

Implementação e análise de técnica para estimação de SNR baseado em Deep Learning

João Henrique Silva Delfino, Roberto Michio Marques Kagami, Juliano Silveira Ferreira e Luciano Leonel Mendes

Resumo—As pesquisas em torno de sistemas de radiocomunicação, como as futuras redes 6G, buscam a implementação de enlaces que assegurem a qualidade de serviço mesmo em condições desfavoráveis. Uma métrica fundamental para análise das condições do canal de comunicação é a relação sinal-ruído ou SNR (Signal-to-Noise Ratio). O presente trabalho apresenta uma abordagem de implementação de um algoritmo inovador de estimação desta métrica, baseado em inteligência artificial, a qual proporciona precisão superior à técnica que utiliza NMSE (Normalized Mean Squared Error).

Palavras-Chave—Deep Learning, NMSE, Relação sinal-ruído, 6G.

Abstract—The researches about radio communication systems, as future 6G networks, look for the implementation of communication links that ensure the quality of service even under adverse conditions. A fundamental metric to analyze communication channel condition is the Signal-to-Noise Ratio. This work presents an implementation approach of an innovative algorithm of estimation for this metric based on artificial intelligence, which provides superior precision to the technique that uses Normalized Mean Squared Error.

Keywords—Deep Learning, NMSE, Signal-to-Noise Ratio, 6G.

I. INTRODUÇÃO

A concepção da Sexta Geração de Rede Móvel Celular (6G) [1], deve ser orientada para o atendimento de requisitos demandados pelo aperfeiçoamento e avanço das aplicações existentes, por aplicações emergentes e também novos cenários de uso. A evolução da rede é viabilizada em virtude da adoção de tecnologias e soluções inovadoras, disruptivas e efetivas. Dentre tecnologias elencadas como habilitadoras para compôr a futura rede destaca-se a Inteligência Artificial (IA), que é prevista para ser adotada em diferentes camadas da rede.

A rede 6G deve ser flexível e adaptativa para prover suporte adequado às diferentes aplicações previstas em sua concepção. O desenvolvimento de novas tecnologias, as quais possibilitam uma melhor definição do MCS (Modulation Coding Scheme) de operação do enlace de comunicação do sistema, por exemplo, é fundamental para assegurar um QoS (Quality of Service)

João Henrique Silva Delfino, CRR, Instituto Nacional de Telecomunicações, Santa Rita do Sapucaí-MG, e-mail: joao.delfino@inatel.br; Roberto Michio Marques Kagami, Instituto Nacional de Telecomunicações, Netanya, Israel, e-mail: roberto.michio@mtel.inatel.br; Juliano Silveira Ferreira, CRR, Instituto Nacional de Telecomunicações, Santa Rita do Sapucaí-MG, e-mail: silveira@inatel.br; Luciano Leonel Mendes, CRR, Instituto Nacional de Telecomunicações, Santa Rita do Sapucaí-MG, e-mail: lucianol@inatel.br. Este trabalho foi parcialmente financiado por FAPEMIG, CNPq-Brasil, FAPESP através do projeto SAMURAI (20/05127-2), RNP/MCTI por meio do projeto Brasil 6G (01245.010604/2020-14) e BRITISH GOVERNMENT DIGITAL ACCESS PROGRAMME (DAP) e NIC.BR através do projeto TVWS NIC.BR.

adequado à tais aplicações. Um dos parâmetros essenciais para avaliar e definir as condições do canal e, por conseguinte, definir o modo de operação do sistema, é a relação sinal ruído ou SNR (Signal-to-Noise Ratio).

Diversas técnicas de estimação de SNR tem sido utilizadas de maneira efetiva, como é o caso da NMSE (Normalized Mean Square Error) [2]. Pode-se destacar que o uso de técnicas de IA para refinar esta estimativa tem se tornado viável, principalmente em contextos que possibilitam a geração de conjuntos de dados que podem ser utilizados no treinamento de modelos de ML (Machine Learning). Tais técnicas de IA têm sido cada vez mais empregadas na estimação e predição [3] precisa de parâmetros que envolvem sistemas de comunicação complexos como as redes 5G e 6G.

