

light_data_3.10/result/3.14:

(1) /rand_bias0.3: 采样率 10M, 接收速率 60M, 均匀分布, 偏置电流 0.3A。

1. /norm_LS: 用 LS 算法, 求出各个幅度信号的 NMSE。之前的 LS 都是用一个信号来生成 LS 矩阵, 这次是用很多信号来生成 LS 矩阵。

2. /single_amp: 单一幅度数据作为训练数据, 且数据归一化。发送信号是均匀分布的随机信号, 采样率为 10M, 接收速率 60M, 偏置电流 0.3A。

2.1 /Threenonlinear1:

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 200 ,
DropPeriod = 12 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(size(xTrain{1},2)/miniBatchSize
origin rate = 1.000000e+07 , receive rate = 6.000000e+07
H order = 48
Hidden Units = 60
```

3. /mix_amp: 混合幅度数据作为训练数据, 且数据归一化。发送信号是均匀分布的随机信号, 采样率为 10M, 接收速率 60M, 偏置电流 0.3A。

3.1 /Threenonlinear1:

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 48
Hidden Units = 60
```

3.2 /Linear1: 一层线性网络

```
Linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 48
Hidden Units = 60
```

light_data_3.11/result/3.14:

(1) /rand_bias0.3: 采样率 10M, 接收速率 150M, 均匀分布, 偏置电流 0.3A。

1. /norm_LS: 用 LS 算法, 求出各个幅度信号的 NMSE。

```
LS
amp begin = 1 , amp end = 101 , amp step = 2
data_num = 100
origin rate = 1.000000e+07 , receive rate = 1.500000e+08
H order = 75
```

2. /single_amp: 单一幅度数据作为训练数据, 且数据归一化。发送信号是均匀分布的随机信号, 采样率为 10M, 接收速率 150M, 偏置电流 0.3A。

2.1 /Threenonlinear1: data num=25

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 200 ,
DropPeriod = 12 ,
DropFactor = 0.100000 ,
amp begin = 1 , amp end = 101 , amp step = 2
data_num = 25
validationFrequency is floor(size(xTrain{1},2)/miniBatchSize
origin rate = 1.000000e+07 , receive rate = 1.500000e+08
H order = 75
Hidden Units = 60
```

2.2 /Threenonlinear2: data num=50

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 200 ,
DropPeriod = 12 ,
DropFactor = 0.100000 ,
amp begin = 1 , amp end = 101 , amp step = 2
data_num = 50
validationFrequency is floor(size(xTrain{1},2)/miniBatchSize
origin rate = 1.000000e+07 , receive rate = 1.500000e+08
H order = 75
Hidden Units = 60
```

2.3 /Threenonlinear3: data num=100

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 200 ,
DropPeriod = 12 ,
DropFactor = 0.100000 ,
amp begin = 1 , amp end = 101 , amp step = 2
data_num = 100
validationFrequency is floor(size(xTrain{1},2)/miniBatchSize
origin rate = 1.000000e+07 , receive rate = 1.500000e+08
H order = 75
Hidden Units = 60
```