**在有界度图中独立集的改进逼近**

Magntús M. Halldórsson, Jaikumar Radhakrishnan

**摘要：**在有界度图中寻找最大独立集是一个被充分研究的NP完全问题。我们研究了两种求近似解的方法，得到了集中改进的性能比。

第一种是在我们之前的论文中介绍的子图移除方案。利用更好的组件算法（component algirithms），我们得到了一个具有Δ/6(1+o(1))性能比的有效方法。然后我们给出了Ajtai等人的定理用于求无团图的独立集个数的一个实现，并利用这一实现在我们的方案中得到了 O(Δ/log log Δ)的性能比。这是第一个 o(Δ)的比率。

第二种是Berman和Fürer的局部搜索方法，他们证明了一个很好的性能比率，但需要极多的时间。我们展示了在最大度为Δ的图上，如何大大减少计算需求，同时保持相同的性能比大致在(Δ+3)/5。然后我们展示了他们算法的缩小版本，有着(Δ+3)/4的性能比，在以前合理有效的方法的界限实现了改进。

**1 前言**

图的独立集是一个顶点的集合，它的任意两个顶点都不相邻。在图论和组合优化中，寻找独立集的最大顶点数是一个非常重要的问题。

假设这个问题是NP困难的，那么最有希望的方法就是开发能够找到高质量近似解的启发式算法。算法的性能比的定义是最优解的大小与算法的大小之间的最坏比率。尽管付出了相当大的努力，还是没有一种针对独立集问题的算法的性能比小于O(nlog2n)[6]，其中n是输入图中的顶点数。近年来关于交互式证明系统的研究结果，在著名的Arora et al的论文[3]中达到了顶峰，表明不能期望有常数近似算法，事实上，n1/4的比率是无法到达的[4]。

考虑到一般情况下这个问题的明显难度，很自然地会问，什么限制能使这个问题更容易近似。也许最自然和最频繁发生的情况是最大顶点度以一个常数为界。正如考虑成对元素之间的冲突会联想到独立集(或团)问题，当问题的关键参数固定时，自然会想到有界度的变体。

对有界度版本（B-IS），求精确解仍是NP完全问题，但求近似解却变得相当容易。事实上，任何一个在最大度为Δ的图上寻找最大独立集的算法都有和Δ相关的性能比。这个问题也属于原始的最大SNP完全问题[11]，所以[3]的结果暗示存在一个常量c>1，c近似是NP苦难的，自然意味着没有多项式时间近似模式存在（除非P=NP）。这自然导致我们会寻找B-IS可以近似的最佳常数。

我们通过研究两种最近的方法来处理这个问题: 前者是一种从图中去除小团的算法模式。这个想法起源于[6]，其可以追溯到Erdös[7]——来自于没有小团的图比一般图包含可证明的更大的独立集的观察。此外，这些更大的解可以被有效地找到。对于几乎没有不相交团的图，我们可以手动删除所有的团，并在其他部分找到承诺的改进解。另一方面，具有多个不相交团的图不能包含一个非常大的独立集，从而给出了最优解的上界。因此，无论在哪种情况下，我们的性能比率都将得到提高。

我们先前使用这个模式来改进最小度贪婪算法的逼近，从(Δ+2)/3提升到 Δ/3.81(1+o (1))[8]。这次，通过对Khanna等人的一个简单局部搜索算法[10]进行新的研究，我们得到了一个令人惊讶的强比率，Δ/6(1+o (1))。

我们也用这个模式来回答一个诱人的问题：迄今为止所有的B-IS近似结果仅仅有限改善Δ前面的系统，o(Δ)的性能比有没有可能实现？答案是肯定的，我们提出了一个有O(Δ/log logΔ)性能比的算法。作为关键的一步，我们给出了Ajtai, Erdös, Kolmlós和Szemerédi[1]关于不含小团的系数图的独立集数量的存在定理的确定性实现。