O presente trabalho apresenta a implementação e validação de uma técnica de estimação de SNR que é mais precisa e confiável que a estimativa dada pelo NMSE, empregando como ferramenta auxiliar uma rede neural. O algoritmo denominado de LDNN-SNRE (Low-Complexity Deep Neural Network for Signal-to-Interference-Plus-Noise Ratio Estimation) foi proposto e validado em [4]. A principal contribuição deste artigo, portanto, é apresentar detalhes de uma abordagem de implementação do referido algoritmo, bem como a avaliação de desempenho de estimação, considerando sua integração em um sistema de radiocomunicação baseado no transceptor Brasil 6G [5].

Visando alcançar seus objetivos, este trabalho foi organizado em 6 seções. A Seção II apresenta uma visão geral do sistema de comunicação utilizado na avaliação da implementação. A Seção III define a métrica NMSE e sua aplicação no contexto do modelo do sistema. A Seção IV apresenta a segunda métrica a ser analisada neste trabalho, sendo ela um modelo de IA, utilizando uma Rede Neural. A Seção V apresenta o ambiente de simulação utilizado e as ferramentas desenvolvidas para a avaliação das métricas. A Seção VI apresenta as análises acerca das métricas citadas anteriormente. A Seção VII, e última, apresenta as conclusões do artigo.

II. VISÃO GERAL DO SISTEMA DE RADIOCOMUNICAÇÃO

O sistema de radiocomunicação empregado para a avaliação da implementação do algoritmo LDNN-SNRE é formado pelas etapas essenciais da camada física do transceptor Brasil 6G. Este é parte de uma plataforma de pesquisa e testes de determinadas tecnologias que podem vir a integrar a futura rede de comunicação móvel, conforme prevê o projeto Brasil 6G [5].

As etapas do sistema de radiocomunicação consideradas para integração e avaliação estão ilustradas na Figura 1 e

estão apresentadas nas subseções seguintes. Tais etapas foram desenvolvidas utilizando o *framework* GNU Radio [6] e considerando o conceito de SDR (*Software Defined Radio*).

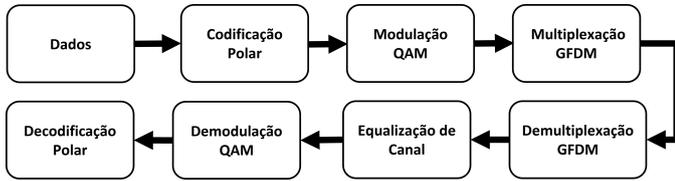


Fig. 1. Etapas principais que compõe o sistema de radiocomunicação.

A. Codificador e decodificador Polar

O código corretor de erro, FEC (*Forward Error Correction*), possui um importante papel dentro do sistema de radiocomunicação, ao viabilizar a detecção e correção de erros, impactando diretamente na qualidade do enlace de comunicação. O código FEC utilizado no sistema é o código Polar [7]. A implementação utiliza como base a biblioteca de código aberto AFF3CT [8] e provê suporte à diferentes taxas de codificação. A taxa de codificação a ser empregada em um determinado enlace de comunicação é definida com base no MCS e este, por sua vez, é definido a partir das condições do canal de comunicação.

B. Mapeador e demapeador QAM

O sistema de radiocomunicação empregado suporta os seguintes esquemas de mapeamento e demapeamento QAM (*Quadrature Amplitude Modulation*): QPSK (*Quadrature Phase Shift Keying*), 16-QAM, 64-QAM e 256-QAM. A definição de qual mapeamento será utilizado a cada instante é definido de forma semelhante à taxa do codificador, ou seja, a partir do valor de MCS.

