

Analisis Efek Aliasing Pada Sinyal Audio Dengan Variasi Frekuensi Sampling Pada Lagu Berjudul Terhebat

Hasyiyati Shabrina¹, Inda Ahmayani², Khalifah Audya Eka Putri³,
Endah Setyowati^{4*}

^{1,2,3,4}Universitas Pendidikan Indonesia, Indonesia, Sistem Telekomunikasi
e-mail: ¹hasyyatii27@upi.edu, ²indaahmyn.18@upi.edu,
³khalifahaudya.14@upi.edu, ⁴endahsetyowati@upi.edu

Abstrak

Aliasing merupakan fenomena yang muncul dalam pengolahan sinyal digital ketika frekuensi sampling tidak cukup tinggi untuk merepresentasikan frekuensi tertinggi dari sinyal, sehingga mengakibatkan distorsi. Penelitian ini menganalisis efek aliasing pada sinyal audio dengan melakukan variasi frekuensi sampling menggunakan lagu "Terhebat" sebagai sampel. Frekuensi sampling yang diuji meliputi 11025 Hz, 22050 Hz, 32000 Hz, 44100 Hz, dan 88200 Hz, dan proses analisis dilakukan dengan perangkat lunak MATLAB. Hasilnya menunjukkan bahwa frekuensi sampling rendah, seperti 11025 Hz dan 22050 Hz, menyebabkan distorsi signifikan pada sinyal, sedangkan frekuensi 44100 Hz memberikan kualitas audio terbaik dengan efek aliasing minimal. Sebaliknya, frekuensi sampling yang sangat tinggi, seperti 88200 Hz, tidak memberikan peningkatan kualitas yang berarti. Kesimpulan penelitian ini didasarkan pada metode eksperimental yang dilakukan pada visualisasi sinyal dan evaluasi subjektif, dengan rujukan pada teori pengolahan sinyal, khususnya Teorema Nyquist-Shannon, yang menekankan pentingnya pemilihan frekuensi sampling yang tepat untuk meminimalkan efek aliasing dan menjaga kualitas audio.

Kata kunci — Aliasing, Frekuensi Sampling, Sinyal Audio, MATLAB, Teorema Nyquist-Shannon

Abstract

Aliasing is a phenomenon that appears in digital signal processing when the sampling frequency is not high enough to represent the highest frequency of the signal, resulting in distortion. This research analyzes the aliasing effect on audio signals by varying the sampling frequency using the song "Terhebat" as a sample. The sampling frequencies tested include 11025 Hz, 22050 Hz, 32000 Hz, 44100 Hz, and 88200 Hz, and the analysis process is performed with MATLAB software. The results show that low sampling frequencies, such as 11025 Hz and 22050 Hz, cause significant distortion to the signal, while the frequency of 44100 Hz provides the best audio quality with minimal aliasing effects. In contrast, very high sampling frequencies, such as 88200 Hz, do not provide any significant quality improvement. The conclusions of this study are based on experimental methods conducted on signal visualization and objective evaluation, with reference to signal processing theory, particularly the Nyquist-Shannon Theorem, which emphasizes the importance of selecting the right sampling frequency to minimize aliasing effects and maintain audio quality.

Keywords — Aliasing, Frequency Sampling, Audio Signal, MATLAB, Nyquist-Shannon Theorem

I. PENDAHULUAN

Aliasing merupakan efek di mana sinyal yang dihasilkan dari proses sampling memiliki frekuensi yang berbeda dari sinyal asli [1]. Teorema Nyquist-Shannon menetapkan bahwa frekuensi sampling harus setidaknya dua kali frekuensi maksimum

dari sinyal asli untuk menghindari aliasing. Jika tidak memenuhi syarat ini, frekuensi tinggi dalam sinyal asli akan tercampur dengan frekuensi rendah dalam sinyal yang disampling, menyebabkan distorsi dan penurunan kualitas sinyal [2].

Dalam konteks audio, aliasing dapat mengakibatkan perubahan totalitas dan pengenalan frekuensi yang tidak diinginkan karena merusak integritas musik asli. Misalnya didalam proses digitalisasi lagu, frekuensi sampling yang tidak memadai dapat menghasilkan artefak audio yang mengganggu, membuat yang mendengarkan menjadi tidak nyaman [3].

Pengolahan sinyal suara dilakukan melalui serangkaian proses untuk memperoleh hasil berupa informasi mengenai durasi atau waktu berbicara atau mengeluarkan bunyi dalam bentuk grafik sinyal suara dan spektrumnya. Proses akuisisi sinyal suara ini dimulai dari sinyal suara yang diinput oleh pemrogram melalui mikrofon pada laptop. Sinyal suara tersebut kemudian diproses oleh sistem ADC sehingga sesuai dengan frekuensi sampling, dikuantisasi, dan dikonversi menjadi kode biner. Kode biner ini dimasukkan ke dalam DAC untuk menghasilkan sinyal analog yang sesuai dengan karakteristik sinyal input [4].

