### 第二十六次2020年6月15日-6月21日

#### 1 实验方面

**我做了什么：**

本周我对实验代码重新进行了规整，把exact cover数据集批量处理成CNF公式作为SAT求解器的输入。原有的编译器版本是可以对exactly-one约束进行识别和处理，也可以处理非exactly-one约束。但由于exact cover数据集编码得到的CNF公式中的长子句都是exactly-one约束，因此我对之前版本的代码进行了进一步简化，减少了一些不必要的步骤。在实验过程中，某些测试用例运行时出现内存溢出，我花费了很多时间去给虚拟机更多内存，并且试图提高求解器的空间效率。

**取得的进展：**

在规整代码过程中，我发现了之前代码的漏洞，修复了实验结果有时候与正确结果存在倍数差异的问题。目前测试得到的模型数实验结果都是正确的。

在bell数据集上运行得到的实验结果比D4更好，这是之前不常出现的。我猜测是因为bell数据集得到的exactly-one约束都是长度很长的子句，因此单独处理的优势更明显。

**得到的实验结果：**

使用exact cover数据集bell进行测试，bell-1到bell-9可以正常运行得到结果。bell-10到bell-12包含子句数目十万以上，运行时显示内存溢出。我尝试了各种方法给虚拟机分配更多空间（目前16G），但最终还是没能得到结果，太大的数据集在我目前的虚拟机上运行不出来。

**接下来计划：**

由于bell数据集中的测试用例规模跨度很大，两个数据集之间的规模跨度甚至达到一个数量级，因此缺少中等规模数据集的实验结果。如下图所示，实验效果并不太明显，我觉得目前得到的实验结果很不完整，有待继续实验。接下来我将寻找更合适规模的测试用例，以期待得到更好的实验结果。

图片包含 游戏机, 文字

描述已自动生成