O demapeamento QAM emprega o processo de decisão suave (*soft-decision*), para obter os valores de LLR (*Log-Likelihood Ratio*) associados a cada símbolo QAM demapeado. Uma versão de baixa complexidade do cálculo de LLR, baseado em [9], foi implementada visando reduzir o custo computacional.

C. Forma de onda GFDM

O GFDM (*General Frequency Division Multiplexing*) [10] é a forma de onda utilizada no transceptor Brasil 6G e empregada como base nos testes de validação da implementação algoritmo LDNN-SNRE. O GFDM é uma forma de onda com múltiplas portadoras, flexível, podendo ser parametrizada a fim de atender a diferentes aplicações e cenários, e que tem o OFDM (*Orthogonal Frequency Division Multiplexing*) como um caso particular de sua parametrização. Esta forma de onda foi adotada devido a mesma proporcionar uma reduzida emissão fora da faixa, OOB (*Out-of-Band Emission*), quando comparada ao OFDM. Sendo assim, ela torna-se adequada para aplicações que consideram o uso compartilhado do espectro com outros sistemas e tecnologias, como ocorre no caso de aplicações de TVWS (*TV White Space*).

D. Estimação de canal e equalização

A estimação de canal é realizada com base em portadoras pilotos que ocupam posições determinadas no espectro de transmissão, cuja informação transmitida é previamente conhecida e determinada por uma sequência *Zadoff-Chu*[11]. O valor recebido em tais portadoras são comparados com os valores da sequência a fim de computar as componentes de ganho e fase para cada subportadora e estimar, assim, as condições do canal. Tal estimativa, interpolada para a toda a banda do canal é empregada no processo de equalização a fim de mitigar os efeitos causados pelo canal de comunicação.

III. ESTIMAÇÃO DE SNR BASEADA NO NMSE

Nesta seção é introduzida a métrica base utilizada no transceptor Brasil 6G para a estimação de SNR. Esta estimativa é realizada na etapa de demodulação QAM e utiliza como base o cálculo do NMSE dos símbolos QAM recebidos. Tal aferição é de suma importância, dado que o sistema opera de maneira adaptativa e tem seu MCS definido em função deste valor. A implementação da referida métrica se dá por meio de um cálculo iterativo, o qual é realizado para todos os k símbolos QAM recebidos. O valor em dB da SNR estimada com base no cálculo do NMSE é dado pela seguinte equação:

$$\chi(m) = -10 \cdot \log \left[\frac{\sum_{i=0}^{k-1} (\hat{m}_i - m_i)^2}{\sum_{i=0}^{k-1} \hat{m}_i^2} \right] dB \quad (1)$$

onde χ é o valor da SNR estimada com base no NMSE, sendo este calculado, por sua vez, em função de m , conjunto dos símbolos recebidos, e de \hat{m} , os símbolos estimados após a demodulação QAM com decisão abrupta (*hard decision*).

IV. IMPLEMENTAÇÃO DO ALGORITMO DE ESTIMAÇÃO DE SNR BASEADO EM *Deep Learning*

O algoritmo de estimação de SNR denominado de LDNN-SNRE, detalhado em [4] e ilustrado na Figura 2, emprega duas etapas principais: cálculo da função MSE modificada e inferência da rede neural. A função MSE modificada estima a SNR utilizando como base o cálculo do MSE (*Mean Squared Error*), porém, com fatores de ponderação adicionais que foram propostos com objetivo de aumentar a exatidão da estimativa. Ainda assim, um modelo de DL (*Deep Learning*) é responsável por estimar um fator de correção complementar, o qual permite refinar a estimativa de SNR computada.

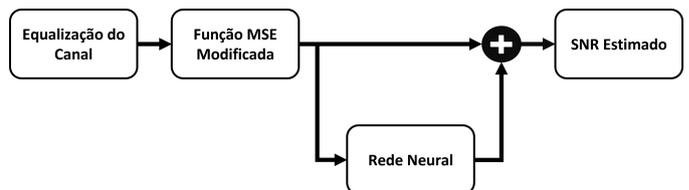


Fig. 2. Diagrama macro do algoritmo LDNN-SNRE.

