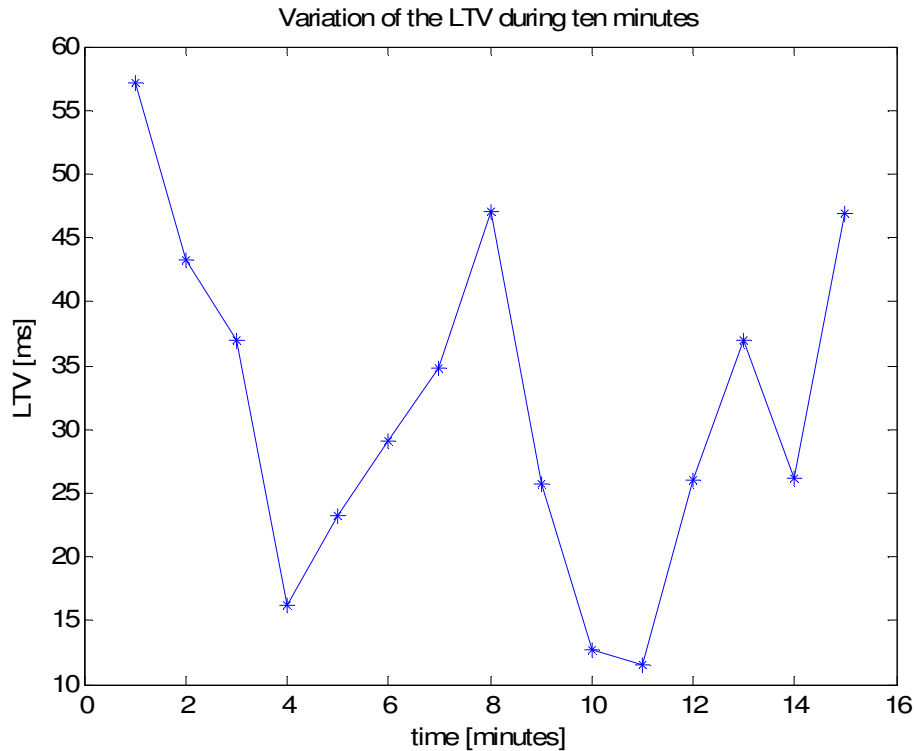| Gestational week	First centile	Third centile	Tenth centile
26	10.7	11.75	12.75
27	10.75	11.75	12.75
28	10.65	11.5	12.75
29	10.5	11.5	13
30	10.4	11.5	13
31	10.4	11.75	13
32	10.4	11.75	13.25
33	10.5	12	13.25
34	10.5	12	13.50
35	10.75	12.25	14
36	10.8	12.75	14.25
37	10.85	12.75	14.50
38	11	12.75	14.75
39	10.85	13	14.75
40	10.85	12.75	14.75
41	11.75	13	14.75

Figure 8. An example of how the long-term variation varies over fifteen minutes within the fetal trace . In the graph the x-axis is the time [min] and the y-axis is the long term variation [ms].

5.5 Short-term variation

Unlike LTV, the short-term variation is independent of the baseline and is a measure of how fast the heart rate varies. In Huntleigh's system the epoch-to-epoch is calculated as an estimation of STV. The variation is also measured by first discarding samples that contain more than 50% signal loss [7]. Then the differences between neighboring epochs, averaged pulse intervals in milliseconds were calculated.

Then two averages are calculated, first one for each minute of the signal and then one over the entire signal [7]. The averaging is performed in two steps rather than one. It is performed to have equally weighting to each minute instead of each epoch. This is due to greater signal loss is associated with high variation which means that the number of valid epochs is less than the number during low variation. If an average of the STV is performed in one step it would be biased towards low variation in the trace. The estimation of the STV is very important since it is shown that a trace without any high variation has a STV that correlates well with intrauterine death and development of metabolic academia. [7]

In Figure 9 the STV is estimated for each minute of a 15 minutes analysis, before the second and final averaging. What we can see is that the STV has a good variation during the three first minutes and then has a period with small epoch-to-epoch changes.
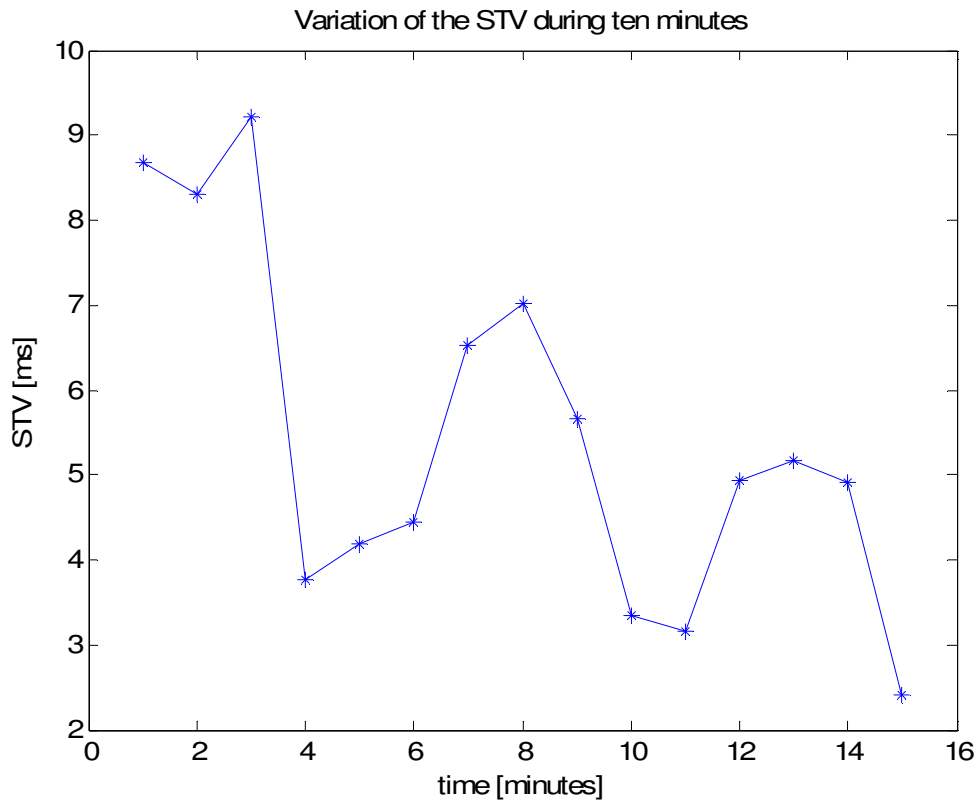
Figure 9. The figure shows how the short-term variation changes during the analyzed fifteen minutes of the trace, the x-axis is the time [min] and the y-axis is the short-term variation [ms].

5.6 Basal heart rate

The basal heart rate is an average of the averaged fetal heart rate, measured over all episodes of low variation. If no episode of low variation is detected in the trace, the peak value which was calculated during the estimation of the baseline is chosen [7]. The basal heart rate is measured in beats per minute.

5.7 Sinusoidal patterns

To be able to determine if a sinusoidal component is likely to be found in the trace the LTV and STV are examined. In absence of a sinusoidal rhythm STV and LTV are strongly correlated and the ratio is calculated to be 0.186+/-0.024(mean+/-SD). If STV divided by LTV is less than 2 SD below the mean a low-frequency sinusoidal component is likely to be found and the user is informed. If the ratio is more than one SD above the mean, further analysis is instigated. Then a sinusoidal component could be confirmed if there is a peak in the frequency distribution at 2-5 cycles per minute. The peak-to-peak distribution of the fetal heart rate is calculated over a minimum period of 20 minutes.

5.8 Dawes/Redman criteria

The output from the system is a categorization of the complete trace based on the parameters described and a set of criteria called the Dawes/Redman criteria. These have been adjusted over more than twenty years, but my program is based on the latest version [7].

