### 第二十八次2020年6月29日-7月5日

#### 1 exact cover

Exact cover：P = ⟨X, S⟩ ，P是一个二元组。X是有限个元素集合，S是由X的若干子集组成的集合。P的一个解 S\*是S的子集，且满足X中每个元素在S\*中恰好出现一次。

满足以下条件的集合为一个精确覆盖：

S\*中任意两个集合没有交集，即X中的元素在S\*中出现最多一次

S\*中集合的全集为X，即X中的元素在S\*中出现最少一次

Exact Cover 问题分三类：决策、计数、枚举。分别对应有没有解、有多少个解、枚举每个解。

DLX算法由Knuth提出，是一种基于回溯的深度优先搜索技术，该技术基于X算法并使用一种叫做舞蹈链的数据结构实现 。舞蹈链是由双向链表演变的交叉十字双向循环链。

删除p ;恢复p ;

手机屏幕截图

描述已自动生成

#### 2. 论文

Dancing with Decision Diagrams: A Combined Approach to Exact Cover

本文通过使用零抑制二进制决策图（ZDD）来表示一组精确覆盖，在DLX上有效地使用了备忘录缓存。 ZDD是Binary Decision Diagram（BDD）的一种变体，表示一组有向无环图（DAG）。我们修改DLX以输出代表精确覆盖集的ZDD。由于我们可以使ZDD包含子问题的解决方案，即子ZDD（子图），因此通过存储代表解决方案集的子ZDD根节点的地址，备忘录缓存仅需要为每个问题提供恒定的内存。此外，如果子问题的解决方案表示为ZDD，则可以在固定时间内完成。因此，将ZDD与DLX结合使用可以有效地使用备忘录缓存，并可以加快搜索速度。就我们所知，这是改进DLX的第一项工作。我们针对覆盖问题的各种基准实例进行了实验，并确认我们的建议比DLX快几个数量级。除了加速DLX，我们的建议还具有输出ZDD的优点。 ZDD支持有用的操作，这些操作可以按与ZDD大小成比例的时间运行。例如，ZDD支持模型计数，子集枚举，找到使目标函数最大化的最佳子集，以及在集合族上进行有效的二元运算，称为族代数。这些操作使操作一组精确的盖板变得容易。例如，使用族代数可以轻松找到满足其他约束的精确覆盖。

我们通过扩展DLX提出了一种新算法DXZ。 我们的创新在于将DLX与零抑制的二进制决策图相结合。 DXZ可以充分利用备忘录缓存，因此可以显着提高速度。 该算法输出表示所有精确覆盖集的有效ZDD，因此可以使用ZDD操作轻松地操作一组精确覆盖。