Peneliti menggunakan software MATLAB sebagai alat yang kuat untuk analisis dan visualisasi data. *Matrix Laboratory* (MATLAB) pada awalnya bukanlah bahasa pemrograman. MATLAB merupakan kalkulator matriks interaktif sederhana yang dibangun dari pustaka perangkat lunak matriks LINPACK dan EISPACK. Beberapa tipe dan struktur data baru, termasuk bilangan bulat, tipe logis, struktur, serta objek, diperkenalkan pada akhir 1990-an. Peningkatan lingkungan komputasi MATLAB telah mendominasi perkembangan dalam beberapa tahun terakhir, termasuk ekstensi ke desktop, perbaikan besar pada sistem objek dan grafis, dukungan komputasi paralel, serta "Live Editor" yang menggabungkan program, teks, output, dan grafik ke dalam satu dokumen interaktif. [5], [6]. MATLAB menyediakan berbagai fungsi untuk mengilustrasikan dan menganalisis efek aliasing. Dengan menggunakan MATLAB, praktikan dapat mensimulasikan proses sampling, memvisualisasikan sinyal asli dan sinyal yang disampling, serta mendemonstrasikan bagaimana aliasing mempengaruhi kualitas sinyal audio [7].

Tujuan dari penelitian ini adalah untuk menganalisis efek aliasing pada sebuah lagu menggunakan software MATLAB. Penelitian dimulai dengan menggunakan sinyal audio asli, kemudian melakukan sampling pada frekuensi yang berbeda-beda, menganalisis dan membandingkan hasilnya untuk menunjukkan bagaimana aliasing terjadi, serta bagaimana hal itu mempengaruhi kualitas audio.

II. METODE PENELITIAN

Penelitian ini menggunakan metode eksperimen untuk menganalisis efek aliasing pada sinyal audio dengan variasi frekuensi sampling menggunakan lagu 'Terhebat'. Pendekatan eksperimen dipilih karena memungkinkan peneliti untuk memanipulasi variabel independen. Dalam hal ini frekuensi sampling digunakan untuk

mengamati pengaruh terhadap variabel dependen, yaitu efek aliasing pada sinyal audio. Prosedur penelitian eksperimen ini terdiri dari beberapa tahap, dimulai dengan mengidentifikasi variabel penelitian. Variabel independen adalah frekuensi sampling yang akan dimanipulasi dengan menggunakan nilai-nilai tertentu, seperti 11025 Hz, 22050 Hz, 44100 Hz, 32000 Hz, dan 88200 Hz. Variabel dependen adalah efek aliasing pada sinyal audio yang akan diamati sebagai respon terhadap manipulasi variabel independen.

Penelitian ini menggunakan aplikasi MATLAB dan fitur-fitur di dalamnya, seperti filter digital sebagai instrumen utama untuk menghasilkan data yang dibutuhkan. Filter digital adalah algoritma atau prosedur matematika yang mengolah sinyal masukan digital dan menghasilkan isyarat digital yang memiliki sifat tertentu sesuai dengan tujuan filter [8]. Secara teknis, proses pengumpulan data dimulai dengan mempersiapkan sampel audio dalam format WAV. File WAV adalah format standar yang banyak digunakan dalam berbagai jenis *game* dan multimedia. Perkembangan teknologi dengan berbagai media dan kemudahan akses membuatnya lebih mudah untuk bertukar informasi di berbagai tempat [9]. Setelah mengganti format audio yang akan digunakan menjadi format WAV, selanjutnya yaitu membuat skrip MATLAB untuk memproses audio dengan berbagai variasi frekuensi sampling yang telah ditentukan. Data yang dihasilkan dari MATLAB berupa plot spektrum frekuensi dan perubahan karakteristik audio lainnya dicatat dan dianalisis. Analisis data dilakukan dengan membandingkan efek aliasing pada setiap frekuensi sampling, serta mengamati pola yang muncul akibat manipulasi variabel independen [10].

Hasil eksperimen dan analisis data kemudian digunakan untuk menarik kesimpulan tentang pengaruh variasi frekuensi sampling terhadap efek aliasing pada sinyal audio. Kesimpulan ini dapat memberikan informasi berharga tentang bagaimana memilih frekuensi sampling yang tepat untuk mengurangi efek aliasing dan meningkatkan kualitas audio.