Na presente seção apresenta-se uma introdução sobre a função MSE modificada, assim como detalhes do processo de criação, treinamento e validação do modelo da rede neural, realizados durante a implementação do algoritmo LDNN-SNRE.

A. Função MSE Modificada

A função MSE modificada ou MMSE (*Modified Mean Squared Error*) utiliza como base o cálculo de MSE dos símbolos QAM, considerando a demodulação com decisão abrupta. Tal função emprega fatores complementares ao MSE a fim de mitigar os erros que ocorrem quando as componentes reais e imaginárias dos símbolos recebidos ultrapassam os limites de decisão. A distância deste limite é dada por $d_{min}/2$, onde d_{min} é a menor distância entre dois símbolos adjacentes do diagrama da constelação. Porém, a ultrapassagem mencionada só pode ser detectada quando o módulo das coordenadas excede o limite máximo dado por $S_{max} + d_{min}/2$, onde S_{max} é o máximo valor, em módulo, que as componentes reais e imaginárias podem assumir na constelação utilizada. Sendo assim, S_{max} é definido em função dos símbolos da extremidade ou borda da constelação.

A ultrapassagem do limite de decisão de um símbolo só é explícita e identificável na condição que um símbolo recebido assume valores além da extremidade da constelação, ou seja, valores superiores a $S_{max} + d_{min}/2$. Um fator ponderador é introduzido nos cálculos visando compensar as referidas ultrapassagens que não foram identificadas de maneira explícita, as quais vão além do limite de decisão de um símbolo, mas não ultrapassam a extremidade máxima da constelação.

B. Dataset para o treinamento da rede neural

A fim de gerar o *dataset*, que é a base de dados a ser utilizada no treinamento da rede neural, realizou-se uma simulação no GNU Radio, cujo objetivo é associar os valores medidos pela função MSE modificada aos valores de SNR em teste e definidos para o canal de comunicação.

A geração dos dados ocorre em um processo iterativo, o qual foi executado para cada uma das modulações: QPSK, 16-QAM, 64-QAM e 256-QAM. A Figura 3 apresenta um diagrama em blocos da simulação criada para a geração do *dataset*. Vale ressaltar que a faixa de SNR considerada no treinamento é de -40 a 40 dB em passos de $0,1$ dB. Dito isso, para cada valor desta faixa, amostras de símbolos são aleatoriamente geradas e mapeadas de acordo com as respectivas constelações. Os valores então mensurados pelo bloco da função MSE modificada são registrados em um arquivo que compõe o *dataset* criado.

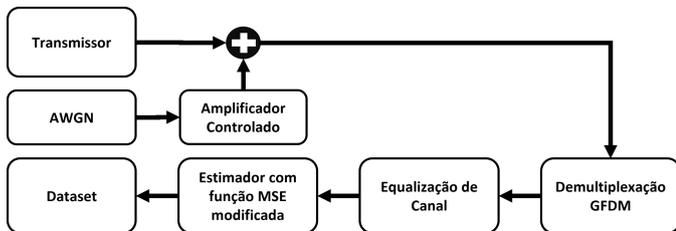


Fig. 3. Diagrama da simulação de geração do *dataset*.

C. Treinamento da rede neural

Para a criação e treinamento do modelo da rede neural, utilizou-se a biblioteca Keras, que é uma API (*Application*

Programming Interface) de alto nível, que facilita a realização de tal processo. O modelo criado é considerado de baixa complexidade, em virtude da utilização de apenas quatro camadas, sendo duas delas ocultas. Ainda assim, utilizou-se a MSE como função de custo, além do ADAM (*Adaptive Moment Estimation*) como algoritmo de otimização. Por fim, para avaliação do modelo durante o processo de treinamento, é utilizada a precisão como métrica de verificação. A Figura 4 apresenta o modelo criado, bem como as descrições das camadas e respectivas funções de ativação.

