

第三周工作总结

张泽宇

2022 年 3 月 13 日

这一周的主要进行的工作包括有：

- 对上周仿真模型进行改进，添加了故障电阻，去掉了电路中不必要的部分，使电路更为简单。
- 补充了正常情况下的数据集，包括正常情况下负载不同时电流波形，不同故障电阻下负载的故障电流。
- 对神经网络进行了调整，主要体现在网络超参数的设定，进行了不同的尝试。同时编写了 command window 接口程序，用训练集和验证集之外生成的图片进行了测试。
- 参照资料学习并搭建了一个简单的三相电路，进行了初步学习。

以下为详细叙述

1 仿真模型改进

1.1 电路结构调整

我听取了教授关于故障电阻的建议，在全波整流电路中某一桥臂上添加了一个故障电阻，这个电阻为可变电阻，受到脉冲信号的控制，在第 0.06s 时由 0Ω 变为 200Ω 。

这我有一个疑问，就是关于故障电阻，我的理解是通过调节故障电阻的值，可以模拟线路发生不同情况下产生的多余阻抗，比如输电线绝缘层的破损程度不同，产生的阻抗也就不同，这就可以很好的用故障电阻表示。但是在像我这个模型中的电子电路里，故障电阻最多只能模拟短路和开路的情况，调节它的阻值是否有现实的意义？还是只是为了获得不同情型下的电流波形。

另外，在构造故障电阻模型的过程中，我使用的是可调电阻 Variable Resistor 加阶跃信号 Step 两个模块，这样虽然可以在 0.06s 时模拟出故障发生，但是不同的阻值需要手工设定，想要获得多一些的数据，比如 10 组 20 组，或者更多的地方需要修改数值，可能就得费点事情。有没有其他方便的方法或者模块可以在 simulink 中使用？

1.2 仿真时间的调整

我按照教授给出的建议，将仿真时间调整到 10ms，在第 6ms 的时候故障发生。在 50 赫兹的工频下，这样可以根据故障发生后两个周期的波形改变判断是否发生故障。

2 补充了正常情况下的数据集

通过调整负载电阻不同的阻值，得到负载电流随负载大小变化的一系列正常与故障样本。然后再在上周的数据拓展方面进行进一步的处理，得到了近 1200 张相关的图片。在数据集的制作方面，我总结得出的模式就是“matlab 仿真 +python 数据增强”。

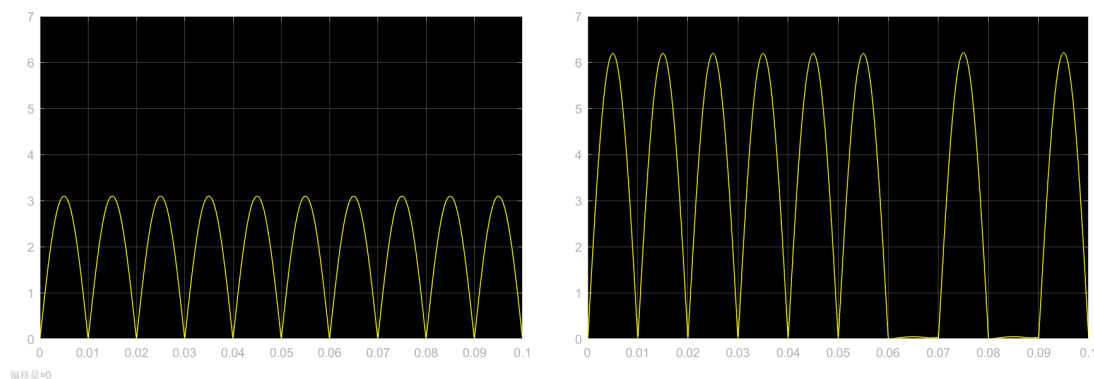


图 1: 部分正常与故障图像

但是，在这一方面也存在着和之前类似的问题，负载电阻无法实现自动的阻值调整，依旧需要手工修改设定。当需要调整的量较大时，比较费时费力。

3 对神经网络进行调整

3.1 调参

我查阅了一些资料，对神经网络的调参方法与调参技巧进行了一些学习。主要的调参过程和原则如下：

1. 恰好拟合时，训练集和测试集的 LOSS 都已经收敛，且很接近，模型能够很好地拟合训练样本和测试样本，泛化能力强。此时需要以优化网络设计为方向调整参数，使达到同样的效果下，网络结构更小、更轻量。