III. HASIL DAN PEMBAHASAN

Dalam Penelitian ini, peneliti melakukan pengamatan terhadap efek aliasing pada sinyal audio lagu “Terhebat” dengan menggunakan variasi frekuensi sampling yang berbeda. Proses dimulai dengan mengonversi format file audio dari MP3 menjadi WAV agar dapat diproses menggunakan MATLAB. Selanjutnya, peneliti membuat skrip dalam MATLAB untuk membaca *file* audio, mengatur frekuensi sampling, dan memvisualisasikan sinyal audio dalam domain waktu.

```
1 % Membaca file audio.wav
2 [audio, fs] = audioread('terhebat.wav');
3 fs = 32000;
4 sound(audio,fs);
5
6 % Menghitung panjang sinyal audio
7 nSamples = size(audio, 1);
8
9 % Menghitung durasi sinyal audio dalam detik
10 duration = nSamples / fs;
11
12 % Membuat sumbu waktu
13 time = linspace(0, duration, nSamples);
14
15 % Menampilkan sinyal audio menggunakan plot
16 plot(time, audio);
17 xlabel('Waktu(detik)');
18 ylabel('Amplitudo');
19 title('Sinyal Audio');
20
21 % Menyesuaikan skala sumbu x agar sesuai dengan durasi sinyal audio
22 xlim([0 duration]);
23
```

Gambar 1. Syntax untuk Menampilkan Garfik Sinyal

Keterangan Gambar 1:

[audio, fs] = audioread('terhebat.wav'); membaca data dari file bernama 'terhebat.wav' dan mengembalikan data sampel audio (audio) dan frekuensi sampel (fs) fs = 32000;

Mengatur frekuensi sampling menjadi 32000 Hz sound (audio, fs); memainkan sinyal audio (audio) dengan frekuensi sampling (fs) nSamples = size(audio, 1);

Menghitung Panjang sinyal audio dengan fungsi size duration = nSamples / fs;

Durasi dihitung dengan membagi jumlah sampel dengan frekuensi sampel time = linspace(0, duration, nSamples); linspace(0, duration, nSamples) membuat vector dengan nSamples elemen yang merata terdistribusi antara 0 dan duration (durasi sinyal audio) plot(time, audio);

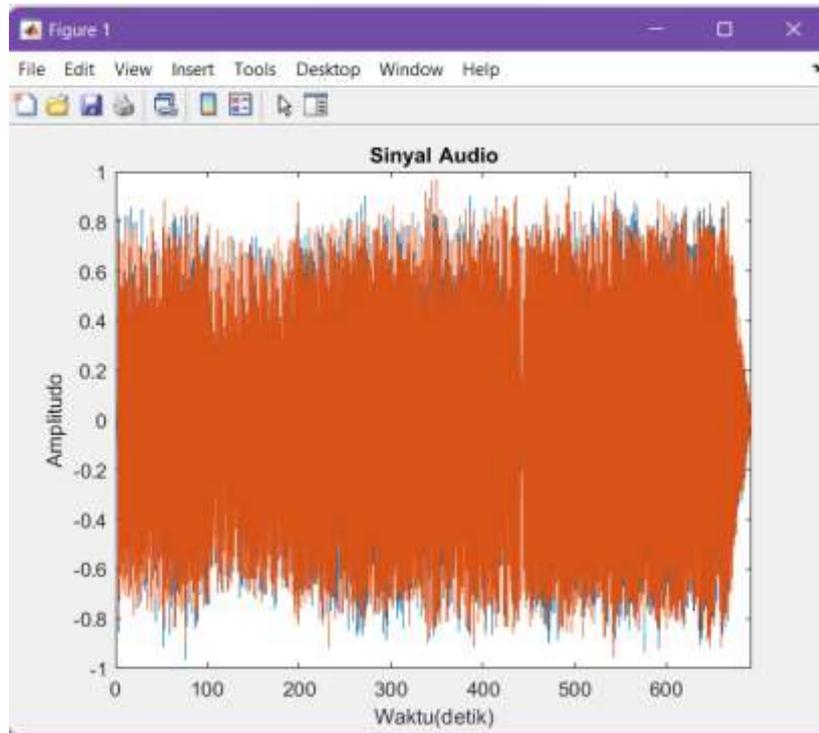
Membuat plot dua dimensi yang menampilkan data audio sebagai fungsi dari waktu xlabel('Waktu (detik)');

Memberi label pada sumbu x dengan teks 'Waktu(detik)' ylabel('Amplitudo');

Memberi label pada sumbu y dengan teks 'Amplitudo' title('Sinyal Audio');

Memberi judul pada plot dengan teks 'Sinyal Audio' xlim([0 duration]); xlim([0 duration]) mengatur batas sumbu x dari 0 hingga duration

- a. Pengamatan Efek Aliasing pada Audio dengan Frekuensi Sampling sebesar 11025 Hz.



