

---

## Adaptive Delta Modulation Simulation and Analysis Using MatLab

Djadjat Sudaradjat<sup>1</sup>, Andi Rosano<sup>2</sup>

<sup>1,2</sup> Universitas Bina Sarana Informatika, Jakarta, Indonesia

---

### ARTICLE INFORMATION

*Artikel History:*

Received: July 20, 2023

Revised: August 24, 2023

Accepted: August 25, 2023

---

*Keyword:*

ADM  
Speech Processing  
Speech Compression

### ABSTRACT

*Adaptive Delta Modulation (ADM) is an efficient signal coding method for handling rapid changes in analog signals compared to conventional Delta Modulation (DM) methods. This research aims to simulate and analyze the ADM and DM methods using Matlab software. In this study, we implemented an adaptive ADM algorithm to generate accurately and efficiently encoded signals. Additionally, we analyzed and compared the performance of the ADM method with the DM method in handling dynamic changes in analog signals, considering parameters such as the number of coding bits, sampling rate, and compression level. Through in-depth analysis, we evaluated the advantages and limitations of the ADM method compared to conventional DM methods in real-world applications, especially in the field of telecommunications and the processing of analog signals into digital signals. The simulation results demonstrate the superior performance of ADM compared to conventional DM. By adaptively adjusting the delta level, the system can reduce distortion and generated noise. The calculation results show that the sum\_of\_squared\_error for DM is 0.7690, while for ADM it is 0.3470. Specifically, this paper focuses on simulating and comparing ADM and DM methods in terms of handling high pulse delta changes to overcome slope overload. The findings of this research can provide valuable insights for further development and improvement of the ADM method, as well as establish a foundation for practical applications in the computer field.*

---

### Corresponding Author:

Djadjat Sudaradjat,  
Electrical Engineering Department,  
Universitas Bina Sarana Informatika,  
Jl. Kramat Raya No. 98, Jakarta, Indonesia,  
Email: [djadjat.dsj@bsi.ac.id](mailto:djadjat.dsj@bsi.ac.id)

---

### INTRODUCTION

By simulating the ADM system, researchers will find it easier to search for the optimal pulse height value in the ADM system. The system for converting analog signals to digital signals has become a necessity, considering that the tools currently used are digital computers. One of the analog-to-digital conversion systems is the delta modulation system. In delta modulation, only one bit is used to encode each change in the input signal, where a rising signal is encoded into a one bit and a falling signal is encoded into a zero bit (Mashhadi, Malekmohammadi, & Marvasti, 2019). This can save the bit rate which, when

transmitted over the transmission channel, can save the transmission bandwidth, and when stored in computer storage media, can save storage space. However, if the slope of the signal rises and falls sharply, the delta modulation system cannot follow these signal changes, which is called slope overload (Sudaradjat, 2019). To overcome this problem, an adaptive delta modulation system was developed that has variable or adaptive pulse height for both one and zero bits. In this paper, a comparison between the delta modulation system and the adaptive delta modulation system using Matlab application will be simulated and analyzed. Adaptive Delta Modulation (ADM) is a pulse modulation

---

DOI: <https://doi.org/10.31294/paradigma.v25i2.>



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)