A flow chart is presented below, Figure 10, of how the Dawes/Redman criteria are used. The corresponding number in the boxes can be found in the following list:

1. Does the recording contain one period of high variation?
2. Is STV greater than 3.0 ms?
3. If STV is lower than 4.5 ms, is then the LTV averaged across all episodes of high variations greater than the third percentile for gestational age?
4. Is there any evidence of a high-frequency sinusoidal rhythm?
5. Is there at least one acceleration?
6. Is the fetal movement rate greater than 20/hour?
7. Is the LTV averaged across all episodes of high variation greater than the tenth percentile for gestational age?
8. Is there at least one fetal movement?
9. Does the recording contain at least three accelerations?
10. Are there any decelerations exceeding 100 lost beats?
11. Is the recording less than 30 minutes?
12. Is there more than one deceleration between 21-100 lost beats?
13. Are there any decelerations greater than 20 lost beats?
14. Is the basal heart rate within the interval 116-160 beats/minute?
15. Is the LTV within 3 SDs of its estimated value?
16. Is the STV greater than 5 milliseconds?
17. Is there an episode of high variation with more than 0,5 fetal movements per minute?
18. Is the basal heart rate equal or greater than 120 beats/min?
19. Is the signal loss less than 30%?
20. Are there any suspected artifacts at the end of the recording?
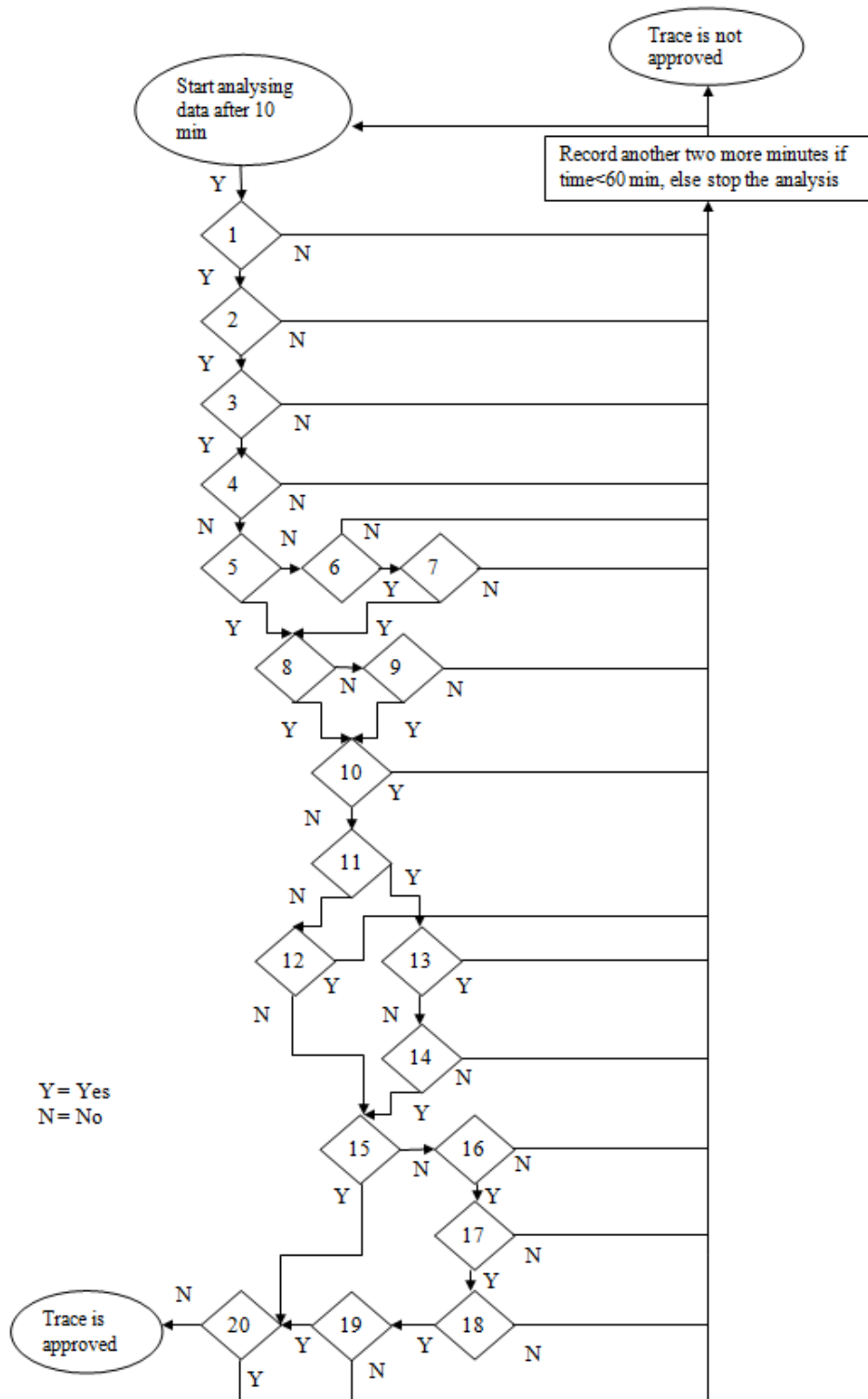
Figure 10. A flow chart for the implemented program.

6. Validation

This chapter demonstrates the results and discussion of the validation of my implemented algorithm. Even though many articles have been published concerning the Dawes/Redman algorithm it is not possible to find one article where the whole algorithm is described. Therefore different articles have been used while implementing the algorithm. Since many articles were used, dated from 1982 until

2002, it creates an uncertainty if whether or not the implemented algorithm is exactly the same as the original version. Even though I found potential interesting details in some articles concerning the Dawes/Redman algorithm, it was hard to interpret the information since it was written a bit fuzzy in a popular scientific way.

6.1 Pre-processing

While implementing the error algorithm, described in articles, I discovered that my implementation removed too few samples compared to the original version [18-19]. I then modified the filter to remove even more data, but still the algorithm removes less compared to the original.
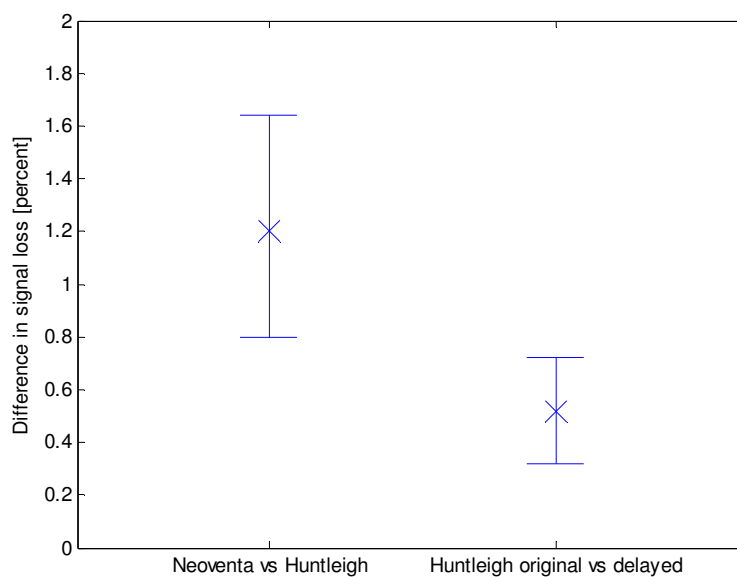
Figure 11. A figure showing the difference in signal loss between the compare versions. In the graph the x-axis is the two comparisons and the y-axis is the differences in signal loss measure in percent.

From Figure 11 the differences in signal loss are illustrated. The signal loss depends on the amount of removed samples compared to the whole analysed trace. The cross indicates the median values of the evaluations and the limits indicate the 2.5th and 97.5th percentiles. That means that we will have a 95% confidence level. We can assess that the variation between Neoventa's and Huntleigh's version is greater than the variance within Huntleigh's original and delayed version. That indicates that either does Huntleigh's algorithm define the parameter "signal loss" in another manner or that the error algorithm removes less or more data compared to Neoventa's version.