Gambar 2. Hasil Simulasi pada Efek Aliasing Audio Sinyal Musik dengan F_s 11025 Hz

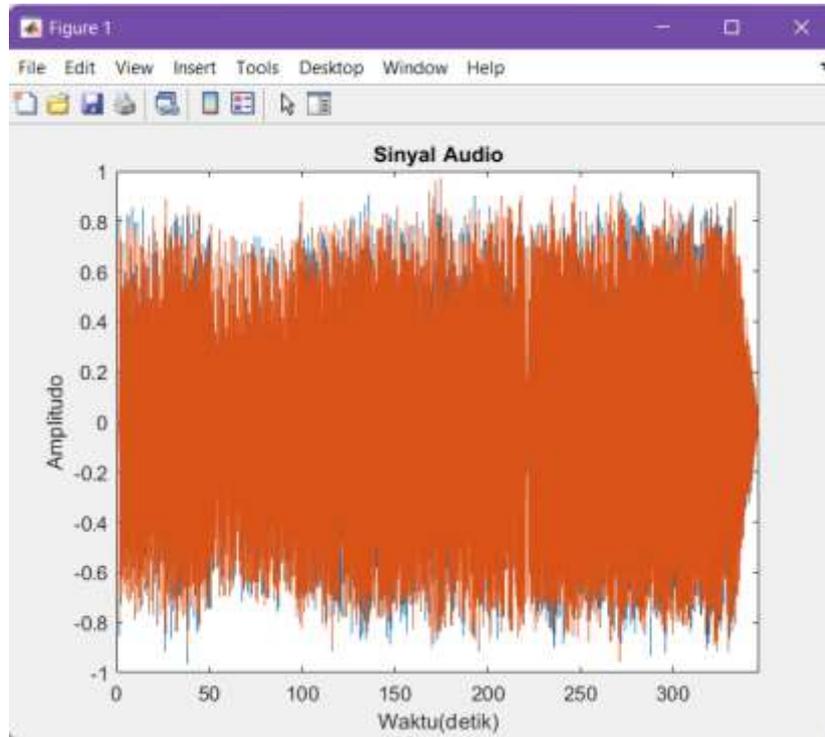
Jumlah sampel per detik yang diambil dari sinyal kontinu untuk membuat sinyal diskrit disebut frekuensi sampling, yang berarti 11025 sampel per detik diambil dari sinyal asli. Efek penggunaan frekuensi sampling tergantung pada aplikasinya dan karakteristik sinyal yang disampling. Frekuensi sampling harus setidaknya dua kali lipat dari frekuensi tertinggi sinyal asli. Misalnya, dengan frekuensi sampling 11025 Hz, frekuensi tertinggi sinyal asli yang dapat direkonstruksi dengan baik adalah 5522.5 Hz, jadi akan terjadi aliasing jika sebagian frekuensi sinyal asli lebih tinggi dari ini.

Frekuensi sampling 11025 Hz dianggap cukup rendah dalam konteks audio. Sebagai contoh, CD audio standar menggunakan frekuensi sampling 44100 Hz, yang memungkinkan pengolahan frekuensi audio hingga 22050 Hz sampai sekitar 20 kHz dari seluruh spektrum pendengaran manusia, tetapi dengan 11025 Hz kualitas audio akan lebih rendah karena frekuensi tertinggi yang dapat diwakili hanya 5522.5 Hz, yang berarti banyak detail suara hilang, terutama pada frekuensi yang lebih tinggi.

Pada penelitian ini, peneliti hanya merubah frekuensi sampling sehingga jika hanya frekuensi sampling yang diubah tanpa mengubah jumlah sampel, durasi musik

akan berubah. Durasi musik akan bertambah jika frekuensi sampling diturunkan, dan akan berkurang jika frekuensi sampling dinaikkan. Pada percobaan pertama ini peneliti menggunakan frekuensi sampling sebesar 11025 Hz, sehingga durasi musik yang dihasilkan bertambah lebih panjang.

- b. Pengamatan Efek Aliasing pada Audio dengan Frekuensi Sampling sebesar 22050 Hz.