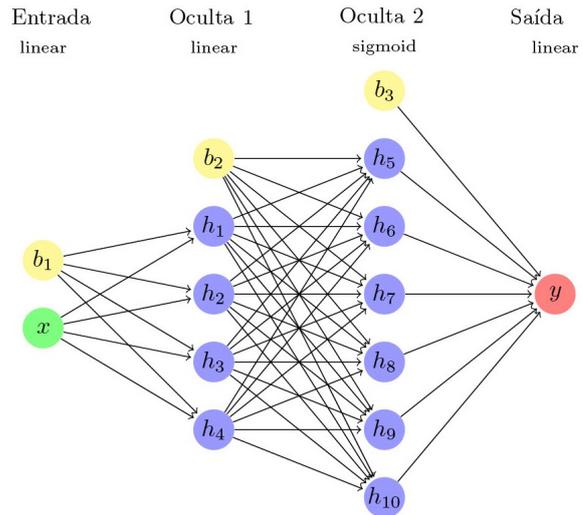


Fig. 4. Estrutura da rede neural criada.

D. Validação do modelo

A Figura 5 ilustra a resposta de um dos modelos criados a partir conjunto de dados apresentados durante o treinamento considerando a modulação 256-QAM. Pode-se observar que a estimação gerada a partir do modelo converge para os valores dos dados, evidenciando a capacidade do modelo em estimar a relação não linear subjacente entre as variáveis. De forma semelhante, os modelos criados para as diferentes modulações foram avaliados e validados.

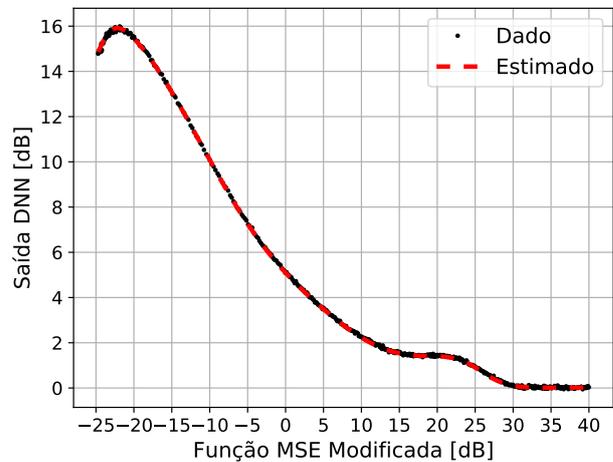


Fig. 5. Resposta de estimação de SNR do modelo criado (256-QAM).

V. AMBIENTE DE INTEGRAÇÃO E AVALIAÇÃO DE DESEMPENHO

Com intuito de viabilizar a avaliação de desempenho do algoritmo de estimação de SNR baseado em DL desenvolvido, foi realizada a integração do mesmo com um sistema de radiocomunicação, o qual é uma versão simplificada do transceptor Brasil 6G, conforme descrito na Seção II. Como este sistema foi implementado no ambiente GNU Radio, foi necessário portar o algoritmo de estimação de SNR desenvolvido para este mesmo ambiente, a fim de viabilizar sua integração e teste.

A criação de um ambiente e avaliação de desempenho do algoritmo, desenvolvido e integrado ao sistema de radiocomunicação citado, exigiram a criação de etapas específicas de processamento no GNU Radio. Uma destas etapas é o próprio algoritmo de estimação proposto. A seguir estão apresentados detalhes do ambiente de integração e avaliação desenvolvido.