technique used in digital communication systems to transmit analog signals over digital channels (Mashhadi, Malekmohammadi, & Marvasti, 2019) (Niranjan & M.N.Suma, 2012) (Pranjay Gupta, 2015) (Al Smadi, Al Taweel, & Igried, 2016). In ADM, the amplitude of the analog signal is quantized to a number of levels, and the difference between the original signal and the quantized signal (known as the error signal) is transmitted. One of the main advantages of ADM is its simplicity, making it easy to implement in hardware. However, ADM has a high quantization noise level, which can be reduced by using adaptive techniques. In adaptive delta modulation, the step size used for quantization is dynamically adjusted based on the input signal amplitude. This allows the system to use a smaller step size for low-amplitude signals and a larger step size for high-amplitude signals, resulting in better signal-to-noise ratio (SNR) performance (Sudaradjat, 2019). The adaptation algorithm used in ADM is based on feedback loops, where the error signal is used to adjust the step size. This can be achieved using various techniques, such as a simple proportional-integral (PI) controller or more sophisticated adaptive algorithms based on neural networks or fuzzy logic. Overall, ADM is a useful technique for simple digital communication systems where SNR requirements are not too high. Its simplicity and low computational requirements make it a popular choice for applications such as voice transmission over low-bandwidth channels or in wireless sensor networks (Peric, Denic, & Despotovic, 2021). The working principle of the delta modulation system is to compare the input signal  $x(t)$  with the delta modulation signal  $x_q(n)$ , as seen in Figure 1. If  $x(t) > x_q(n)$ , a bit 1 will be generated and  $x_q(n)$  will increase one step to follow the change in  $x(t)$ . Conversely, if  $x(t) < x_q(n)$ , a bit 0 will be generated and  $x_q(n)$  will decrease one step to follow the change in  $x(t)$ , and so on. However, when the slope of the  $x(t)$  signal is steep, the  $x_q(n)$  signal is difficult to match the slope of  $x(t)$ , a condition known as slope overload. Slope overload can occur mainly if  $x(t)$  changes too quickly and  $x_q(n)$  is unable to follow  $x(t)$ . The working requirements of the Linear Delta Modulation system to avoid slope overload can be derived as follows. Assume that the sampling period  $T_s$  is much smaller than the highest sinusoidal component of the signal's period. In Figure 2, during one  $T_s$  interval,  $x_q(n)$  changes by  $D$ , which is equal to one sample step (delta step). Then the slope of  $x_q(n)$  is  $D/T_s$  or  $D \cdot f_s$ , where  $f_s$  is the sampling frequency. Assume that  $x(t)$  is a sinusoid with the equation:

$$x(t) = A \sin \omega t \quad \dots\dots\dots(1)$$

$$\text{therefore } \frac{d x(t)}{dt} = \omega A \cos \omega t \quad \dots\dots\dots(2)$$

Assuming that, in order to prevent overload, the slope of  $x(t)$  must always be smaller than the slope of  $x_q(n)$ , and at the limit, it should be equal to the slope of  $x_q(n)$ .

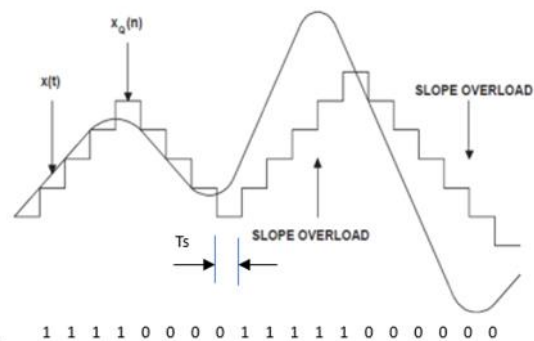
Therefore, to avoid overload, the following condition must be met:

$$\left| \frac{dx(t)}{dt} \right|_{max} = \omega A \leq D \cdot f_s \quad \dots\dots\dots(3)$$

Therefore, to prevent overload, we should:

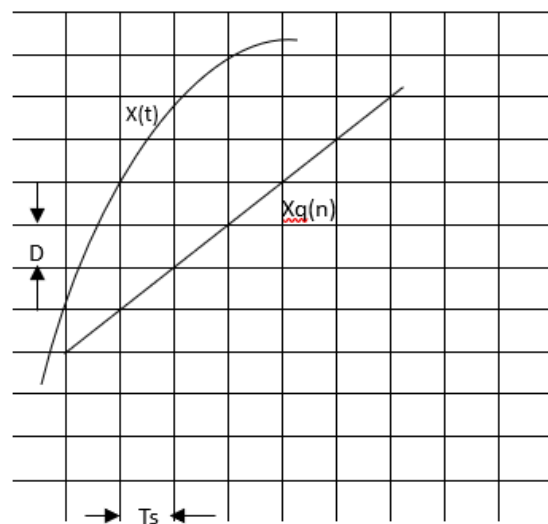
$$A_{max} = \frac{D \cdot f_s}{\omega} \quad \dots\dots\dots(4)$$

From equation (3), it can be concluded that in order to avoid overload, the condition must always be met: the amplitude of the signal component must decrease as the frequency increases.



