AI and Physical Layer Communications

Xiang-Gen Xia
University of Delaware

Is artificial intelligence (AI) truly useful in physical layer communications? The answer was obvious before 2015. AI was popular about 30 years ago and has become popular again after Google's computer player AlphaGo winning of Go match in 2015. Nowadays, AI is popular in every corner of the world. Of course, people would say AI is useful in physical layer communications. If my answer was no, I would have to bear a huge burden of offending many people, which I did not mean to, although.

In physical layer communications, there are three parts that are transmitter, channel, and receiver. At the transmitter side, AI had not found killer applications in source, channel coding, or modulations in the past. Source coding was one of the most active subjects in the 1990s and almost every institute had multiple research groups working on image and video compressions. AI was one of the tools researchers had tried intensively but unfortunately in terms of compression ratio given the same image/video quality fidelity, it was not better than other image/video compression techniques, such as those used in JPEGs and MPEGs, discrete cosine transform (DCT) and discrete wavelet transform (DWT) etc., while its computational complexity was much higher. In speech compression applications, the standard and well-used technique is code-excited linear prediction (CELP), although AI (deep learning) has its very successful applications in speech/language recognitions.

For channel coding, AI does not provide a method to find a good code, while the optimal encoding is basically random coding that is opposite with AI philosophically. On the other hand, the optimal modulations have been well understood in the past. Talking about combined coding and modulation, i.e., trellis coded modulation (TCM), searching the optimal TCM schemes is complex and not solved. If AI is treated as a method of optimization, one may apply AI but it is not sure whether it is better than the existing TCM designs. I do not see any rational for AI to be applicable for this problem.

The physical channel part is probably the part where most researchers think that AI may be useful in physical layer. However, I think that this part is one of the most clear parts where AI is not useful. A physical channel has two important parts one may want to pay attention to. One part is the multipaths that exist in either wireline or wireless channels and make a (broadband) channel to have intersymbol interference (ISI). I always think that dealing with ISI is one of the most important tasks in the past decades in communications engineering of either wireline, such as computer modem, or wireless systems, such as cellular and WiFi systems. Since an ISI/multipath channel is well modeled as a linear system, one does not need to use AI as a kind of nonlinear approximation to approximate a linear system, and the current channel estimation

methods are already optimal. It does not matter for a single input single output (SISO) or multiple input multiple output (MIMO) system.

The other is the channel along time, i.e., the fading channel. In this regard, a channel (or a path) may be a function of time. For a wireline channel, since the physical hardware is fixed, no motion is involved, the channel is fixed as well, i.e., in this case the channel is constant. So, no AI is needed. For a wireless channel, if no terminals move, such as a WiFi channel, the channel is usually stationary and does not change along the time. Thus, no AI is needed either. For a cellular channel, it may have motion. In this case, the channel is time varying. An underwater acoustic communications channel is also time varying. Of course, if one can track and predict the channel very accurately and quickly in a reasonably long time, it may save the training overhead. The question is: whether AI can provide an accurate and quick channel prediction over multiple data frames? Does it have fast and sufficient enough training samples or stationary enough environment available? Or is it worthwhile using AI instead of a simple training in every frame in such an application? I believe that this had been negatively studied well in the 1990s already.

Furthermore, for AI based techniques, they usually use training, where true channel information is used to train. However, no one knows a true channel in practice and a channel has to be estimated by using pilot signals anyway. An estimated channel may be different from the true channel. Thus, the training assumption in AI based techniques is problematic.

In the receiver side, one is channel acquisition that is already mentioned earlier, and the other is decoding/demodulation. Before turbo and LDPC codes, the channel decoding for wireless communications was mainly Viterbi decoding for convolutions codes, which is already optimal, i.e., the maximum-likelihood (ML) decoding. For turbo and LDPC codes, although the true ML decoding is too complex, the iterative decoding achieves a good tradeoff between performance and complexity. Now some people may say that, since the ML decoding is too complex, particularly when an ISI channel is combined in consideration, AI based decoding may perform better than the existing iterative decoding, since AI may be thought of as an optimization method. Furthermore, if an AI based method can be used in decoding with better performance and affordable complexity, it may not be necessary to use turbo or LDPC code, and a random code may be used. Below let me elaborate it more.

The reason why the above ML decoding is not used currently is mainly because of the high computational complexity. One can treat AI as a kind of optimization tool as follows. If all the parameters in an optimization problem are searched optimally, it may be too complex. Now the idea is to decompose the whole parameter space to some smaller parameter subspaces and some of them are used to train (or learn in the language of AI) and find the optimal one. The found optimal one is then used to predict the other parameters in other subspaces. Then, a natural question is: is it better than other well-used suboptimal methods, such as fixing all the parameters in one set, optimize the parameters in another smaller set, then alternate the optimizing parameters iteratively. In fact, the M-algorithm in convolutional code decoding is

already using a smaller searching set. Can AI based algorithms do better than these? My answer would be no.

AI is an area of research and has applications in some areas, such as speech recognition and computer vision etc. But any technique cannot be universal and has its limitations. So is AI.