我们考虑的后一种方法是基于Berman 和 Fürer[5]的，它可以被描述为一种局部搜索算法，其在当前解的补充中进行额外的搜索。对于任意固定常数h，他们的算法在低最大度图有出色的性能比。当Δ是偶数时为(Δ+3)/5+1/h；当Δ是奇数时为(Δ+3.25)/5+1/h。

不幸的是，这个方法花费非常多的时间。局部搜索邻域是搜索一个改进的解集合，它包括所有与当前解的差(或距离)为具有σ个顶点的连通图的解。为了得到它们的结果，我们必须搜索一个大小为σ=的邻域，这意味着搜索复杂度为 。特别地，复杂性对近似h的大小的依赖是双指数的。例如，为了在Δ=4时获得1.6的比率，这种方法需要的时间复杂度。

我们通过收紧分析来解决这个可行性问题，从而大大减少了对邻域大小的要求。特别是，我们得到了相同的性能比的同时，缩小邻域大小为，并消除了对h的指数依赖。虽然仍然不完全实用，但这接近这种方法的极限。

我们还观察到，小的邻域产生了惊人的良好性能比。在只有2Δ大小的邻域，可以很容易地在O(n2)时间内实现并得到(Δ+3)/4的比率。这是以前实用的有界度独立集算法的最佳比率(Δ+ 2)/3的改进[8]。

**3 局部搜索方法**

**3. 1 算法**

本算法基于Berman 和 Fürer[5]的算法，是一种局部搜索方法——但带有变形。局部性，或者说邻域，其自然定义如下: 一个解的集合，它在很少的顶点上与当前解不同(就对称集合差而言)。从任意一个解开始(例如一个极大独立集)，我们搜索所有严格大于它的相邻的解。这个过程会一直持续下去，直到找不到任何改进为止。

我们的变形是这个方法的一个递归应用，其可以作为这个解的补充。一旦找到一个不可改进的解 A，就把它应用到 COMP (A)中，其被定义为由A中至少有两个相邻节点的子图的诱导子图(当Δ=3时，一个相邻节点就足够了)。由于这个子图的最大度小于等于2，递归会结束于最优解的二度图。考虑到这个新的更简单的解，如果它比原来的解更大，我们就令其成为我们当前的解，并继续尝试改进它; 否则，我们退出并宣布在这个搜索过程中的最简单的解决方案。

1.1.1 标题三级标题五号宋体

正文

**参考文献**

参考文献小五号，只列举最主要的，必须是公开发表的书刊才能列入，最少不得少于5条。文献按文章中出现先后顺序排列

**(各类文献严格按照主页上的《参考文献规范》)**

1. 专著：［序号］ 作者.题名[M].出版地：出版者，出版年：起止页码.
2. 期刊：［序号］ 作者(多作者用逗号分开，超过3个者用“，等”代替).文章题目[J].刊物名称，年，卷(期)：起止页码.
3. 会议论文集：［序号］ 作者.题名[C]//编者.论文集名.出版地：出版者，出版年：起止页码.
4. 英文会议：［序号］ 作者.题名[C]//Proceedings of 会议名称.出版地：出版者，出版年：起止页码
5. 学位论文：［序号］ 作者.题名[D].保存单位XX学位论文，年份.
6. 报告：［序号］作者.题名[R].保存地点：保存单位，年份.
7. 报纸文章：［序号］ 作者.题名[N].报纸名，出版日期(版次).
8. 标准：［序号］标准编号，标准名称[S].出版地：出版者，出版年.
9. 专利: ［序号］专利所有者.专利题名[P].专利国别:专利号,公开日期.
10. 电子文献:主要责任者.电子文献题名[电子文献标识/载体类型]. [发表或更新日期].电子文献的出处或可获得地址.
11. 电子文献标识:[DB]-数据库 [CP]-计算机程序 [EB]-电子公告
12. 电子文献载体类型:[OL]-联机网络 [MT]--磁带 [DK]-磁盘 [CD]-光盘