A. Amostrador

Este bloco, desenvolvido em Python, realiza a amostragem dos valores da SNR estimados pelas métricas: NMSE, função MSE modificada e o valor corrigido pela rede neural. Além disso, este também é responsável pelo controle da SNR do canal, através do envio de uma mensagem PMT (*Polymorphic type*), contendo este valor, para o bloco de ganho controlado. O funcionamento da unidade de processamento se baseia em uma máquina de estados, cujos eventos são apresentados nos subtópicos a seguir:

- **Mudança de SNR:** A realização da amostragem só poderá ser iniciada caso ocorra uma mudança da SNR do canal. Para que essa alteração aconteça, a simulação não deve ter atingido o valor máximo pré-definido.
- **Intervalo de espera:** Uma vez que foi computado e enviado o valor de SNR para a unidade de ganho controlado, um intervalo de espera é definido e, após isso, o processo de amostragem pode ser iniciado. Isso se faz necessário para que nenhuma das métricas seja impactada por um atraso, realizando-se uma análise sem interferência de condições anteriores.
- **Amostragem:** Esta etapa tem seu funcionamento definido a partir de dois parâmetros: a quantidade de amostras e a frequência de amostragem. Tais valores determinam, diretamente, o tempo total em que as amostras serão coletadas para cada valor de SNR configurada no canal de comunicação. A Eq.2 apresenta como é computado o k -ésimo valor médio de SNR para as n amostras coletadas em cada uma das métricas:

$$\gamma_k = \frac{1}{n} \sum_{i=0}^{n-1} \beta_i \quad (2)$$

onde β_i é o i -ésimo valor de SNR estimado pela métrica.

- **Final:** O estado final é atingido quando não há mais valores de SNR a serem tratados, sendo as informações obtidas salvas em seus respectivos arquivos. Os valores armazenados serão analisados na seção VI deste artigo.

B. Amplificador Controlado

Como a simulação utiliza um canal AWGN (*Additive White Gaussian Noise*), esta unidade tem um funcionamento simples e bem definido. Conforme citado anteriormente, o bloco de amostragem define o valor de SNR do canal. Desenvolvido em Python, seu objetivo é atuar como um amplificador do ruído gaussiano através do valor da SNR recebida pela porta de mensagem.

C. Estimador baseado em LDNN-SNRE

A implementação do algoritmo LDNN-SNRE é feita através da criação de um bloco embutido no GNU Radio, utilizando a linguagem Python. Como o treinamento dos modelos foi realizado de maneira *off-line*, estes e seus respectivos pesos são carregados no processo de inicialização do bloco. Além disso, outras funcionalidades são incluídas, como: cálculo da função MSE modificada e predição para cada um dos modelos. Em seu funcionamento, após a etapa de equalização do canal, os símbolos recebidos são utilizados pela função de erro implementada a fim de estimar o valor de SNR. A partir disso, este valor computado é utilizado como entrada para a rede neural, a qual realizará a estimativa da correção necessária, permitindo atingir a SNR configurada para o canal.

VI. ANÁLISE DE DESEMPENHO

A análise e compreensão das métricas informadas foi realizada através de uma simulação no *software* GNU Radio, utilizando os blocos e cadeia de comunicação previamente abordados. O amostrador foi configurado para operar a uma frequência de amostragem de 50 Hz, coletando 10 mil amostras durante o processo de aquisição de dados. Tal unidade de processamento torna-se responsável por amostrar os valores de NMSE, função MSE modificada e o valor corrigido através da Rede Neural implementada, por 200 segundos para cada valor de SNR configurada para o canal.

A faixa de valores de SNR empregada durante o processo foi de -10 dB a 40 dB. Vale ressaltar que o ruído gaussiano é inserido após a geração dos símbolos GFDM e as estimativas são realizadas após o processo de equalização do canal. Dessa maneira, a Figura 6 ilustra a disposição lógica dos blocos, omitindo alguns processos da cadeia de comunicação, que já foram citados na seção II.