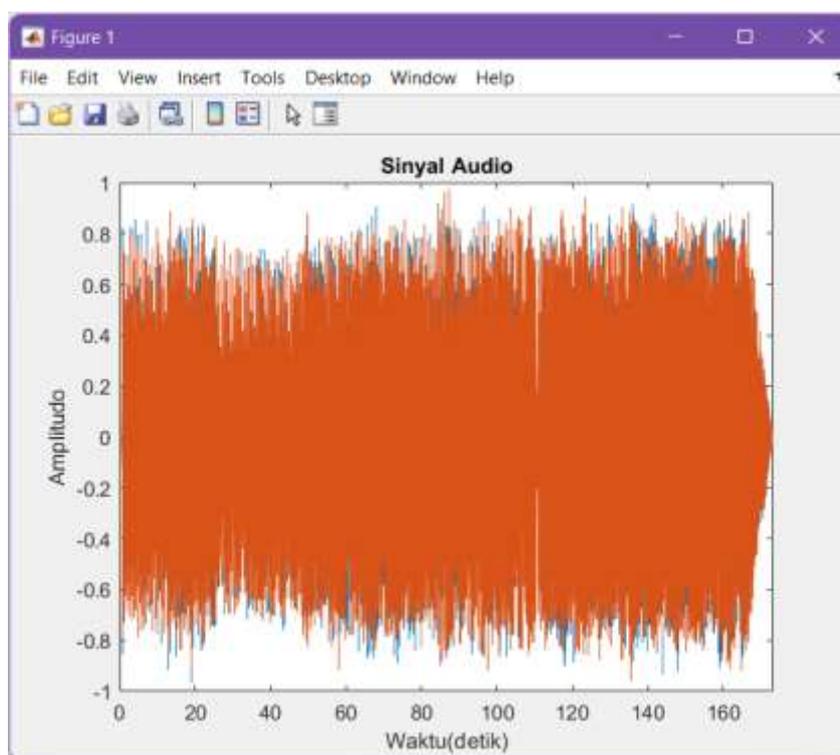
Gambar 3. Hasil Simulasi pada Efek Aliasing Audio Sinyal Musik dengan F_s 22050 Hz

Gambar sinyal audio di atas menunjukkan distorsi dan aliasing yang cukup besar. Sepertinya bentuk gelombangnya tidak sempurna, dan ada bagian frekuensi tambahan yang tidak seharusnya ada. Ini adalah hasil dari fakta bahwa frekuensi sampling 22050 Hz tidak memenuhi kriteria Nyquist untuk merekam sinyal audio dengan kualitas yang baik. Berdasarkan teorema Nyquist, frekuensi sampling minimal yang diperlukan untuk merekam sinyal audio dengan akurat adalah dua kali lipat frekuensi maksimum sinyal tersebut. Untuk aplikasi sinyal audio seperti musik atau suara manusia, frekuensi maksimum yang dapat didengar oleh telinga manusia adalah sekitar 20 kHz. Oleh karena itu, frekuensi sampling 22050 Hz tidak cukup untuk merekam sinyal suara dengan benar, karena frekuensi ini hanya sedikit lebih tinggi dari frekuensi tertinggi yang dapat didengar oleh manusia.

Akibatnya, sinyal audio dengan frekuensi sampling 22050 Hz mengalami efek aliasing. Aliasing adalah pencampuran antara frekuensi tinggi dan rendah, yang menyebabkan komponen frekuensi tinggi yang tidak seharusnya ada dalam sinyal audio asli. Akibatnya, sinyal audio menjadi distorsi, yang mengakibatkan kualitas sinyal audio

yang lebih rendah secara keseluruhan. Untuk output suara, efek aliasing yang signifikan pada frekuensi sampling 22050 Hz dapat menyebabkan suara yang terdengar tidak alami dan cacat. Selain itu, detail frekuensi tinggi mungkin tidak terekam dengan baik, sehingga output suara mungkin terdengar kurang jernih dan kurang detail.

- c. Pengamatan Efek Aliasing pada Audio dengan Frekuensi Sampling sebesar 44100 Hz.



