OFDM Equalize Symbols module

In IEEE 802.11a/g/p, the OFDM Equalize Symbols module at the receiver's core function is to correct channel distortions (like multipath fading) by using known pilot signals (reference signals) to estimate the channel's effect, then applying a complex scalar multiplication to each data subcarrier, effectively "undoing" the channel's impairment to recover the original transmitted data symbols. It turns frequency-selective fading into flat fading for each subcarrier, making demodulation easier and ensuring data integrity.

How it works:

1. Channel Estimation: The receiver identifies known pilot symbols embedded in the OFDM frame, which are inserted by the transmitter at specific subcarriers.
2. Channel Model Derivation: By comparing the received pilot signals with the known transmitted pilot signals, the module estimates the channel's complex frequency response (gain and phase shift) across different subcarriers.
3. Equalization (Complex Multiplication): For each subcarrier, the estimated channel response is used to perform a complex scalar multiplication on the corresponding received data symbol.
4. Symbol Recovery: This process effectively removes the channel's distortion, yielding an estimate of the original transmitted symbols, which are then sent for demodulation and decoding.

Why it's crucial for 802.11a/g/p:

- Combats Multipath: Wireless channels experience reflections (multipath), causing delayed, faded copies of the signal, leading to Intersymbol Interference (ISI).
- Subcarrier-Specific Correction: OFDM divides data across many narrow subcarriers; equalization treats each subcarrier somewhat independently, simplifying correction.
- Robustness: This module is vital for maintaining high data rates in dynamic environments like Wi-Fi (802.11a/g) and Vehicle-to-Everything (V2X) communications (802.11p).

Channel Estimation

Besides the phase, also the magnitude of the carriers has to be corrected, which is also performed by the OFDM Equalize Symbols block. This is especially important if QAM-16 or QAM-64 encoding is utilized, where also the magnitude carries information. Non-linearities in the magnitude might be caused by imperfect channel filters in the hardware. The current implementation of our block assumes the magnitude of the carriers to be sinc-shaped and corrects based on that assumption. However, this shape could be seen to depend also on the sender, as we experienced differences when we using different transmitters. Thus, this equalization needs some further improvement, as it currently restricts our receiver to BPSK and QPSK modulations. This block also removes DC, guard and pilot subcarriers and thus subsets the 64 symbol input vector into 48 symbols.

Limitation of the OFDM Equalize Symbols Module to BPSK and QPSK

The current implementation of the *OFDM Equalize Symbols* module performs equalization based on a simplified constellation-based decision rule. This approach is adequate for BPSK and QPSK, where the decision boundaries are linear and the constellation points are evenly spaced on the unit circle. However, the equalizer does not incorporate the more complex decision regions required for higher-order modulations such as 16-QAM or 64-QAM, which require amplitude-dependent scaling and more sophisticated channel compensation. As a result, the receiver is restricted to demodulating only BPSK and QPSK subcarriers.

Additionally, the module removes DC, guard, and pilot subcarriers from the 64-point OFDM symbol. After discarding these non-data subcarriers, the equalizer reduces each symbol vector from 64 subcarriers to 48 data-bearing subcarriers, which are then forwarded to the demodulation stage. This preprocessing is essential for isolating the useful information but also reflects that the current equalizer is tailored to lower-order constellations and requires further development to support higher-order modulation schemes.

Signal Field Decoding

The OFDM Decode Signal. In each frame, the short and long training sequences are followed by the signal field, which is a BPSK modulated OFDM symbol encoded with a rate of 1/2 that carries in formation about the length and encoding of the following symbols. Again, the start of the frame and, thus, the posi tion of the signal field is tagged in the sample stream. For decoding of the convolutional code, the IT++ library is used. If the signal field is decoded successfully, i.e., if the rate f ield contains a valid value and if the parity bit is correct, OFDM Decode Signal annotates the sample stream with a tag, carrying a tuple of encoding and length of the frame. This tag is used by the following block to decode the payload