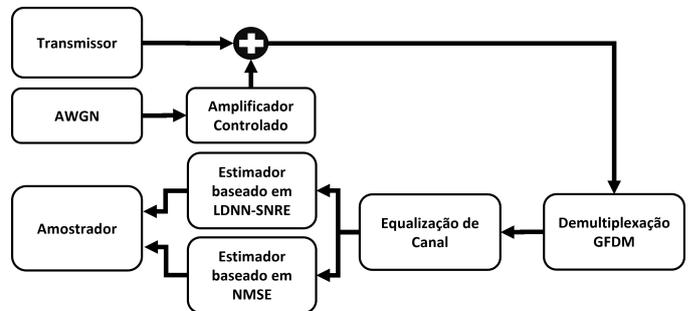


Fig. 6. Diagrama da Simulação.

A simulação foi executada com código MCS fixo e em seu maior valor, ou seja, o enlace operando com a modulação 256-QAM e taxa de codificação mínima. A Figura 7 apresenta os

resultados obtidos por este processo, além da constelação de recepção com SNR em 27 dB. Os valores estimados por cada uma das métricas estão identificados no gráfico.

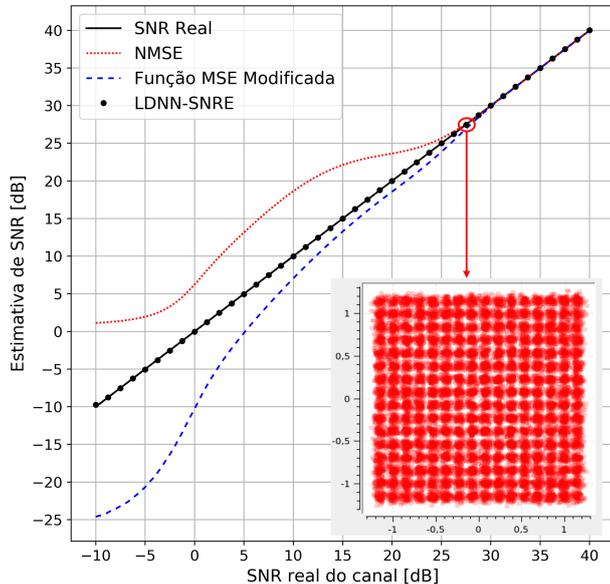


Fig. 7. Resultado da análise de desempenho para 256-QAM.

Com os resultados obtidos através simulação realizada, observa-se que a NMSE apresenta estimativas imprecisas para baixos valores de SNR. Torna-se evidente, ao analisar a curva desta métrica, que a convergência de sua estimativa à SNR configurada para o canal de comunicação ocorre a partir de 27 dB. Dessa maneira, para valores abaixo deste citado, o comportamento desta métrica apresenta valores que superestimam os valores reais. Por outro lado, ao analisar a métrica da função MSE modificada, observa-se uma maior precisão quando comparado ao NMSE. Logo, ao utilizar um fator ponderador, o qual compensa os eventos internos não detectáveis e incidentes dentro dos limites máximos da constelação, obtém-se estimativas que se aproximam ao valor real em um intervalo maior de valores de SNR.

Por fim, ao utilizar o modelo de DL, verifica-se a obtenção de estimativas convergentes ao valor real da SNR do canal de comunicação. Dito isso, a correção realizada pela rede neural para o valor obtido pela função MSE modificada mostra-se eficaz mesmo com valores baixos de SNR. Desta forma, a SNR obtida por meio do algoritmo LDNN-SNRE permite que a definição do valor de MCS a ser empregado no enlace de comunicação seja realizada de forma mais assertiva e otimizada, assegurando, assim, o QoS.

VII. CONCLUSÃO

Este trabalho apresentou uma abordagem de implementação do algoritmo LDNN-SNRE, bem como sua integração e validação em um sistema de radiocomunicação baseado no transceptor Brasil 6G. Por meio de um ambiente de testes desenvolvido, foi possível realizar uma comparação direta entre as estimativas de SNR obtidas através do cálculo do NMSE e da implementação do LDNN-SNRE. Pode-se concluir que esta técnica, que emprega um modelo de DL, apresentou

maior precisão quando comparado ao NMSE. Dessa maneira, a implementação desenvolvida pode ser utilizada, para melhorar a eficiência de enlaces de radiocomunicação, uma vez que a estimativa de SNR é utilizada como base para definir parâmetros primordiais, como o MCS.