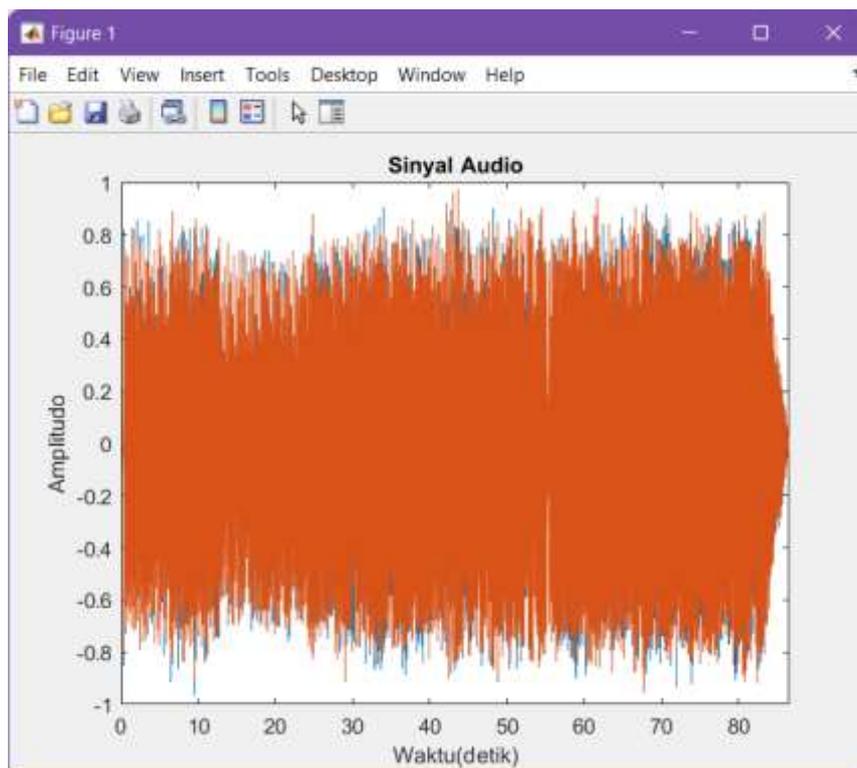
Gambar 4. Hasil Simulasi pada Efek Aliasing Audio Sinyal Musik dengan F_s 44100 Hz

Efek aliasing pada audio dengan frekuensi sampling sebesar 44100 Hz dapat diamati ketika sinyal audio memiliki komponen frekuensi yang melebihi setengah dari frekuensi sampling tersebut, yaitu 22050 Hz. Dalam kasus ini, jika sinyal audio memiliki komponen frekuensi yang melebihi 22050 Hz, efek aliasing dapat terjadi. Misalnya, pertimbangkan sebuah sinyal audio yang memiliki komponen frekuensi pada 25000 Hz. Ketika sinyal ini direkam dengan frekuensi sampling 44100 Hz, frekuensi 25000 Hz akan melebihi frekuensi Nyquist (setengah dari frekuensi sampling), yang berarti efek aliasing dapat terjadi.

Efek aliasing akan terjadi karena sinyal frekuensi yang lebih tinggi dari 22050 Hz akan "dilipat" kembali ke dalam rentang frekuensi yang dapat direpresentasikan oleh sistem, menciptakan distorsi atau artefak yang tidak diinginkan. Dalam kasus ini, komponen frekuensi 25000 Hz akan terlipat kembali ke dalam rentang frekuensi yang dapat direpresentasikan (0-22050 Hz), yang dapat menghasilkan distorsi yang tidak diinginkan dalam audio yang direkam atau direproduksi. Untuk mencegah efek aliasing,

penting untuk menggunakan filter anti-aliasing sebelum merekam atau merepresentasikan sinyal audio digital. Filter ini akan memastikan bahwa komponen frekuensi yang melebihi frekuensi Nyquist disaring sebelum sinyal direkonstruksi, sehingga menghasilkan reproduksi audio yang lebih akurat dan bebas dari distorsi yang disebabkan oleh efek aliasing, dengan audio yang dihasilkan setia terhadap audio asli.

- d. Pengamatan Efek Aliasing pada Audio dengan Frekuensi Sampling sebesar 88200 Hz.



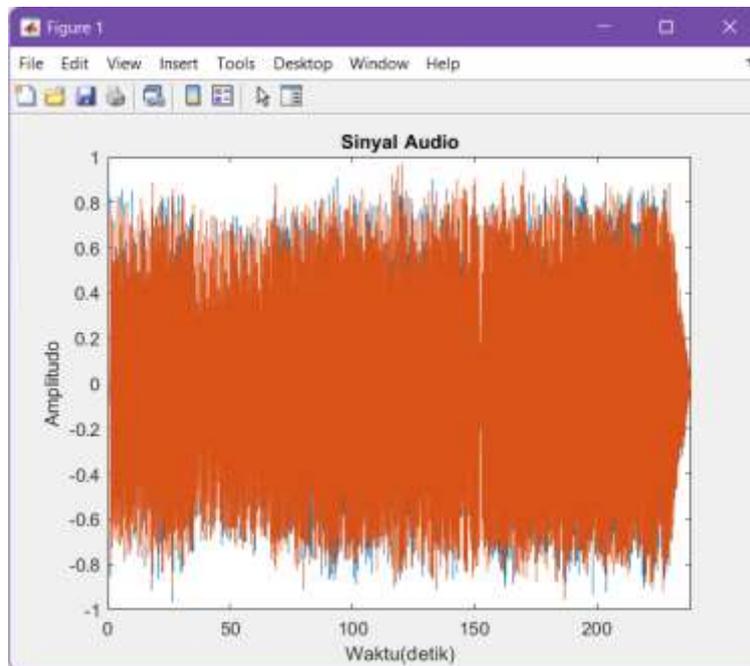
Gambar 5. Hasil Simulasi pada Efek Aliasing Audio Sinyal Musik dengan F_s 88200 Hz