Source: (Niranjan & M.N.Suma, 2012)

Figure 1. Slope Overload in Delta Modulation



Source : (Sudaradjat, 2019)

Figure 2. Slope Overload

**RESEARCH METHOD**

The steps of the research method conducted involve simulating the DM and ADM systems using Matlab. The performance of the DM and ADM systems

is then compared based on the simulation results to observe the differences. Simulation facilitates researchers in searching for the optimal adaptive value of the ADM system. In the Adaptive Delta Modulation (ADM) system, a variable step size, denoted as  $D$ , is created to follow the changes in the input signal  $x(t)$ . This ensures that the output signal  $x_q(n)$  can continuously track the variations in the input signal  $x(t)$ . In order to analyze the ADM system in this study, simulations were conducted using the MATLAB application. Adaptive Delta Modulation (ADM) is a type of delta modulation that utilizes a variable step size to enhance the performance of regular delta modulation (DM). ADM attempts to track the input signal by adjusting the step size based on the difference between the input signal and the modulator output.

The provided function, `adeltamod`, implements ADM and takes four arguments: `sig_in`: the input signal to be modulated, which should be a vector. `Delta`: the minimum step size, which will be multiplied by  $2^n$  if necessary. This value determines the granularity of the output signal and affects the signal-to-noise ratio (SNR) of the modulated signal. `ts`: the desired sampling period for the ADM output. This value should be an integer multiple of the input signal's period. If not, the function will round it to the nearest integer. `td`: the original sampling period of the input signal `sig_in`.

The output of the function is the ADM output signal, `ADMout`.

Here is an example usage of the `adeltamod` function:

```
function [ADMout] = adeltamod(sig_in, Delta, ts, td)
```

```
    if(round(ts/td) >= 2)
        Nfac = round(ts/td); %Nearest integer
        xsig = downsample(sig_in,Nfac);
        Lxsig = length(xsig);
        Lsig_in = length(sig_in);

        ADMout = zeros(Lsig_in); %Initialising output

        cnt1 = 0; %Counters for no. of previous
        consecutively increasing
        cnt2 = 0; %steps
        sum = 0;

        for i=1:Lxsig

            if (xsig(i) == sum)

                elseif (xsig(i) > sum)

                    if (cnt1 < 2)
                        sum = sum + Delta; %Step up by Delta,
                        same as in DM
                    elseif (cnt1 == 2)
```

```
                        sum = sum + 2*Delta; %Double the step
                        size after
                                %first two increase
                    elseif (cnt1 == 3)
                        sum = sum + 4*Delta; %Double step size
                    else
                        sum = sum + 8*Delta; %Still double and
                        then stop
                                %doubling thereon
                    end

                    if (sum < xsig(i))
                        cnt1 = cnt1 + 1;
                    else
                        cnt1 = 0;
                    end

                    else

                        if (cnt2 < 2)
                            sum = sum - Delta;
                        elseif (cnt2 == 2)
                            sum = sum - 2*Delta;
                        elseif (cnt2 == 3)
                            sum = sum - 4*Delta;
                        else
                            sum = sum - 8*Delta;
                        end

                        if (sum > xsig(i))
                            cnt2 = cnt2 + 1;
                        else
                            cnt2 = 0;
                        end

                    end

                end

            end

        end

    end
```

In this example, the input signal is a sinusoidal wave with a frequency of 10 Hz, sampled at a frequency of 1000 Hz. ADM is implemented with a minimum step size of 0.1 and a desired sampling period of  $2 \cdot td$ . The output of the ADM is plotted along with the original input signal.

## RESULTS AND DISCUSSION

In this simulation, MATLAB was used to model and simulate the ADM and DM systems for encoding a sinusoidal signal, and then compare the two systems. As seen in figure 3, it can be observed that in the DM system, steep changes in the input signal cannot be followed, resulting in slope overload. On the other hand, the ADM system can track sharp changes in the input signal, resulting in a smaller slope overload compared to the DM system.