One difference that I discovered using simulated signals was that some heart beat intervals were rounded down to nearest integer in the transformation from milliseconds to beats per minute in Huntleigh's program which is remarkably when it is more common to round up. In Neoventa's program I always rounded up.

One example of how the signal loss is estimated in Huntleigh is given below in Figure 12. According to the program the signal loss at 20 minutes was 14%, but it does not exactly correspond to the drawn figure. Where the black trace becomes green it indicates interpolated values and orange means that the samples were removed as unreliable data. If the amount of time when the trace is green is added

it seems to exceed 14% error loss. Therefore I have trouble relying in their estimated signal loss-parameter.

The signal loss depends upon the signal pre-processing before the analysis, which is removing noise and unreliable samples. In order to adjust Neoventa's version to become more similar to Huntleigh's a method could be to remove more samples that do not stick to the basal heart rate. I have noticed that some decelerations are identified by my program and not in the original because their error algorithm removes more data. These removed samples affect of course all the evaluated parameters as well.

However I do think that Huntleigh's algorithm sometimes removes too much data from the analysis. Since it is said that it detects for example arrhythmia, I question whether or not it is correct to remove almost everything that does not stick to the averaged signal. One drawback could be that important data is actually removed.
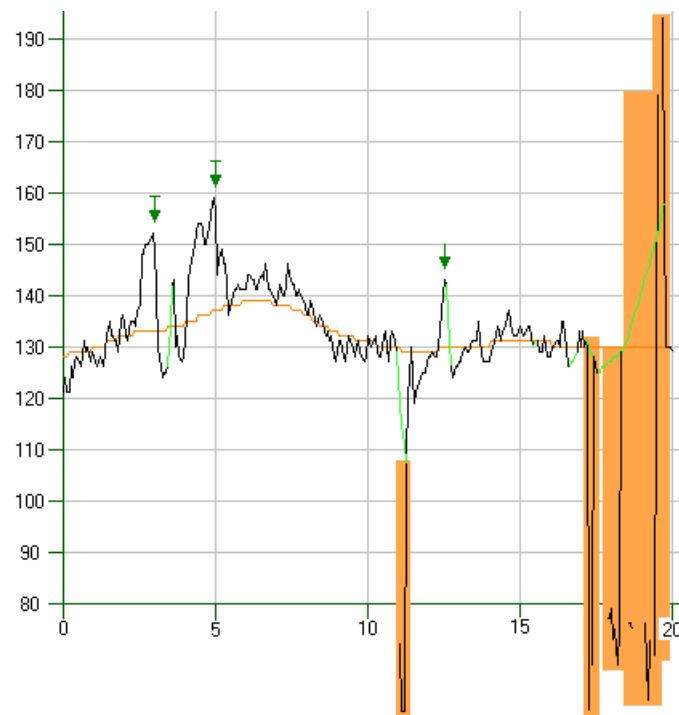
Figure 12. An example of how the signal loss is measured and calculated in the original Dawes/Redman algorithm, the x-axis is the time [min] and the y-axis is the heart rate [bpm].

6.2 Baseline

In my baseline fitting algorithm I used a Butterworth low-pass filter during the filtration of the averaged signal. I tried to match it to the given data from Table 1 but it was difficult to make it match perfectly. My constructed filter was of the first order and had a normalized cut-off frequency at 0.0194 Hz. The constructed low-pass filter with its frequency response is displayed below in Figure 13 which is marked blue. The frequency response can be compare to the red curve which illustrates the filter used in the original Dawes/Redman algorithm.

The maximum and minimum values of the baseline during the ten first minutes of the analysis were estimated and compared against each corresponding trace. As we can assess from Figure 14 and Figure 15 there is a distinction between Neoventa's and Huntleigh's version. A possible explanation is that I have noticed that Huntleigh's baseline always tends to start at an average value for the analysed trace, while my algorithm starts with the first low pass-filtered fetal heart rate. That means that if the evaluation for example starts during an acceleration or deceleration my baseline variation tends to be greater in one direction comparing to Huntleigh's.

I have also noticed that the baseline after the five first minutes in my program corresponds well to the one in Huntleigh's. Many of the analysed parameters in the Dawes/Redman criteria are depending upon the baseline and therefore it is important to have a good estimation. The estimation of the baseline is a question of interpretation, and even among doctors and midwives with many years of experience their baseline-fit within the same trace varies. The long-term variation, accelerations and decelerations use the fitted baseline for estimating their values which make them very sensitive to changes. It is not a bad result that the baseline in Neoventa's algorithm varies 1.5 bpm more than the Huntleigh's itself during the ten first minutes. Since it is not a big variance and relatively easy to modify.
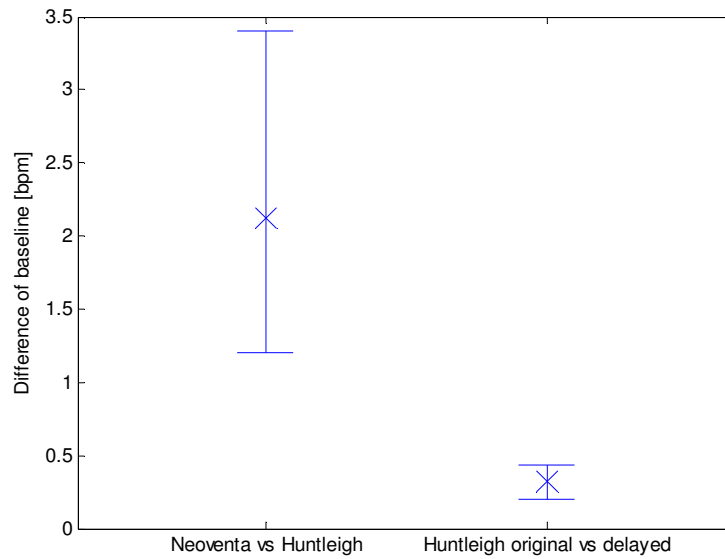
Figure 14. The graph shows the differences between the compared traces in baseline maximum during the first ten minutes, the x-axis is the two comparisons and the y-axis is the differences in baseline [bpm].
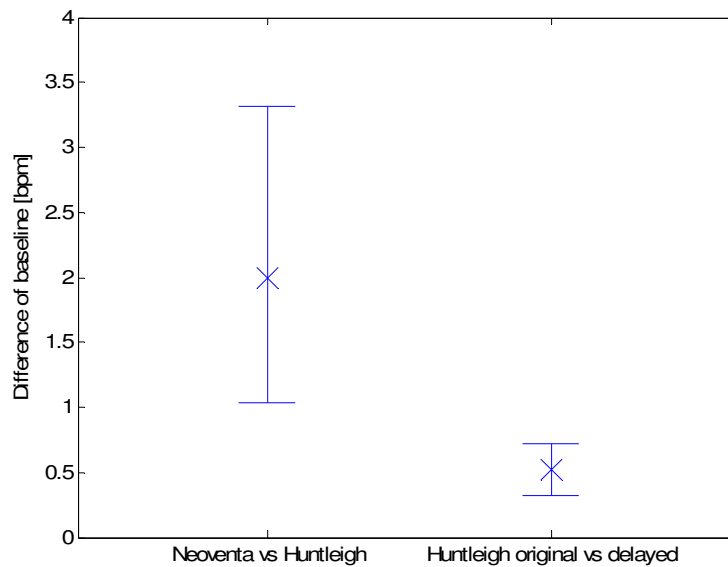
Figure15. The picture indicates how the minimum baseline during the first ten minutes differs from the compared trace, the x-axis is the two comparisons and the y-axis is the differences in baseline [bpm].