Aliado a tais fatores previamente citados, os resultados alcançados evidenciam a capacidade dos modelos de DL em lidar com dados complexos e realizar inferências precisas, desde que se tenha um *dataset* adequado para o treinamento do modelo. O uso de DL para estimação de parâmetros do canal de comunicação se torna viável e promissor para as futuras redes de comunicação, como o 6G, visto que, em muitos casos, é complexo encontrar uma solução analítica ou algorítmica para a estimação de parâmetros envolvidos nos enlaces de comunicação.

AGRADECIMENTOS

Este trabalho foi desenvolvido em prol dos projetos Samurai, Brasil 6G e TVWS NIC.BR. Os autores agradecem a FAPEMIG, CNPq-Brasil, FAPESP, BRITISH GOVERNMENT DIGITAL ACCESS PROGRAMME (DAP) e NIC.BR, além do Instituto Nacional de Telecomunicações pelos recursos e apoio financeiro para sua realização.

REFERÊNCIAS

- [1] G. Wikström, J. Peisa, P. Rugelund, N. Johansson, S. Parkvall, M. Girnnyk, G. Mildh, and I. L. Da Silva, "Challenges and technologies for 6g," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*, 2020, pp. 1–5.
- [2] A. M. Khan, V. Jeoti, M. Rehman, and M. Jilani, "Noise power estimation for broadcasting ofdm systems," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2017, pp. 1–6.
- [3] T. Ngo, B. Kelley, and P. Rad, "Deep Learning Based Prediction of Signal-to-Noise Ratio (SNR) for LTE and 5G Systems," in *2020 8th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, 2020, pp. 1–6.
- [4] R. Kagami and L. Mendes, "A low-complexity deep neural network for signal-to-interference-plus-noise ratio estimation," in *Anais do I Workshop de Redes 6G*. Porto Alegre, RS, Brasil: SBC, 2021, pp. 1–6. [Online]. Available: <https://sol.sbc.org.br/index.php/w6g/article/view/17227>
- [5] Projeto Brasil 6G, "Contribuições para a Camada Física de Redes 6G," 2022, [Online]. Disponível em: <https://inatel.br/brasil6g/brasil-6g/336-brasil-6g-meta-3-atividade-3-1-camada-phy/file>. Acessado em: 28 de abril de 2023.
- [6] GNU Radio, "GNU Radio Web Site," 2023, [Online]. Disponível em: <http://www.gnuradio.org>. Acessado em: 28 de abril de 2023.
- [7] M. S. Arti Sharma, "Polar code appropriateness for ultra-reliable and low-latency use cases of 5g systems," *International Journal of Networked and Distributed Computing*, vol. 7, p. 93–99, 2019.
- [8] M. L. K. H. C. L. R. T. O. A. D. B. T. T. V. P. e. a. A. Cassagne, O. Hartmann, "Aff3ct: A fast forward error correction toolbox!" *SoftwareX*, vol. 10, 2019.
- [9] I. Ali, U. Wasenmüller, and N. Wehn, "A high throughput architecture for a low complexity soft-output demapping algorithm," *Advances in Radio Science*, vol. 13, pp. 73–80, 2015. [Online]. Available: <https://ars.copernicus.org/articles/13/73/2015/>
- [10] N. Michailow, M. Matthé, I. S. Gaspar, A. N. Caldeilla, L. L. Mendes, A. Festag, and G. Fettweis, "Generalized frequency division multiplexing for 5th generation cellular networks," *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3045–3061, Set. 2014.
- [11] D. Chu, "Polyphase codes with good periodic correlation properties (corresp.)," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 531–532, 1972.