Faktor penting yang harus diperhatikan selama proses konversi sinyal analog ke digital pada audio adalah frekuensi sampling, yang menentukan seberapa sering sinyal analog diambil sampel dalam satu detik. Jika frekuensi sampling terlalu rendah, dapat menyebabkan efek aliasing, di mana frekuensi tinggi yang tidak dapat direkonstruksi dengan benar akan tampak seperti frekuensi yang lebih rendah. Sebaliknya, jika frekuensi sampling terlalu tinggi, seperti 88200 Hz, dapat menyebabkan efek yang tidak diinginkan pada kualitas audio. Ketika frekuensi sampling kurang dari 20 Hz, efek aliasing biasanya menjadi masalah. Ini karena frekuensi sampling yang sangat tinggi, seperti 88200 Hz, jauh melebihi frekuensi audio yang dapat didengar oleh telinga manusia.

Umumnya, frekuensi sampling antara 44100 Hz dan 48000 Hz dianggap cukup untuk merekam dan mereproduksi sinyal audio berkualitas tinggi tanpa efek aliasing

yang terdengar. Frekuensi sampling yang lebih tinggi, seperti 88200 Hz, dapat digunakan dalam situasi tertentu di mana dibutuhkan presisi yang sangat tinggi atau dalam aplikasi profesional.

- e. Pengamatan Efek Aliasing pada Audio dengan Frekuensi Sampling sebesar 32000 Hz.



Gambar 6. Hasil Simulasi pada Efek Aliasing Audio Sinyal Musik dengan F_s 32000 Hz

Gambar di atas menunjukkan plot sinyal audio dalam domain waktu dengan frekuensi sampling (f_s) 32000 Hz. Sinyal tersebut terlihat kompleks dan tidak periodik, yang menunjukkan bahwa itu adalah sinyal audio sebenarnya, seperti suara atau musik. Amplitudo sinyal berkisar antara -1 dan 1, yang merupakan rentang amplitudo normal sinyal audio yang telah dinormalisasi. Dalam sinyal audio, terlihat lonjakan amplitudo yang cukup tinggi di beberapa bagian. Ini menunjukkan bahwa sinyal tersebut mengalami perubahan amplitudo yang cepat.

Gambar tersebut menunjukkan bahwa sinyal audio dengan frekuensi sampling 32000 Hz tidak mengalami distorsi atau efek aliasing yang signifikan. Ini menunjukkan bahwa frekuensi sampling ini cukup untuk merekam sinyal audio dengan kualitas yang baik tanpa aliasing yang terlihat. Berdasarkan teorema Nyquist, frekuensi sampling minimal yang diperlukan adalah dua kali lipat dari frekuensi maksimum sinyal audio. Untuk aplikasi audio seperti musik atau suara manusia, frekuensi maksimum yang dapat didengar oleh telinga manusia sekitar 20 kHz, sehingga frekuensi sampling 32000 Hz sudah cukup.

Namun demikian, komponen frekuensi di atas 16 kHz mungkin masih mengalami sedikit aliasing pada sinyal dengan frekuensi sampling 32000 Hz. Efek aliasing ini mungkin tidak terlalu signifikan dan tidak terdengar terlalu buruk oleh

telinga manusia, jadi output suara yang dihasilkan dari sinyal audio dengan frekuensi sampling 32000 Hz mungkin terdengar cukup jernih dan tidak terdistorsi secara signifikan. Namun, ada kemungkinan bahwa frekuensi tinggi di atas 16 kHz mungkin tidak terekam secara menyeluruh, sehingga output suara mungkin kehilangan beberapa detail pada frekuensi tinggi tersebut.

IV. KESIMPULAN

Dari hasil percobaan melakukan pengamatan efek aliasing pada sebuah lagu “Terhebat” yang telah didownload dan dianalisis menggunakan *software* MATLAB. Peneliti mengetahui bahwa untuk menghindari masalah aliasing dapat menggunakan filter anti-aliasing. Karena filter ini berfungsi untuk menghapus atau mengurangi komponen frekuensi yang melebihi setengah dari frekuensi sampling sebelum dilakukannya proses sampling. Dengan cara ini, frekuensi sinyal asli yang berada di atas frekuensi sampling dapat dibatasi dan dihindari dalam proses mendapatkan sinyal waktu diskrit. Teknik ini memberikan pengurangan efek aliasing dengan cara penghilangan komponen sinyal yang besar, frekuensinya lebih tinggi dari yang diterima oleh alat perekam. Lalu, nilai frekuensi sampling sangat berpengaruh pada output audio karena menentukan seberapa baik sinyal audio dapat direpresentasikan dalam bentuk digital.