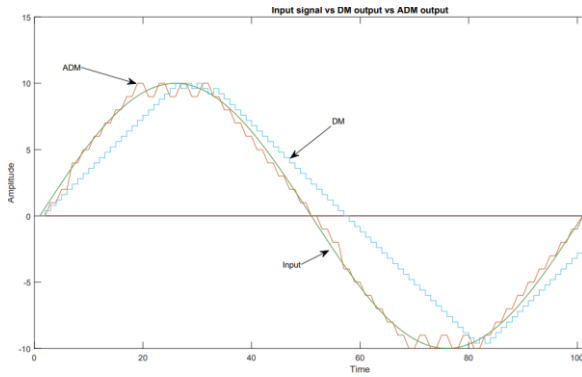


Figure 3. Input, DM, and ADM

The comparison of the coding results between ADM and DM can be seen in the figure 4. It is apparent that the quantization error in the DM system is larger compared to the one generated by the ADM system. Similarly, from the calculation results, DM  $\text{sum\_of\_squared\_error} = 0.7690$  and ADM  $\text{sum\_of\_squared\_error} = 0.3470$ , indicating that the quantization error produced by the DM system is larger compared to the one generated by the ADM system. Therefore, the ADM system exhibits better quality in terms of S/Nq compared to the DM system.

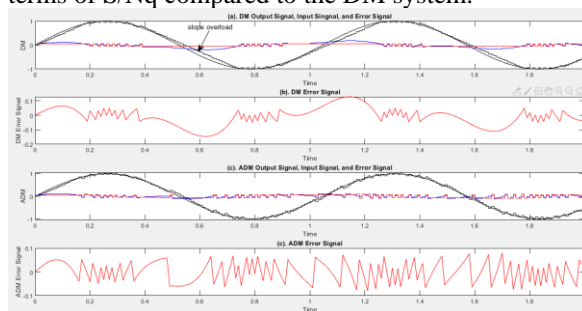


Figure 4. The comparison of the coding results between ADM and DM (The image is obtained from commands in Matlab coding)

The simulation analysis results indicate that the ADM system can encode sinusoidal signals better than the DM system. Furthermore, the adaptive capability of the ADM system has proven to be effective in maintaining good signal quality even in the presence of slope overload, which is a limitation in the DM system that the ADM system can overcome.

However, the simulation also reveals that the ADM system is still vulnerable to distortion in the encoded signals, especially for high-frequency signals. This distortion occurs because the ADM system uses only one delta bit to encode each sample, thus there is an upper limit to the amplitude changes that can be encoded. This can reduce the quality of the received signal.

Additionally, the simulation also demonstrates that the encoding speed of the ADM system is relatively slow compared to other digital modulation methods such as Pulse Code Modulation

(PCM). This is due to the fact that the ADM system needs to wait for the next sample before encoding the previous one. This slow encoding speed may limit the use of ADM in applications that require high speed.

Overall, the simulation analysis results suggest that the ADM system is suitable for use in channels with limited bandwidth and in applications that do not require high speed. However, the drawbacks of the ADM system, namely signal distortion and slow encoding speed, need to be considered when selecting the appropriate digital modulation method for a given application.

The results of this simulation analysis provide an understanding of the performance of the ADM system and insights into optimizing system parameters to achieve better quality of the output signal.

## CONCLUSION

The simulation using MATLAB has confirmed that the use of adaptive ADM can produce digital signals with better quality compared to conventional delta modulation (DM). By adaptively adjusting the delta level, the system can reduce distortion and generated noise. The calculation results show that the  $\text{sum\_of\_squared\_error}$  for DM is 0.7690, while for ADM it is 0.3470. From the observations, it is evident that the ADM system has better quality compared to the DM system.

The simulation also demonstrates that system parameters such as delta level, threshold level, and adaptation speed can be optimized to achieve desired results. Proper parameter settings can help improve the quality of the output signal and reduce distortion.

The use of MATLAB as a simulation tool provides flexibility and ease in implementing the adaptive ADM system. The powerful features of MATLAB allow for accurate modeling and comprehensive performance evaluation.