主要的方法可以有：

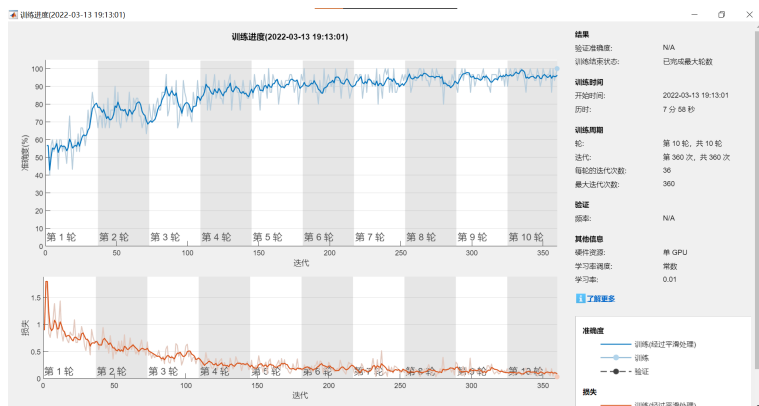
- 减少网络层数
 - 减少不同的层的参数，主要是卷积核的数量
 - 适当调大 batchsize(但这个尝试之后不太行，显示出计算机内存不够的提示)
2. 欠拟合时，从 LOSS 曲线上看，训练集和测试集从趋势上还没有收敛。可以做的调整是：

- 增加数据集，进行数据清洗
 - 加大训练迭代次数，有可能是网络还没训练完
 - 进一步衰减调小学习率
 - 添加更多的层，也有可能是网络容量不够
 - 去掉正则化约束的部分，L1、L2 正则（正则主要是为了防止过拟合）
 - 加入 BN 层加快收敛速度
 - 增加网络的非线性度（ReLU）
3. 过拟合：从 LOSS 曲线上看，都趋向收敛，但是测试集上的 LOSS 很高，甚至出现回升现象。说明泛化能力差，有可能训练集和测试集数据分布不一致，更加可能的是模型太复杂。可以调整的有：
- 增加样本数量，训练样本太少或者说小样本问题，很容易导致过拟合。
 - 数据增强
 - 早停法（Early stopping），从 LOSS 不在下降的地方拿到模型，作为训练好的模型
 - 降低网络的复杂度（深度）
 - L1 regularization, L2 regularization
 - Dropout 策略
 - 适当降低 Learning rate
 - 适当减少 epoch 的次数
4. 完全不收敛，LOSS 一直都很高。
- 先检查数据，输入数据、预处理、数据增强、标签等
 - 网络设计或者参数使用错误
 - LOSS 设计，优化函数

具体到这个模型中来看，经过许多次的调节后，确实达到了一个很好的水平，基本上满足了完全拟合的要求，准确率提升到了 95% 以上，甚至有几迭代过后能够达到 100%。当然，这也和需要解决的问题太过简单有关，正常和故障时的负载电流图像之间差别很大。