One potential enhancement of Neoventa's version, to make it more similar to Huntleigh's, could be to replace the first fetal heart rate samples with the basal heart rate or an averaged value of the whole signal before the low pass-filtration. Then the big variations of the baseline would not be included, if the trace would start during a deceleration or acceleration. Figure 16 is an example where the orange trace is the baseline in Huntleigh's version, the blue is the old baseline in my algorithm and the green trace is the new baseline. For the new baseline I replaced the 5 first samples, before low-pass filtering, with an average of the whole signal. This replacement is just performed for the baseline fitting. A results is that the new baseline follows Huntleigh's baseline much better than before.
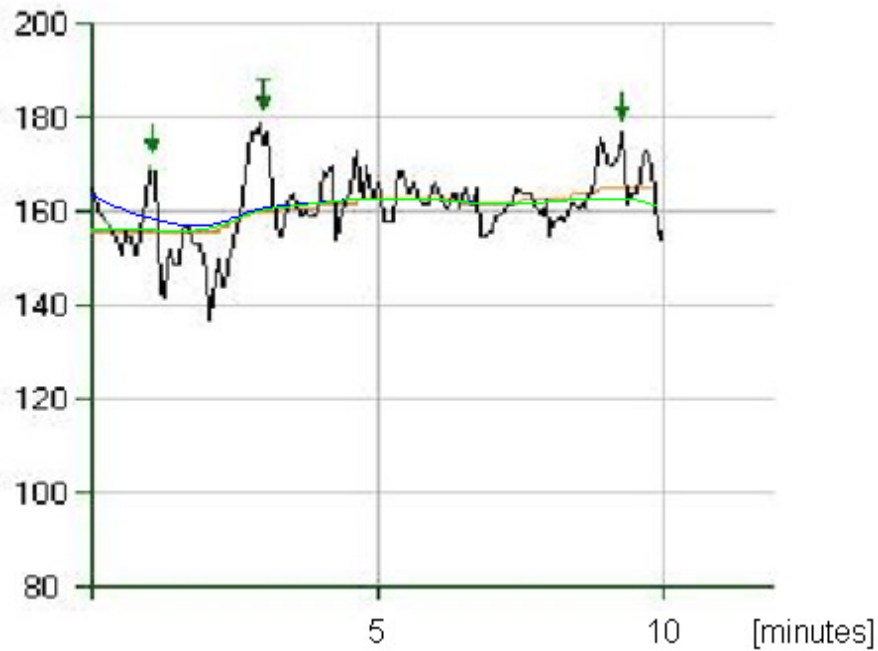
Figur 16. An example of a bpm-trace where the first samples are set to the average of the signal before filtering to the baseline. The new baseline is marked green and the used baseline in the algorithm is marked red.

6.3 Accelerations and decelerations

To be able to validate how many accelerations that were misclassified I counted the differences in found accelerations between the two compared traces. Figure 17 shows the results for the comparison between Neoventa's and Huntleigh's version respectively Huntleigh's and a delayed analysis. Both two evaluations examined 25 traces each. It can be seen from the diagram that Huntleigh's algorithm itself has less variance compared to the variation between Neoventa's and Huntleigh's version. Taking into account that the validation between Neoventa's and Huntleigh's version identified 126 accelerations and just 85 in the other validation, a conclusion is that there is not a big difference between the two versions.
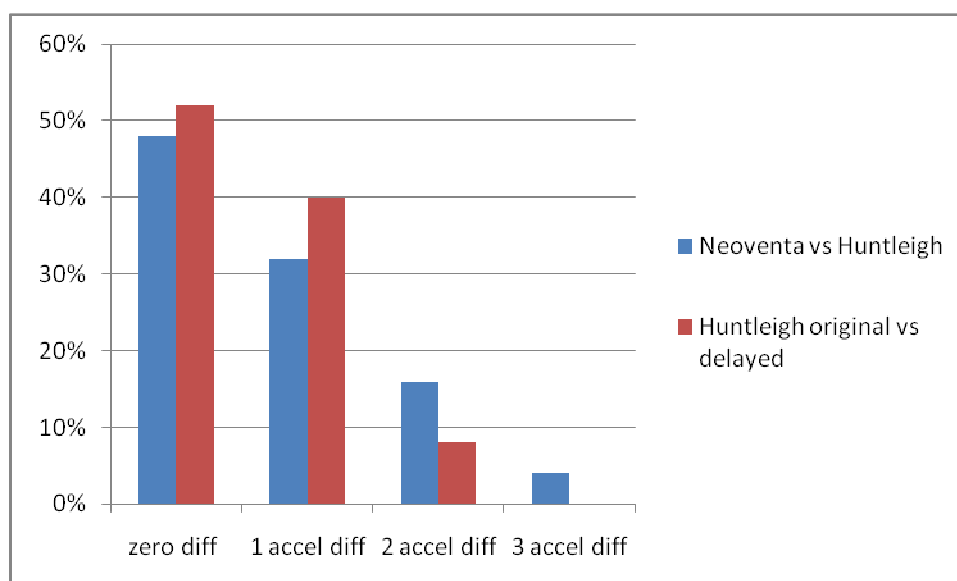
Figure 17. It shows the results from the validation of the implemented algorithm in found accelerations. X-axis represents number of traces [percent] and y-axis the difference in found decelerations between the algorithms.

23

I have noticed that Huntleigh's program may discriminate small accelerations lying very close to each other, which my program is not able to detect because of the averaging over 1/16 minutes. I draw the conclusion that maybe Huntleigh's program does not use averaged data when counting the accelerations, even though it is said in the articles that the analysis uses average data [7].

The differences in identified decelerations between Neoventa's and Huntleigh's version were also documented. The result showed that Neoventa's version detected one extra deceleration in three traces and one less in two traces. Huntleigh's algorithm detected one extra deceleration in one trace when the recording was delayed 10 seconds. The variation between the Neoventa's and Huntleigh's comparison is greater than the variation for Huntleigh's program itself. But taking into account that the total amount of decelerations was nearly twice as big, 19 vs 11 found decelerations, during the evaluation between Neoventa's and Huntleigh's version the result is accepted.

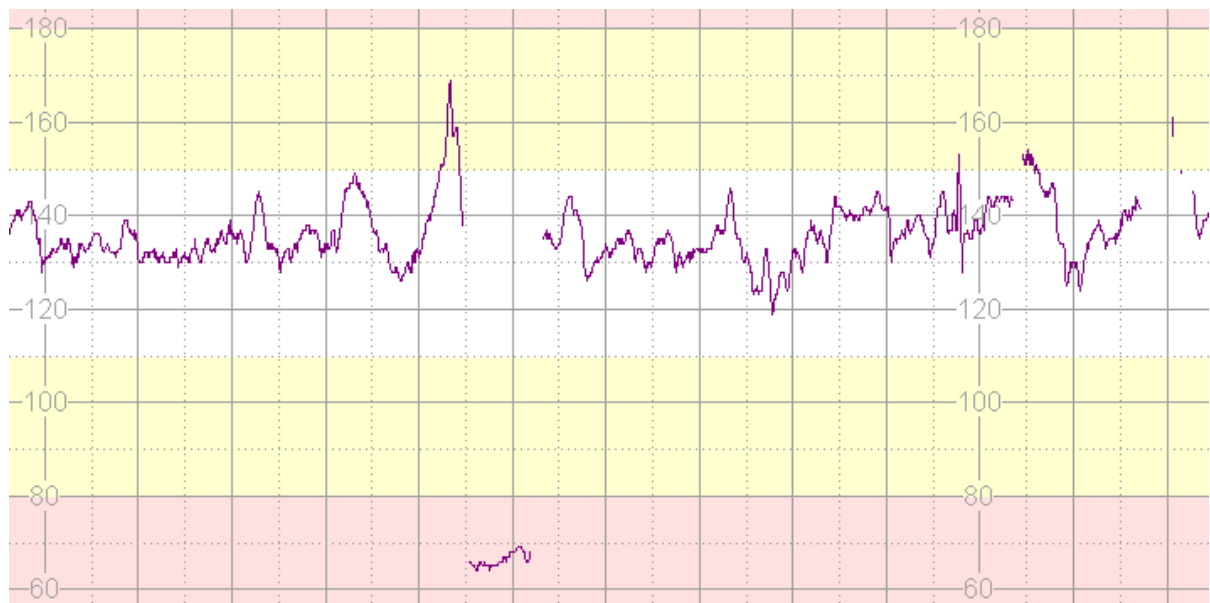**Figure 18. An example of a misclassified deceleration by Neoventa's algorithm. In the graph the x-axis is the time [min] and the y-axis is the heart rate [bpm].**