## REFERENCES

- Al Smadi, T., Al Taweel, F., & Igried, A.-K. (2016). Estimation and Design Techniques for Adaptive Delta Modulation Using Otas. *American Journal of Science, Engineering and Technology*, *http://www.sciencepublishinggroup.com/j/ajs et*, 49-52.
- DENIC, B., PERIC, Z., & DESPOTOVIC, V. (2017). Three-Level Delta Modulation for Laplacian. *Advances in Electrical and Computer Engineering*, *https://www.academia.edu/33890323/Three\_Level\_Delta\_Modulation\_for\_Laplacian\_So urce\_Coding*, 95-102.
- Mansour, C., Achkar, R., & Haidar, G. A. (2012). Simulation of DPCM and ADM Systems. *International Conference on Computer Modeling and Simulation*,

- [https://www.academia.edu/11142775/Simulation\\_of\\_DPCM\\_and\\_ADM\\_Systems](https://www.academia.edu/11142775/Simulation_of_DPCM_and_ADM_Systems), 1-6.
- Mashhadi, M. B., Malekmohammadi, S., & Marvasti, F. (2019). Sparsity Promoting Reconstruction of Delta Modulated Voice Samples by Sequential Adaptive Thresholds. *Arxiv*, <https://arxiv.org/pdf/1902.03425.pdf>, 1-16. doi:<https://doi.org/10.48550/arXiv.1902.03425>
- Niranjan, & M.N.Suma. (2012). Adaptive Delta Modulation Techniques. *Special Issue of International Journal of Computer Applications (0975 – 8887)* <https://research.ijcaonline.org/icedsp/number2/icedsp1016.pdf>, 28-31.
- Peric, Z., Denic, B., & Despotovic, V. (2018). Multilevel Delta Modulation with Switched First-Order Prediction for Wideband Speech Coding; *ELEKTRONIKA IR ELEKTROTEHNIKA*, DOI: <https://doi.org/10.5755/j01.eie.24.1.20156>, 46-51.
- PERIC, Z., DENIC, B., & DESPOTOVIC, V. (2019). Novel Two-Bit Adaptive Delta Modulation Algorithms. *INFORMATICA*, DOI: <http://dx.doi.org/10.15388/Informatica.2019.200>, 117-134.
- Peric, Z., Denic, B., & Despotovic, V. (2021). Algorithm based on 2-bit adaptive delta modulation and fractional linear prediction for Gaussian source coding. *IET Signal Processing*, <https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/sil2.12040>, 410-423.
- Pranjoy Gupta, I. S. (2015). Performance Analysis of different Discrete Time and Discrete Amplitude Representation Techniques used for Encoding the Speech Waveform Digitally. *International Journal of Applied Engineering Research*; [https://www.researchgate.net/publication/281650667\\_Performance\\_Analysis\\_of\\_different\\_Discrete\\_Time\\_and\\_Discrete\\_Amplitude\\_Representation\\_Techniques\\_used\\_for\\_Encoding\\_the\\_Speech\\_Waveform\\_Digitally](https://www.researchgate.net/publication/281650667_Performance_Analysis_of_different_Discrete_Time_and_Discrete_Amplitude_Representation_Techniques_used_for_Encoding_the_Speech_Waveform_Digitally), 34577-34585.
- Sinha, A., Bhowmick, T., & Sinha, S. (2016). Practical Approach of Producing Delta Modulation and . *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, [https://www.academia.edu/27297635/Practical\\_Approach\\_of\\_Producing\\_Delta\\_Modulation\\_and\\_Demodulation](https://www.academia.edu/27297635/Practical_Approach_of_Producing_Delta_Modulation_and_Demodulation), 87-94.
- Sudaradajat, D. (2019). Perbandingan Algoritma Pengkodean Suara Manusia : Adaptive Delta Modulation and Linear Prediction Coding. *Paradigma*, DOI: [10.31294/p.v20i2; https://ejournal.bsi.ac.id/ejournal/index.php/paradigma/article/view/4641/pdf](https://ejournal.bsi.ac.id/ejournal/index.php/paradigma/article/view/4641/pdf), 23-28.
- Sudaradajat, D. (2019). Teknik Adaptive Pada Modulasi Delta. *Paradigma, Volume XXI No. 2 September 2019*, <http://ejournal.bsi.ac.id/ejournal/index.php/paradigma>, DOI: <https://doi.org/10.31294/p.v21i2.5632>, 137-142.
- Ukarsha Patil, R. K. (2018). A SURVEY ON SIGMA DELTA MODULATION BASED ADC DESIGN. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & MANAGEMENT*, <http://www.ijesmjournal.com/issues%20PDF%20file/Archive-2018/Jan-mar-2018/9.pdf>, 1-5.