对比来看，当学习率调高一些，到 0.02 时，训练效果明显下降，出现了欠拟合，无法收敛的现象。

尽管如此，网络仍旧存在有训练时间过长的的问题，单次训练需要 7 至 8 分钟不等。网络在结构设计方面仍然需要进行改动。

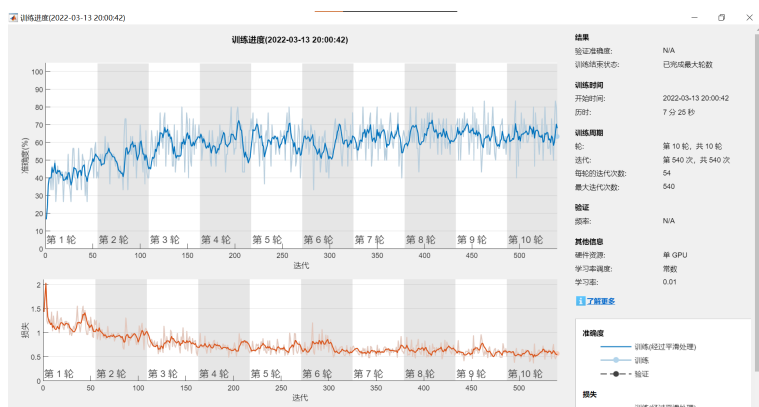


在单 GPU 上训练。
正在初始化输入数据归一化。

轮	迭代	经过的时间 (h h : m m : s s)	小批量准确度	小批量损失	基础学习率
1	1	0 0 : 0 0 : 0 5	5 6 . 6 7 %	0 . 8 9 4 8	0 . 0 1 0 0
2	5 0	0 0 : 0 1 : 1 1	7 0 . 0 0 %	0 . 5 8 9 6	0 . 0 1 0 0
3	1 0 0	0 0 : 0 2 : 1 8	7 3 . 3 3 %	0 . 6 0 6 5	0 . 0 1 0 0
5	1 5 0	0 0 : 0 3 : 2 7	8 6 . 6 7 %	0 . 2 2 2 2	0 . 0 1 0 0
6	2 0 0	0 0 : 0 4 : 3 3	9 3 . 3 3 %	0 . 3 0 3 6	0 . 0 1 0 0
7	2 5 0	0 0 : 0 5 : 3 8	9 3 . 3 3 %	0 . 0 9 9 4	0 . 0 1 0 0
9	3 0 0	0 0 : 0 6 : 4 0	9 6 . 6 7 %	0 . 0 8 8 6	0 . 0 1 0 0
1 0	3 5 0	0 0 : 0 7 : 4 4	1 0 0 . 0 0 %	0 . 0 8 4 5	0 . 0 1 0 0
1 0	3 6 0	0 0 : 0 7 : 5 8	1 0 0 . 0 0 %	0 . 0 3 2 5	0 . 0 1 0 0

训练结束：已完成最大轮数。

图 2: 最终的训练结果



在单 GPU 上训练。
正在初始化输入数据归一化。

轮	迭代	经过的时间 (h h : m m : s s)	小批量准确度	小批量损失	基础学习率
1	1	0 0 : 0 0 : 0 5	1 6 . 6 7 %	1 . 4 3 4 0	0 . 0 1 0 0
1	5 0	0 0 : 0 0 : 4 3	5 0 . 0 0 %	1 . 3 2 4 2	0 . 0 1 0 0
2	1 0 0	0 0 : 0 1 : 2 2	7 0 . 0 0 %	0 . 7 5 8 7	0 . 0 1 0 0
3	1 5 0	0 0 : 0 2 : 0 1	7 0 . 0 0 %	0 . 6 6 0 4	0 . 0 1 0 0
4	2 0 0	0 0 : 0 2 : 4 1	5 6 . 6 7 %	0 . 6 6 8 8	0 . 0 1 0 0
5	2 5 0	0 0 : 0 3 : 2 2	6 3 . 3 3 %	0 . 6 6 5 2	0 . 0 1 0 0
6	3 0 0	0 0 : 0 4 : 0 3	5 6 . 6 7 %	0 . 5 9 8 7	0 . 0 1 0 0
7	3 5 0	0 0 : 0 4 : 4 5	8 0 . 0 0 %	0 . 5 0 7 9	0 . 0 1 0 0
8	4 0 0	0 0 : 0 5 : 2 7	7 3 . 3 3 %	0 . 5 2 6 0	0 . 0 1 0 0
9	4 5 0	0 0 : 0 6 : 0 9	6 0 . 0 0 %	0 . 6 6 3 6	0 . 0 1 0 0
1 0	5 0 0	0 0 : 0 6 : 5 1	7 0 . 0 0 %	0 . 5 2 0 5	0 . 0 1 0 0
1 0	5 4 0	0 0 : 0 7 : 2 5	6 3 . 3 3 %	0 . 5 6 6 5	0 . 0 1 0 0

训练结束：已完成最大轮数。

图 3: 学习率调整到 0.02 后出现欠拟合

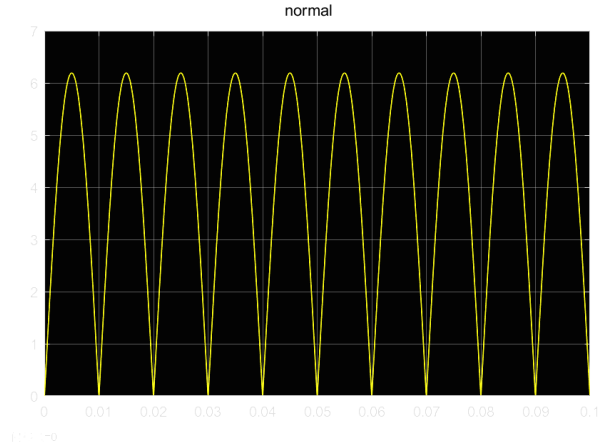


图 4: 预测结果

3.2 command window 接口

把训练之后的网络保存, 然后在 command window 中输入要检测的图片路径, 就可以直接判断出这个图片属于怎样的故障。经过实际测试, 预测结果较好。

程序代码:

```
1 clear;
2 load fault_detection_try;
3
4 file_name = '';
5 im = imread(file_name);
6 figure();
7 im = imresize(im,[594 840]);
8 imshow(im);
9 prediction = classify(net,im);
10 title(char(prediction));
```

4 简单的三相电路

我参照文献《MATLAB / Simulink 仿真在电力系统短路故障中的应用》¹, 简单学习搭建了一个无限大功率电力系统仿真模型。

并且利用 Three-Phase Fault 模块设置了相地之间、两相之间的短路故障, 基本上复现了文章里出现的仿真情况。

但是这一个系统的学习需要下周更为深入的了解。

¹王瀚, 周海峰, 刘天熙, 黄金满, 郑聪. MATLAB/Simulink 仿真在电力系统短路故障中的应用 [J]. 应用能源技术, 2020(07):1-8.

