

已知的BUG记录日志

现在已知定点数仿真工具存在一些bug而且尚未或者无法修复,其中包括代码编写时候还没有完善的部分,也有MATLAB自身的一些隐藏问题,在现阶段编写仿真代码的时候需要注意避开这些容易引发错误的地方.

一、会明显影响仿真精度的BUG

1. 定点数溢出上下界的问题

- 设置仿真时,如果发现GPU和单核仿真精度差异过大,有可能是在GPU上执行的函数设置的溢出上下界存在问题,现有的GPU版本定点数仿真为了兼容MATLAB的定点数对象fi,中间需要做一些从MATLAB定点到整数的转化再传递到GPU,所以可能有时候会遇到溢出设置不对应而导致的精度问题.
- 需要修改的文件是 `**+FAST/+kernel/**` 文件夹下面的两个".m"文件中的up_bound和low_bound参数:
 - FXPGEMMonGPU
 - DepthwiseGEMMGPU
- up_bound和low_bound都修改成64位有符号整数的上下界时一般GPU的精度会恢复,但是有时候为了仿真特定位数的定点数精度,还需要在进行修改.

2. MATLAB在多核间传递定点数对象时候的错误

- 如果使用spmd语句或者parfor语句,请不要在spmd语句和parfor语句中使用含有以numerictype和fimath对象为参数的函数,例如

```
myfunc(input_param1, input_param2, t, f), 其中t为
numerictype类型,f为fimath类型.当在MATLAB的parfor和spmd语句
中使用myfunc的时候,如果从外部传入参数,再调用myfunc,可能会出现错
误 例如:
```

```
f = fimath('CastBeforeSum',0, 'OverflowMode',
'Saturate', 'RoundMode', 'floor', ... 'ProductMode',
'SpecifyPrecision', 'SumMode', 'SpecifyPrecision',
'ProductWordLength',wordlen,
'ProductFractionLength',fraclen, 'SumWordLength',
wordlen, 'SumFractionLength', fraclen);
```

```
t = numerictype('WordLength', wordlen,
'FractionLength',fraclen);
```

```
    spmd  
  
    myfunc(input_param1, input_param2, t, f)  
  
end
```

以上写法可能会出错,具体表现为外部设置的参数在传入函数内部后不一致.这个错误可能和MATLAB的对象模型以及并行语句的对象拷贝方式有关,根本原因还没有查明.

设计带有类似需求的函数的时候应该尽量在函数内部直接对`numericType`和`fimath`类型赋值,而不要传参.

二、不会对精度造成影响但会偶发的错误

1. ActiveSession函数有可能引发错误

- 这个函数主要是用于对Layer类和Net类下面定义的一些函数接口进行声明并返回一个Layer对象,实际操作就是扫描Layer类和Net类文件夹下面的所有函数文件,并把接口函数的名称和各自的参数表、返回值写入到对应的类公有函数定义文件中,以此实现在每次执行的时候调用的类接口都能保持是最新的刚修改过的状态.避免反复修改接口定义和实现文件.
- 但是这个函数偶尔会导致一些错误,函数内部已经设置了出错自动尝试5次,但如果还是遇到错误,可以重复执行直到能正常返回一个Layer对象为止.

2. 使用spmd语句时,再使用labBarrier语句进行核间任务同步有可能出错

错误一般出现在最后一次循环打印的时候,现在还不知道具体的错误原因,但是最好在关键地方使用labBarrier并且如果需要进行长时间仿真,为避免丢失结果,最好用写入日志文件的方式把仿真过程的精度变化记录到文件.