Dalam teorinya, frekuensi sampling yang lebih tinggi dapat membuat audio yang dikeluarkan lebih mendetail, jelas, hilangnya noise dan meningkatkan audio dari suara aslinya. Kemudian, pada *software* MATLAB, peneliti mengubah nilai frekuensi samplingnya menjadi “11025, 22050, 44100, 88200, 32000”. Kami mengetahui bahwa semakin kecil nilai frekuensi samplingnya, maka tempo audio yang dihasilkan akan semakin lambat karena sinyal audio terbatas mengakibatkan kurangnya representasi terhadap perubahan sinyal audio. Sebaliknya, jika nilai frekuensi sampling yang tinggi, maka informasi akan lebih banyak dipertahankan, memungkinkan representasi yang lebih baik dari perubahan kecepatan sinyal audionya sehingga menghasilkan tempo lagu yang sesuai dengan nada aslinya.

Namun, jika nilai frekuensi sampling (fs) terlalu tinggi menyebabkan terjadinya aliasing yang mengakibatkan tempo lagu yang dihasilkan semakin cepat hingga suara lagu tidak terdengar jelas. Karena kemampuan sistem untuk mereproduksi informasi sinyal terbatas, maka pemilihan nilai frekuensi sampling yang tepat penting untuk mempertahankan kualitas audio sesuai dengan suara aslinya. Pada pengamatan yang telah dilakukan, rentang nilai frekuensi sampling 11025-22050 menghasilkan suara yang temponya sangat lambat seperti di *slowmotion*. Di rentang 41000 – 48000 menghasilkan suara yang sesuai dengan nada lagu aslinya meskipun masih terdapat sedikit noise. Lalu, pada frekuensi 82000, suara yang dihasilkan memiliki tempo nada yang sangat cepat hingga tidak terdengar jelas suara aslinya.

DAFTAR PUSTAKA

- [1] E. Zamiri, A. Sanchez, M. S. Martínez-García, and A. de Castro, “Analysis of the aliasing effect caused in hardware-in-the-loop when reading PWM inputs of power converters,” *International Journal of Electrical Power & Energy Systems*, vol. 136, p. 107678, Mar. 2022, doi: 10.1016/j.ijepes.2021.107678.

- [2] Á. Navarro-Rodríguez, P. García, and J. M. Guerrero, "Introduction to digital sampled systems and nyquist sampling theorem," in *Encyclopedia of Electrical and Electronic Power Engineering*, J. García, Ed., Oxford: Elsevier, 2023, pp. 652–656. doi: 10.1016/B978-0-12-821204-2.00127-6.
- [3] C. R. Putri et al., "Program Studi S1 Sistem Telekomunikasi Universitas Pendidikan Indonesia Kampus Daerah Purwakarta 2024".
- [4] I. Febriana, "Simulasi Akuisisi Sinyal Suara Dengan Menggunakan MATLAB".
- [5] C. Moler and J. Little, "A history of MATLAB," *Proc. ACM Program. Lang.*, vol. 4, no. HOPL, pp. 1–67, Jun. 2020, doi: 10.1145/3386331.
- [6] "Introduction to MATLAB | SpringerLink." Accessed: Sep. 25, 2024. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-10-8321-1_1
- [7] H. Nabila, "Pengaruh Frekuensi Sampling Terhadap Waktu dan Akurasi Penentuan Lokasi Menggunakan Metode TDOA Multilaterasi," 2021.
- [8] G. Ariyanto, N. Nurgiyatna, and E. Sudarmilah, "Implementasi Filter Digital Finite Impulse Response Metode Penjendelaan Blackman pada DSP TMS320C6711," *Emitor: Jurnal Teknik Elektro*, vol. 2, no. 2, Art. no. 2, Apr. 2018, doi: 10.23917/emitor.v2i2.5994.
- [9] H. Santoso and M. F. M. Fakhriza, "PERANCANGAN APLIKASI KEAMANAN FILE AUDIO FORMAT WAV (WAVEFORM) MENGGUNAKAN ALGORITMA RSA," *Algoritma: Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 1, Art. no. 1, Apr. 2018, doi: 10.30829/algoritma.v2i1.1615.
- [10] T. B. Santoso, H. Mahmudah, and N. A. Siswandari, *PRAKTIKUM SINYAL DAN SISTEM*. 2010.