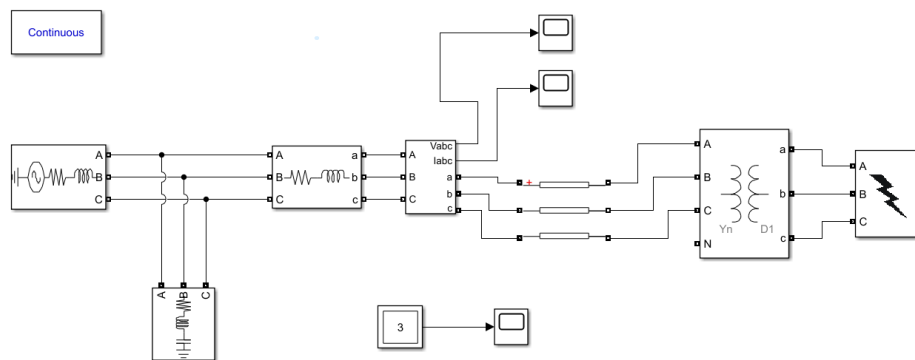


图 5: 无限大功率电力系统仿真

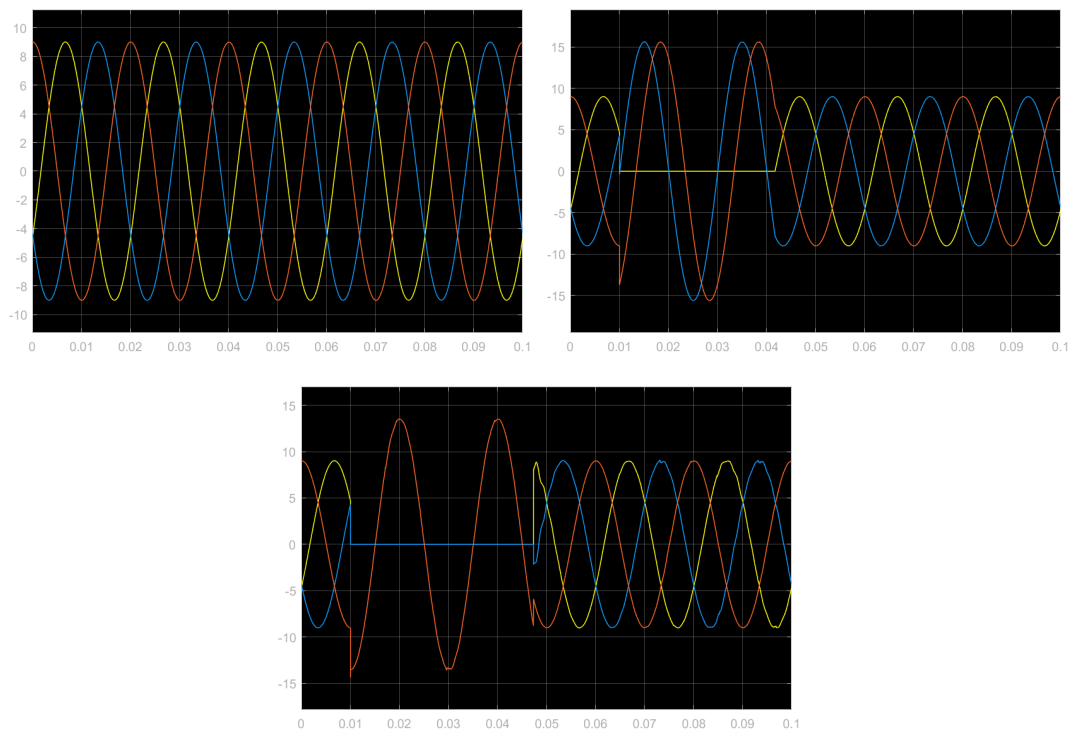


图 6: 正常情况、A 相对地短路、AB 相间短路

5 下周计划

1. 对第二周开始的全波整流电路故障检测神经网络进行最后的整理和完善。
2. 就无限大功率电力系统原理进行学习。
3. 寻找参考一些已有的故障检测神经网络程序，进行学习。