An example of a misclassified deceleration is illustrated in Figure 18. My algorithm interpreted the decrease in heart rate as a deceleration while Huntleigh's program removed all the samples of a heart rate around 70 bpm as signal loss. Since the samples are at the half frequency of the other surrounding samples it is very likely that they were interpreted wrong by the ultrasound algorithm. One enhancement would be to implement such an error algorithm.

6.4 Long-term variation

In Figure 19 it is shown how many minutes of high episodes the two compared traces differed from each other. The median for Neoventa's compared with Huntleigh's version was about 1.4 minutes while Huntleigh's compared with a delayed version differed 0.75 minute. We can also deduce from the graph that the 2.5[th] and 97.5[th] percentiles overlap. The high variation depends upon the long-term variation (LTV) and the LTV depends on the baseline. Possible reasons for the differences are variations of the baseline or of the error algorithm between Neoventa's and Huntleigh's version. If the error algorithm would remove more or less samples this changes the LTV since it measures the maximum and minimum of the trace compare to the baseline for each minute.
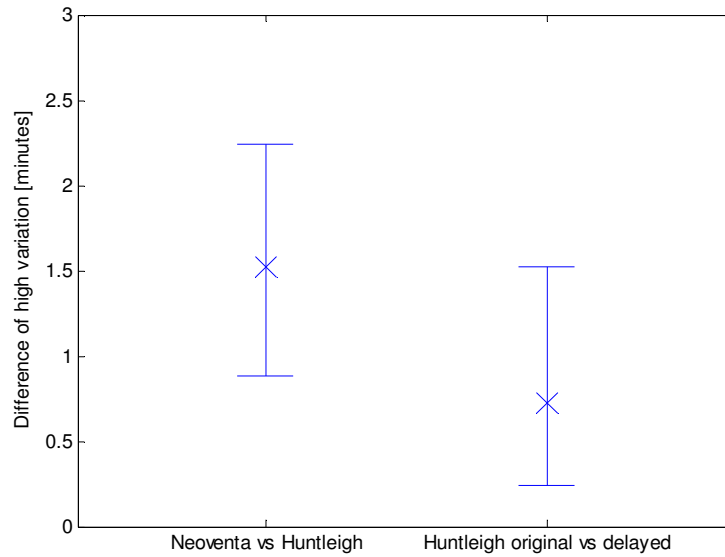
6.5 Short-term variation

In Figure 20 the short-term variation (STV) in both evaluations have a variation with a median around 0.4 milliseconds. The figure shows also that the 97.5th and 2.5th percentiles are overlapping which is reassuring and shows that Neoventa's version estimates STV well compare to Huntleigh's version.
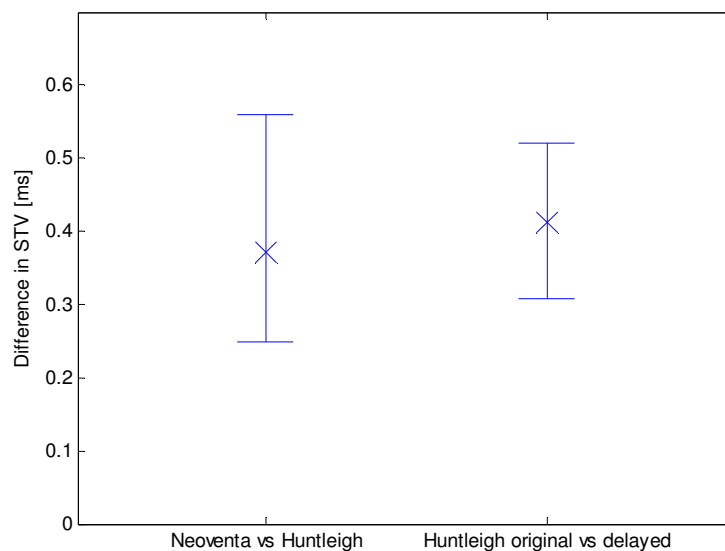
6.6 Basal heart rate

The result in Figure 21 shows that the basal heart rate had almost the same accuracy both in Neoventa's and Huntleigh's version. Huntleigh's delayed version had a median around 1.25 bpm and the comparison between Neoventa's and Huntleigh's version had a median around 0.75 bpm.

According to the figure the confidence interval is greater in Huntleigh's case compared to Neoventa's.
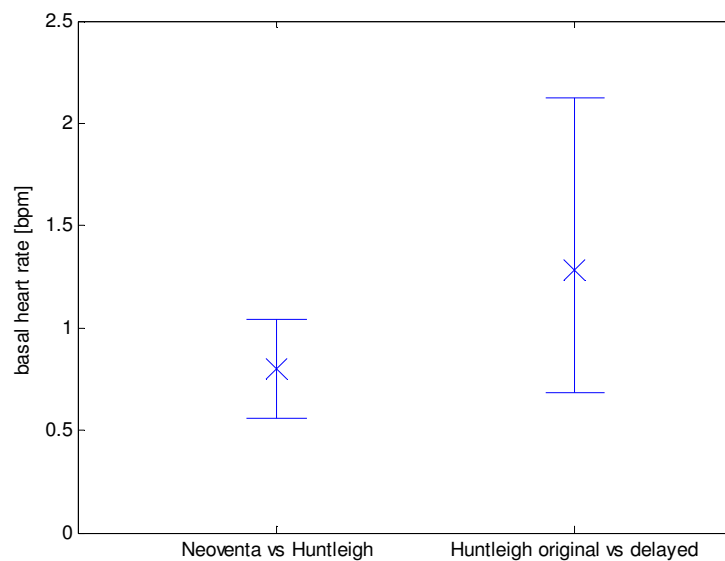
Figure 21. The graph shows how the basal heart rate differs between the compared versions and the calculated cetiles, 2.5th and 97.5th, the x-axis is the two comparisons and the y-axis is the differences in basal heart rate [bpm].

