物联网第三次作业报告

软73 沈冠霖 2017013569

1.脉冲调制解调

我按照作业给定的参数和iotbook的流程进行操作，我的代码都在pulse.py，具体流程如下：

调制部分，每遇到一个0，就增加一个10ms正弦脉冲和10ms空白；每遇到一个1，就增加一个10ms正弦脉冲和20ms空白。为了计算最后一个信号，我在调制的最后增加了一个10ms正弦脉冲。

解调部分，我首先进行[19.5kHZ, 20.5kHZ]的带通滤波，之后进行窗口大小为480（采样频率48000HZ * 时长10ms = 480个点）的滑动窗口傅立叶变换，然后对变换结果进行窗口大小为11的滑动平均滤波，之后用窗口大小480的滑动窗口法求得脉冲峰值位置，通过计算脉冲峰值的间隔来计算原信号，误差阈值为+-10%。

我设置初始信号为随机的0-1序列，选取长度为100，300，1000，每个长度随机测试3次，结果如下：

测试次数\序列长度	100	300	1000
1	正确	正确	正确
2	正确	正确	正确
3	正确	正确	正确

综上所述，因为多次重复测试全都正确，我的脉冲调制解调实现正确。

2.QPSK相位调制解调

我按照作业给定的参数和iotbook的流程进行操作，我的代码都在phase.py，具体流程如下：

调制部分，我先把0-1序列（需要是偶数个，否则我会末尾补一个0）转化为四进制序列，求得i和q序列，之后我生成每个阶段的波形

$$s(t) = i * sin(2\pi ft) + q * cos(2\pi ft)$$

将每个阶段波形组合就能得到整体波形。

解调部分，我将波进行离散化黎曼积分（根据每个采样点的值和对应函数值求和）求得i和q的数值

$$\begin{aligned} i &= 2\int_0^{\frac{1}{f}} (s(t) * sin(2\pi ft)) \\ &= \frac{2}{T*f}\int_0^{T} (s(t) * sin(2\pi ft)) \\ &= \frac{2}{T*f}\sum_{k=1}^{T*f_s} (s(k/f_s) * sin(2\pi fk/f_s) * \frac{1}{f_s}) \end{aligned}$$

$$q = 2\int_0^{\frac{1}{f}}(s(t)*cos(2\pi ft))$$
$$= \frac{2}{T*f}\int_0^{T}(s(t)*cos(2\pi ft))$$
$$= \frac{2}{T*f}\sum_{k=1}^{T*f_s}(s(k/f_s)*cos(2\pi fk/f_s)*\frac{1}{f_s})$$

之后根据i和q的序列还原四进制序列，进而还原0-1序列。

我设置初始信号为随机的0-1序列，选取长度为100，300，1000，每个长度随机测试3次，结果如下：

测试次数\序列长度	100	300	1000
1	正确	正确	正确
2	正确	正确	正确
3	正确	正确	正确

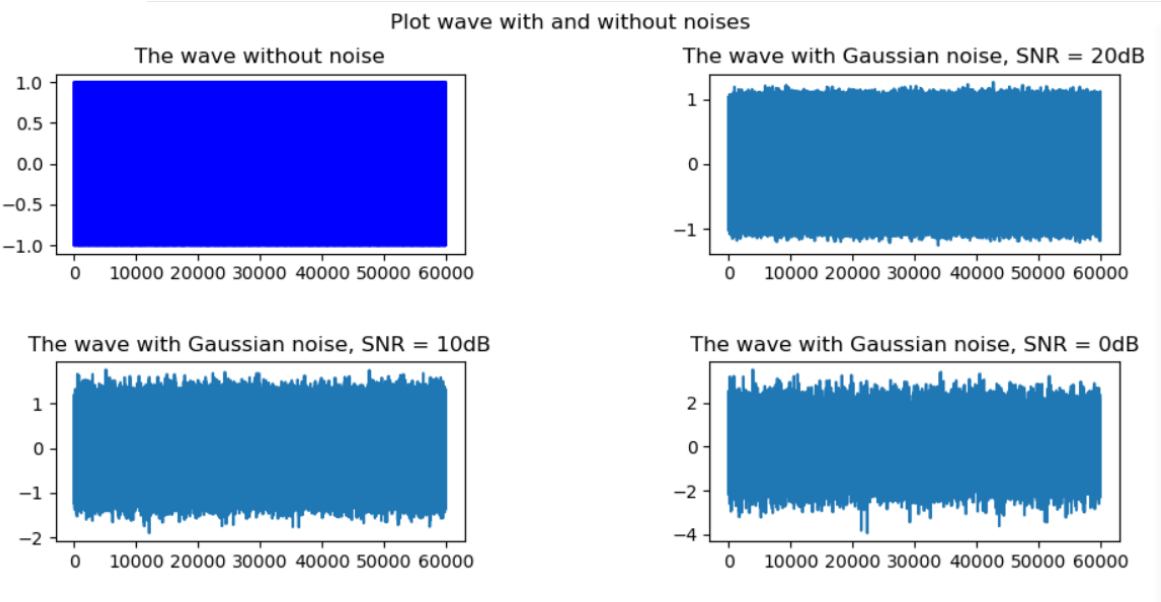
3.噪声

我生成噪声的函数在noise.py下

我计算原信号的均值mean和标准差std，计算得到噪声的均值和方差,生成噪声，再和原信号混合。

$$\mu_N = \mu_s$$
$$\sigma_N = \frac{\sigma_s}{10^{\frac{SNR}{20}}}$$
$$N = \sigma_N * N(0,1) + \mu_N$$

有无噪声的QPSK调制信号对比如下



根据图示可以看出，信噪比越小，整体信号的大小越大，而且波形越尖锐，说明实现基本正确。

我设置初始信号为随机的0-1序列，选取长度为100，300，1000，每个长度随机测试3次，结果如下：

信噪比\序列长度	100	300	1000
20	100%	100%	100%
10	100%	100%	100%
0	100%	100%	100%

可以看出，QPSK算法非常鲁棒。为何如此？我认为，因为i和q只有sqrt（2）和-sqrt（2）两种取值，因此我求得积分之后，可以直接用积分结果的符号来代表i和q，也就是如果积分结果大于0，i/q就是sqrt（2），否则是-sqrt（2）。这样就给积分提供了巨大的容错空间，很难被噪声干扰。

4.总结

这次作业，我了解了脉冲和相位调制解调方法，同时复习了量化和采样，还有傅立叶分析，滤波等技巧。我了解到，相比脉冲调制解调，相位调制解调的QPSK算法更加高效和鲁棒，实现也更为容易。同时，我也了解到，信号处理领域因为要考虑计算机数值计算的溢出、浮点数精度以及实际传输的噪声问题，对算法鲁棒性有较高的要求，也需要一些滤波方法来进行预处理，才能实现很好的效果。