1. I~~i~~ntroduction ~~There are two reasons why deep learning is used.~~

With its recent development ~~of deep learning~~, deep learning ~~It~~ has shown great achievements ~~in~~ when applied to ~~other~~ areas outside the computer science domain. Therefore, ~~it~~ deep learning ~~is~~ has also been applied to physical layer communication research such as channel coding and~~,~~ ~~MIMO(Multi Input Multi Output)~~ antenna technology. There are two reasons why deep learning is used ~~to~~in communication systems~~.~~ ~~Deep learning is has recently been applied to communication research for the following two main reasons:~~ ~~P~~performance improvement and ~~, Reduced~~ complexity reduction~~:~~. ~~The~~Based upon the first reason, ~~is to apply~~ deep learning technology with decoding is applied to communication systems ~~with decoding~~ to achieve better performance. Before the integration of deep learning, ~~First, the goal of communication is that receiver must receive the message accurately, but~~ noise, which is added when ~~the~~ a message passes through ~~the~~ a channel,~~;~~ can ~~so that~~prevent the receiver ~~may not~~ from receiving~~e~~ the message. To improve this, there is a process called encoding and decoding in the classical communication domain. ~~–~~ There~~The first reason is to apply deep learning technology with decoding to achieve better performance.~~ ~~Second~~ is~~,~~ a ~~p~~Popular decoding method, known as the Belief Propagation Algorithm (BPA), ~~or the other~~also named the Sum Product Algorithm (SPA). This algorithm ~~,~~ has good performance, but it ~~consists of~~is composed of many multiplication operations. Therefore, the longer the length of the message used as an input value, the more complicated calculation is. ~~At that~~In such circumstances~~ time~~, the second reason to use deep learning, ~~is~~ reducing complexity is important. To ~~improve this complexity~~solve th~~is~~e complexity problem, there is a solution called the min-sum algorithm (MSA). In this ~~this~~ min-sum algorithm, the complexity problem was improved, but performance loss degradation occurred. To properly adjust this trade-off relationship ~~of~~ between performance and complexity, there are two algorithms~~:.~~ 1.~~First,~~ The ~~a n~~Normalized min-sum algorithm (NMSA) that multiplies correction factor value, which is a constant value, from the check node update process~~.~~ 2.~~ Second,~~ The ~~an o~~Offset min-sum algorithm (OMSA) that adds or subtracts correction factor value from check node update process. By incorporating deep learning in this way into existing communication systems, the correction factor is optimized. Remarkably, ~~r~~Results from the use of these algorithms showed better performance than MSA and improved complexity than BPA, respectively. This is a ~~remarkable~~ point~~can be~~. ~~The reason for use of deep learning these systems is that it wants to improve performance can be further improved. more than before by incorporating deep learning into existing communication system.~~

서식 있음: 글꼴: 굵게, 글꼴 색: 빨강
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
메모 포함[NP1]: Is this an example?
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 빨강
메모 포함[c2]: NMSA, OMSA is method, so I think i ...
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
서식 있음: 글꼴 색: 텍스트 1
메모 포함[NP3]: If you wanted to you could shorten ...
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 빨강
서식 있음: 글꼴 색: 텍스트 1

<related work>

Recently, many researchers have been actively researching methods to incorporate deep learning into channel coding. A formative study, by Nachmani,~~,~~ ~~channel coding's prior researcher,~~ used deep learning in the decoding process.~~,~~ ~~and by~~ By setting different w~~i~~e~~i~~ght values at the edges connecting check nodes(CN) and variable nodes(VN), [1] improved performance ~~was improved~~ by reducing the effect of small cycles in tanner graphs during the decoding process~~, and it is very meaningful in that it used deep learning for first time~~ ~~[1]~~. In [2], unlike deep learning's ~~application~~ applied to BPA, Lugosch applied it to OMS. ~~did a study about~~ OMS. Deep Learning ~~I~~ ~~The OMS algorithm~~ was used to obtain OMS's optimized correction factor value. More computations result in more complexity, more load, higher hardware device~~s~~ temperatures, and permanent performance degradation. Therefore, to improve this, complexity is one of the problems that hardware must solve. Particularly, OMS is an algorithm consisting of addition and subtraction, not a more complex multiplication calculation, ~~a complex calculation,~~ so it is a suitable method for an algorithm to hardware because of its low complexity. This algorithm is called a neural offset min-sum (NOMS). ~~In the case of~~ [3]~~,~~ ~~which~~ conducted ~~a~~ similar research where a~~,~~ neural normalized min-sum (NNMS) was proposed using an optimized correction factor ~~by~~ through deep learning. ~~And~~ Alternatively, Wang et. al suggested ~~another~~ a different ~~way~~method. In order to improve the complexity problem, ~~he~~ they used a sharing ~~shared~~ method that uses the same correction factor value for each iteration, unlike recent ~~one~~ studies that used different correction factor values for each iteration and node. This algorithm is called a Shared Neural NMS (SNNMS).