6.7 Dawes/Redman criteria

After ten minutes the first analysis is performed and then repeated every second minute. If criteria are met the trace is approved. However if more data needs to be collected in order to assess the fetal health, the two programs will announce criteria not met and reasons why. Figure 22 evaluates differences in outcomes, criteria met or not, between the two compared traces in both evaluations. We can assess from this diagram that Neoventa's version seems to have the same precision as Huntleigh's version.
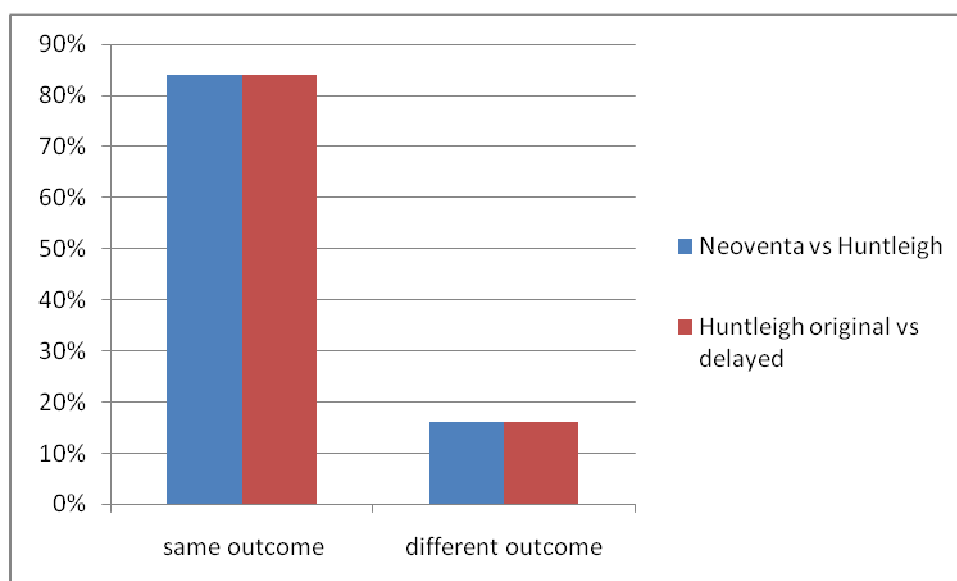
Figure 22. The diagram shows however the versions had the same outcome, criteria met or not, in the evaluation. X-axis represents the number of traces [percent] and y-axis a comparison between the algorithms in outcomes.

26

Possible reasons why my version differed from Huntleigh's were that Neoventa's version found the baseline fit uncertain and a potential high frequency sinusoidal pattern. The ratio between LTV and STV is estimated during the evaluation of a potential high or low frequency sinusoidal pattern. If the ratio is below 0.1380 or above 0.2100 the user is warned. A baseline fit is uncertain if the ration is below 0.1140. From Figure 20 which showed that the STV for Neoventa's version was within the same confidence interval as the STV from Huntleigh's version we can conclude that it could possibly be the LTV that differs between the two versions. The LTV itself depends upon the baseline and the error algorithm which removes unreliable samples.

I realized that by just moving the start a few seconds in Huntleigh's algorithm all the parameters changed and sometimes also the outcome (criteria met or not). It is remarkably that the program is so sensitive to changes of the starting time because basically the same data is examined. I have noticed that when accelerations are misclassified in my program it is often on the verge of being right classified. One reason why the program is sensitive is because it uses averaged heart beats over 1/16 minute. The program will then obtain different averaged data just by moving the start point a few samples.

Another example of that the Huntleigh's program is very sensitive and sometimes incalculable is that the parameters change slightly after having been stored in the program, see one example below in Figure 23. That is confusing and I cannot find any documentation where it is described what this change might depend on.

Real time data		Stored data	
Results	14	Results	14
Signal loss (%)	0	Signal loss (%)	1
Contractions	6	Contractions	5
Movements	0	Movements	0
Basal Heart Rate (bpm)	141	Basal Heart Rate (bpm)	142
Accelerations	4	Accelerations	4
Decelerations	0	Decelerations	0
High minutes (mins)	7	High minutes (mins)	7
Most lost beats	0	Most lost beats	0
Short Term Variation (ms)	9,1	Short Term Variation (ms)	10,0

Figur 23. An example of how the parameters change when the analysis is saved.

The minimum amount of time for a trace to be reassuring is ten minutes. However it is stated in the article that a sinusoidal component can be, at the earliest, recognized after 20 minutes [7]. That

means that if the criteria are met after 10 minutes it cannot totally be reassuring since it has not examined the peak-to-peak distribution where a sinusoidal pattern may be identified.

Another remark is that after a test has met the criteria the monitoring continues and after each second minute a new analysis of the trace is performed. That means that a trace that is reassuring after 10 minutes might not meet the criteria after another 2 minutes. This could make the user insecure whether it is safe to stop the evaluation, even though the criteria are met.

7. Conclusion and future works

There are currently no patents in Europe or US protecting the Dawes/Redman algorithm which could hinder an implementation in STAN S31. During a wide literature study enough information concerning the Dawes/Redman algorithm was found to enable implementation of an analysis founded on the Dawes/Redman criteria.

A conclusion from the validation of the implemented algorithm is that the basics of the algorithm seem to accord with the original Dawes/Redman algorithm. The short-term variation, basal heart rate, accelerations, decelerations and minutes of high episodes have almost the same variance as the original program. However, there is a difference between Neoventa's and Huntleigh's version during the fitting of the baseline. Our algorithm tends to have a slightly greater variance of the baseline during the ten first minutes, comparing to the original.

One improvement of the implemented algorithm could be to always make the baseline start at the basal heart rate which I believe the original algorithm does. This would not just improve the results of the variation of the baseline during the first ten minutes but it would also improve long-term variation and detection of accelerations and decelerations. The pre-processing of the signal also needs to be tougher in Neoventa's algorithm and remove more unreliable samples in order to resemble more to the original version, even though important data may be deleted.

In the present situation I would say that the two algorithms are not identical but very similar to each other. The basics of the implemented program is working as it should and by applying just some small modifications, as mentioned above, is could be used with the same purpose as the original Dawes/Redman analysis.

8. References

[1] http://www.sonicaidfetalcare.com/downloads/application_uk.pdf, 10/6-2010

[2] http://www.babycentre.co.uk/pregnancy/complications/placental-abruption/, 1/10-2010

[3] http://www.preeclampsia.org/, 1/10-2010

[4] ACOG technical bulletin: Fetal heart rate patterns: monitoring, interpretation, and management.

International Journal of Gynecology & Obstetrics, 51, 1995, p 65-74

[5] Fosterövervakning och ST-analys, Neoventa Medical AB, ISBN: 978-91-85927-06-7, 2008

[6] Ingemar Ingemarsson, Eva Ingemarsson: Fosterövervakning med CTG, Studentlitteratur, Denmark ISBN: 91-44-04524-7, 2006

 [7] James Pardey, Mary Moulden, Christopher Redman: A computer system for the numerical analysis of nonstress tests. Am I Obstet Gynecol, vol 186 nr 5, 2002

[8] patents.com, 1/6-2010

[9] espacenet.com, 1/6-2010

[10] google.com/patents, 1/6-2010

[11] www.scirus.com, 7/6-2010

[12] http://chans.lib.chalmers.se/, 7/6-2010

[13] Abdelhak Zoubir, Robert Iskander: Bootstrap Methods and Applications: A Tutorial for the Signal Processing Practitioner. *Signal Processing Magazine*, 24(4), p. 10-19, 2007

[14] Matthew James Thomas: Monitoring electrical muscular activity, Application number: 10/549,732, Filing date: Mar 16, 2004

[15] Duong Nguyen: Fetal outcome predictor and monitoring system, Patent number: 6254537 Issue date: Jul 3, 2001

[16] SonicaidFetalCare: Objective Assessment of Fetal Wellbeing, User guide, Oxford Instruments Medical, Mar 2003

[17] P. Street, G.S. Dawes, M. Moulden, C.W.G. Redman: Short-term variation in abnormal antenatal fetal heart rate records. *American journal of obstetrics and gynecology,* vol 165, p. 515-523, 1991

[18] Dawes GS, Visser GH, Goodman JD, Redman CW.: Numerical analysis of the human fetal heart rate: the quality of ultrasound records*. American journal of obstetrics and gynecology*, vol 141 (1), p.43-52, Sep 1981

[19] Dawes GS, CRS Houghton, CWD Redman: Baseline in human fetal heart rate records. *British journal of obstetrics and gynaecology*, vol 89 (4), p.270-275, Apr 1982

[20] Dawes GS, Redman CW: Improvements in the registration and analysis of fetal heart rate records at the bedside. *British journal of obstetrics and gynaecology*, vol 92, p. 317-325, Apr 1985

[21] Geoffrey Dawes, Mary Moulden, Christopher Redman: System 8000: Computerized antenatal FHR analysis. *J. Perinat. Med*, vol 19, p. 47-51, 1991

9. Appendix

The main Matlab program:

```
clear
clc
close all
s = readstn('FILE.stf');
age=41;
move=s.marker_t;
signal = s.us1_rr;

%change for us1 and us2
t = s.us1_t;
orgs=signal;
orgt=t;

start=22;
stop=10;
signal=signal(start:end-stop);
signal_hr=60000./signal;
t=t(start:end-stop);

t=t-t(1);
move=move-t(1);

figure(11)
b11=plot(t/60000, 60000./signal);
hold on
axis tight
set(b11, 'Color','blue')
grid on
xlabel('time [minutes]')
ylabel('heart rate [bpm]')
title('Original signal')
hold on

%filter away the values that are not inte the range of 65%-175% of the
%average of the two last signals
[rem, time] = remove(signal, t);

%the signal is averaged over 1/16 minute
[datareduc, timereduc, loss, databpm] =  average(rem, time);

%remove samples that jump more than 75ms from the average of the entire signal and then do an instant return to
%the past value
[datareduc, datareduc_hr, timereduc, loss] = removejumps(datareduc, databpm, timereduc, loss);

%find the baseline
[newbase,newbase2, peak] = baseline_new(datareduc, datareduc_hr, timereduc);

%starts the evaluation here, the baseline is then fitted
% start=1;
newbase_hr= 60000./newbase;
newbase_hr2= 60000./newbase2;

%plot the signal and the baseline in Hz
figure(3)
signal=plot(timereduc/60000, datareduc_hr);

hold on

b3 = plot(timereduc/60000, newbase_hr2);
b2 = plot(timereduc/60000, newbase_hr);
grid on
axis tight
age=41;
```

```matlab
set(b3, 'Color','green')
set(b2, 'Color','red')
set(signal, 'Color', 'blue')
title('Averaged heart beats over 1/16 min with the baseline ')
xlabel('time [minutes]')
ylabel('heart rate [bpm]')


[datatemp_hr] = linear(datareduc_hr, time);

%calculate accelerations
[totacc, acc10, acc15, rateacc] = accel_hr(datatemp_hr, newbase_hr, timereduc, loss);

%calculate decelerations
[totdec, dec10, dec20, decels, lostbeat] = decel_hr(datatemp_hr, newbase_hr, timereduc, loss);

dec21=0;
dec100=0;
for(i=1:length(lostbeat))
   if(lostbeat(i)>20 && lostbeat(i)<=100)
      dec21=dec21+1;
   end
   if(lostbeat(i)>100)
      dec100=dec100+1;
   end
end

if(timereduc(length(timereduc))/60000<30 && dec21~=0)
   disp('EVAL: the recording was less than 30 min and there were decelerations greater than 20 lost beats');
elseif(timereduc(length(timereduc))/60000>=30 && dec21>1)
   disp('EVAL: the recording was greater than 30 min and there were more than one deceleration greater than 20 lost beats');
end

if(dec100>0)
   disp('EVAL: there were decelerations greater than 100 lost beats')
end

if(isempty(move))
   move(1)=0;
end

%detection of high and low variation
[truelow, truehigh, ltv, cent3ok, cent10ok, rate, ratel, rateh, bpmhigh, bpmlow] = ltv(datareduc, datareduc_hr, newbase, newbase_hr, loss, timereduc, decels, move, age);

%counts how many minutes there is high variation and low variation
minlow=0;
minhigh=0;
minute=1:length(truehigh);
for(i=1:length(truehigh))
  if(truelow(i)==1)
      if(minlow==0)
         disp('Minutes of low variation:')
         minute
         truelow
      end
      minlow=minlow+1;
   end
end

%look for an episode of high variation
found=0;
for(i=1:length(truehigh))
   if(truehigh(i)==1)
      if(found==0)
         disp('Minutes of high variation:')
         minute
```

```
            truehigh
        end
        found=1;
        minhigh=minhigh+1;
    end
end
if(found==0)
    disp('EVAL: no episode of high variation was found')
end


%calculate shortterm variation
[stv] = stv(datareduc, loss, decels);

if(stv<=3)
    disp('EVAL: STV is less then 3ms')
elseif(stv<4.5 && cent3ok==0)
    disp('EVAL: STV is less then 4.5ms and LTV is less than the third centile for gestational age')
end

if(move(1)~=0)
    fetalrate=length(move)*1000*3600/timereduc(end);
else
    fetalrate=0;
end
if(totacc==0)
    if(fetalrate<20 || cent10ok==0)
        disp('EVAL: There was no accelerations, nor fetal movement rate above 20/h and a LTV below tenth centil')
    end
end

if(move(1)==0 && totacc<3)
    disp('EVAL: there was no fetal movement and the total acceleration was below 3')
end

%calculate basal heart rate
[basehr] = basalhr(datareduc, truelow, peak);

if(time(length(time))/60000<30 && (basehr<116 || basehr>161))
    disp('EVAL: the recording time is less than 30 min and the basal heart rate is not within the inteval 116-160bpm')
end

%look for sinuspattern
[lowfreq, highfreq, mean] = sinus(ltv, stv, datareduc, timereduc);

tot=0;
totnum=0;
arti=0;
for(i=1:length(loss))
    if(loss(i)==100)
        totnum=totnum+1;
    end

    tot=tot+loss(i);
    if(length(loss)-i<16)
        if(loss(i)==100)
            arti=arti+1;
        end
    end
end
arti=arti/16*100;
tot
tot=totnum/length(loss)*100;

if(arti>=50)
    disp('EVAL: possible artifact at the end, do not stop recording')
end
```

```
if(decels(end)==1 && lostbeat(end)>=50)
    disp('EVAL: large deceleration at the end of the trace, do not stop recoring')
end


%must be high variation with more than 0.5 movements per minute
if(ltv>8.7719*stv || ltv<3.876*stv)
    if(stv<=5 || basehr<120 || tot>=30 || rate<0.5)
        disp('EVAL: the trace must have: STV>5ms, basal heart rate>120 bpm, signal loss<30 and rate>0.5')
        disp('EVAL: LTV is not within 3 SDs')
    end
end


%find max and min of baseline during the first 10 min
if(length(newbase_hr)>=160)
temp_baseline=newbase_hr(1:160);
maxbaseline= max(temp_baseline);
minbaseline= min(temp_baseline);
else
    minbaseline=NaN;
    maxbaseline=NaN;
end



%creat a file for evaluation
fid = fopen('Eval.txt', 'w');

fprintf(fid, '1. Signal loss (%%):\t\t\t\t%0.2f\n', tot);
fprintf(fid, '2. Contraction peaks\n');
fprintf(fid, '3. Fetal movements (per hour)\t\t\t%0.0f\n', fetalrate);
fprintf(fid, '   per min in High %0.0f per minute in Low %0.0f\n', ratel, rateh);
fprintf(fid, '4. Basal heart rate (bpm):\t\t\t%0.0f\n', basehr);
fprintf(fid, '5. Accelerations >10bpm and 15s:\t\t%0.0f\n', acc10);
fprintf(fid, '                >15bpm and 15s:\t\t%0.0f\n', acc15);
fprintf(fid, '   -----------------------------------------\n');
fprintf(fid, '   Total accelerations:\t\t\t\t%0.0f\n', totacc);
fprintf(fid, '\n');
fprintf(fid, '6. Decelerations >10bpm and 60s:\t\t%0.0f\n', dec10);
fprintf(fid, '                >20bpm and 30s:\t\t%0.0f\n', dec20);
fprintf(fid, '   -----------------------------------------\n');
fprintf(fid, '   Total decelerations:\t\t\t\t%0.0f\n', totdec);
fprintf(fid, '\n');
fprintf(fid, '7. High episodes (min):\t\t\t\t%0.0f (%0.2f bpm)\n', minhigh, bpmhigh);
fprintf(fid, '   Low episodes (min):\t\t\t\t%0.0f (%0.2f bpm)\n', minlow, bpmlow);
fprintf(fid, '8. Short-term variation (ms):\t\t\t%0.2f\n', stv);
fprintf(fid, '9. Long-term variation (ms):\t\t\t%0.0f \n', ltv);
fprintf(fid, '10. Max/min of baseline after 10 min:\t%0.0f / %0.0f (bpm)\n', maxbaseline, minbaseline);
if(length(lostbeat)>0)
    fprintf(fid, '11. Lost beats of decelerations:\t\t%');
    for(i=1:length(lostbeat))
        fprintf(fid, '%0.1f ', lostbeat(i));
    end
    fprintf(fid, '\n');
end
fprintf(fid, '\n');
fprintf(fid, '   Minute:\t\t\t%');
for(i=1:length(truehigh))
    fprintf(fid, '%0.0f\t', i);
end
fprintf(fid, '\n');
fprintf(fid, '12. high variation:\t%');
for(i=1:length(truehigh))
    fprintf(fid, '%0.0f\t', truehigh(i));
end
fprintf(fid, '\n');
```

```matlab
fprintf(fid, '12. low variation:\t%');
for(i=1:length(truelow))
    fprintf(fid, '%0.0f\t', truelow(i));
end
fprintf(fid, '\n');

%close the file
fclose(fid);


Functions:
function [datafilt, time] = remove(data, time)
remove=0;
datafilt = zeros(1,length(data));
%filter away the values that are not in the range of 65%-175% of the
%average of the two last signals
block=-1;
first=0;
second=0;
for(i=1:length(data))
    %if it is a nan-value it should remain so
    if(isnan(data(i)))
        datafilt(i)=NaN;
    elseif(second~=0 && block<0)
        avg=(first+second)/2;
        %if the value is not within 65%-175% of the two last valid samples
        %it and its 2 next samples should be set to nan
        if ((0.65*avg>data(i) || 1.75*avg<data(i)) && i+2<=length(datafilt))
            block=2;
            datafilt(i)=NaN;
            data(i)=NaN;
            %note that the next two values will still be in the evaluation
            %else we will take away big parts of the signal
            datafilt(i+1)=NaN;
            datafilt(i+2)=NaN;
            remove=remove+1;
        else
            %if the sample is ok it is kept as it is
            datafilt(i)=data(i);
        end
    elseif(second==0)
        %for the first two samples, the values are taken without
        %any evaluation
        datafilt(i)=data(i);
    end

    block=block-1;
    %change the pointer so they alwayas point at thw two last valid samples
    if(isfinite(data(i)))
        second=first;
        first=data(i);
    end
end
function [datareduc, timereduc, loss, databpm] = average(data, time)

%the signal is averaged over 1/16 minute
datareduc=zeros(1,10);
timereduc=zeros(1,10);
index=1;
found=0;
avg=0;
samp=0;
%computes the signal loss for later evaluation of the ltv and stv, the loss
%is stored in percent
loss=zeros(1, length(datareduc));
%at which the next sample should start at
```