The aforementioned ~~r~~Research~~es~~ ~~mentioned~~ used deep learning ~~for~~ to optimizing correction factor. However, there is a study focusing on refining the deep learning architecture for this application. Deep Learning has several architectures such as Deep Neural Network (DNN), Convolutional Neural Network(CNN), and Recurrent Neural Networks(RNN) ~~and so on~~. ~~In~~ [4] ~~, it~~ is a seminal ~~research~~ work using 'RNN'. ~~It is~~ called a 'circular neural network'. ~~and~~ This RNN ~~is a deep learning architecture that~~ utilizes past data for learning through the concept of a recurrent. In other words, it is an algorithm that utilizes not only current inputs but also past data for learning. Remarkably, ~~t And, these research is remarkable. Because it~~ this ~~was~~ first study to incorporate 'RNN' into the decoding process~~, and~~ showed similar performance to ~~previous~~ prior stud~~y~~ies using fewer parameters. Subsequently, researcher improved performance by incorporating "relaxation" into the RNN architecture in [5]. The ~~P~~purpose of using this relaxation concept was to determine how much previous data to use. Notably, this method optimized the decoder relaxation factor through deep learning as opposed to the previous method, brute force simulation.

서식 있음: 글꼴: 굵게, 글꼴 색: 빨강

메모 포함[NP4]: Instead of just listing, compare these works. What was done better, or worse or was improved upon.

메모 포함[NP5]: Use a reporting verb instead of "did a study"

메모 포함[NP6R5]: E.g. Lugosch examined the application of an OMS algorithm on a ....

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

메모 포함[NP7]: Good.

서식 있음: 글꼴 색: 빨강

메모 포함[NP8]: "for an algorithm to hardware". The meaning of this phrase is unclear.

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

메모 포함[c9]: [4] : use RNN architecture -> [5] : use RNN + relaxation.

메모 포함[NP10R9]: Got it.

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 빨강

서식 있음: 글꼴 색: 빨강

However, ~~this~~ 'RNN' ~~-based method~~ has two limitations. To be specific, input vectors are entered sequentially to enable sequential data processing, but 'parallelization operation' is not possible. ~~And~~ The derivative value of tanh, activation function of RNN, is used in this case. However, there is a disadvantage that back propagation information is rarely transferred because a vanishing gradient occurs. [illegible] performance ~~excellence of~~ from the ~~our~~ proposed method, simulations were conducted in BCH code, which is a ~~high density~~ high-density parity check code (HDPC) and low density parity check code (LDPC) with different lengths. ~~And~~ Also, the ~~s~~ S ~~-~~ emiconductor company 'NVIDIA' recently announced an open source simulator '~~s~~Sionna' [6] to help research 5G communication using deep learning. In channel coding, there are many methods to compare performance, and I used Bit Error Rate (BER), which is number of bits that have errors in the process of being transmitted to the number of [illegible] and it will be shown that performance is better when using our proposed method than when using RNN.~~), which is number of bits that have errors in the process of being transmitted to the number of bits received.~~

서식 있음: 글꼴: 굵게, 글꼴 색: 빨강

메모 포함[c11]: Proposed method -> our method

서식 있음: 글꼴 색: 텍스트 1

메모 포함[NP12]: Past tense used since this is referencing paper [5].

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

메모 포함[NP13]: Run on sentence.

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

메모 포함[NP14]: Run on sentence.

메모 포함[NP15]: In general, better. But I still want you to close with a summary the potential outcomes or applications of your work.

메모 포함[NP16R15]: However, you can submit this as is (after the changes I've made in this iteration of the writing).

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴 색: 텍스트 1

서식 있음: 글꼴: (영어) Helvetica, (한글) 굴림, 12 pt, 글꼴 색: 텍스트 1

<reference>

[1] Nachmani, Eliya, Yair Be'ery, and David Burshtein. "Learning to decode linear codes using deep learning."2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton). IEEE, 2016.

[2] Lugosch, Loren, and Warren J. Gross. "Neural offset min-sum decoding." 2017 IEEE International Symposium on Information Theory (ISIT). IEEE, 2017.

[3] Wang, Qing, et al. "A model-driven deep learning method for normalized min-sum LDPC decoding." 2020 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, 2020.

[4] Nachmani, Eliya, et al. "RNN decoding of linear block codes~~.~~," arXiv preprint arXiv:1702.07560(2017).

[5] Nachmani, Eliya, et al. "Deep learning methods for improved decoding of linear codes." IEEE Journal of Selected Topics in Signal Processing 12.1 (2018): 119-131.

[6] Hoydis, Jakob, et al. "Sionna: An Open-Source Library for Next-Generation Physical Layer Research." arXiv preprint arXiv:2203.11854 (2022).