34

```matlab
stop=time(1)+3750;

avgtemp=0;
databpm=zeros(1,10);

first_time=1;
for i=1:length(data)
    first_time=1;
    %is used in computing the signal loss
    samp=samp+1;

    %if the value is finite it is added to the average, else it is added to
    %the loss parameter
    if (isfinite(data(i)))
        avg=avg+data(i);
        %in bpm instead
        avgtemp=avgtemp+60000/data(i);
        found=found+1;
    else
        loss(index)=loss(index)+1;
    end

    %if also the next sample is less then the length of the array it will be checked,
    %if the next time is greater or equal to the next valid time periode it
    %should be evaluated and stored in an array
    if(i+1<=length(time))
        %a new sample should be added since the time has passed
        if(time(i+1)>=stop)
            %if it passed to the following time
            if(time(i+1)<stop+3750)
                timereduc(index)=stop-3750;
                stop=stop+3750;
                %if no valid samples were found it should be put to nan, else
                %an average is made for current time period.
                if(found~=0)
                    datareduc(index)=avg/found;
                    databpm(index)=avgtemp/found;
                    loss(index)=0;
                else
                    datareduc(index)=NaN;
                    databpm(index)=NaN;
                    loss(index)=100;
                end
                index=index+1;
            else
                %if there is a missing sample so it passes the next time limit, it should be filled out with
                %NaN and 100% signal loss
                time(i)
                time(i+1)
                time(i+1)-time(i)
                stop
                while(time(i+1)>stop)
                    timereduc(index)=stop-3750;
                    if(first_time==1 && isfinite(data(i)))
                        loss(index)=0;
                        datareduc(index)=data(i);
                        databpm(index)=60000/data(i);
                        first_time=0;
                    else
                        loss(index)=100;
                        datareduc(index)=NaN;
                        databpm(index)=NaN;
                        index
                    end
                    stop=stop+3750;
                    index=index+1;
```

```matlab
                end
            end
            %initiate the next value for the signal loss
            loss(index)=0;
            avg=0;
            avgtemp=0;
            found=0;
            samp=0;
        end
    else
        %if it is the last average of the array
        if(avg~=0)
            datareduc(index)=avg/found;
            timereduc(index)=time(i);
            loss(index)=loss(index)/samp*100;
            databpm(index)=avgtemp/found;
        else
            %if no value was found the last value is deleated
            datareduc=datareduc(1:index-1);
            timereduc=timereduc(1:index-1);
            loss=loss(1:index-1);
            databpm=databpm(1:index-1);
        end
    end
end

function [data, data_hr, time, loss] = removejumps(data, data_hr, time, loss)

tot=0;
avg=0;
found=0;

%set values to zero that differs more than 75ms from the mean value and
%saves how many samples that are removed in an variable
removed=0;
for i=1:length(data)-1
    if(isfinite(data(i)))
        found=found+1;
        tot=tot+data(i);
        avg=tot/found;

        if(data(i)<avg-75 || data(i)>avg+75)
            %look if it returns to the last sampel value +/-5 ms depending on the resolution
            b=1;

            while(b<3 && i+b<=length(data))
                if(isfinite(data(i+b)))
                    if ((avg-5<data(i+b) && avg+5>data(i+b)) || abs(data(i+b)-data(i))>300)
                        for k=i:i+b-1
                            data(k)=NaN;
                            removed=removed+1;
                            loss(k)=100;
                        end
                        b=5;

                    end
                else
                    for k=i:i+b-1
                        data(k)=NaN;
                        removed=removed+1;
                        loss(k)=100;
                    end
                    b=5;
                end
                b=b+1;
            end
```

```
            end
        end
end
tot=0;
avg=0;
found=0;
%set values to zero that differs more than 75ms from the mean value and
%saves have many samples that are removed in an variable
removed=0;
for i=1:length(data_hr)-1
    if(isfinite(data_hr(i)))
        found=found+1;
        tot=tot+60000/data_hr(i);
        avg=tot/found;

        if(60000/data_hr(i)<avg-75 || 60000/data_hr(i)>avg+75)
            b=1;
            while(b<3 && i+b<=length(data_hr))
                if(isfinite(data_hr(i+b)))
                    if ((avg-5<60000/data_hr(i+b) && avg+5>60000/data_hr(i+b)) || abs(60000/data_hr(i+b)-60000/data_hr(i))>300)
                        for k=i:i+b-1
                            data_hr(k)=NaN;
                            removed=removed+1;
                            loss(k)=100;
                        end
                        b=5;
                    end
                else
                    for k=i:i+b-1
                        data_hr(k)=NaN;
                        removed=removed+1;
                        loss(k)=100;
                    end
                end
                b=b+1;
            end
        end
    end
end
```