Independent sets with domination constraints

Magnús M. Halldórsson[1,3], Jan Kratochvíl[2,⋆], and Jan Arne Telle[3]

[1] University of Iceland, Reykjavik, Iceland. mmh@hi.is
[2] Charles University, Prague, Czech Republic. honza@kam.ms.mff.cuni.cz
[3] University of Bergen, Bergen, Norway. telle@ii.uib.no

Abstract. A ρ-independent set S in a graph is parameterized by a set ρ of non-negative integers that constrains how the independent set S can dominate the remaining vertices ($\forall v \notin S : |N(v) \cap S| \in \rho$.) For all values of ρ, we classify as either $\mathcal{NP}$-complete or polynomial-time solvable the problems of deciding if a given graph has a ρ-independent set. We complement this with approximation algorithms and inapproximability results, for all the corresponding optimization problems.
These approximation results extend also to several related independence problems. In particular, we obtain a $\sqrt{m}$ approximation of the Set Packing problem, where m is the number of base elements, as well as a $\sqrt{n}$ approximation of the maximum independent set in power graphs G^t, for t even.

1 Introduction

A large class of well-studied domination and independence properties in graphs can be characterized by two sets of nonnegative integers σ and ρ. A (σ, ρ)-set S in a graph has the property that the number of neighbors every vertex $u \in S$ (or $u \notin S$) has in S, is an element of σ (of ρ, respectively) [9]. This characterization facilitates the common algorithmic treatment of problems defined over sets with such properties. Previous papers on classification of the complexity of problems from an infinite class include [5, 8]. Unfortunately, the investigations of uniform complexity classification for subclasses of (σ, ρ)-problems have so far been incomplete [7, 10]. In this paper we give a complete complexity classification of the cases where $\sigma = \{0\}$, which constitute maybe the most important subclass of (σ, ρ)-problems.

In this class of problems the chosen vertices are pairwise non-adjacent, forming an independent set. Independent (stable) sets in graphs are a fundamental topic with applications wherever we seek a set of mutually compatible elements. It is therefore natural to study the solvability of finding independent sets with particular properties, as in this case, where the independent set is constrained in its domination properties.

Assume that we have an oracle for deciding membership in $\rho \subset \mathbb{N} = \{0, 1, ...\}$. Let $N(v)$ denote the set of neighbors of a vertex v. Consider the following decision problem:

⋆ Research support in part by Czech research grants GAUK 194 and GAČR 0194/1996.

ρ-IS Problem
Given: A graph G
Question: Does G have an independent set of vertices $S \neq \emptyset$ with
$|S| \geq \min\{k : k \notin \rho\}$ such that $\forall v \notin S : |N(v) \cap S| \in \rho$?

When ρ is the set of all positive integers the ρ-IS problem is asking for an independent dominating set, a problem which is easy since any maximal independent set is also a dominating set. When $\rho = \{1\}$ the ρ-IS problem is asking for the existence of a perfect code, a problem which is $\mathcal{NP}$-complete even for planar 3-regular graphs [6] and for chordal graphs [7]. The natural question becomes: For what values of ρ is the ρ-IS problem solvable in polynomial time? In the next section we resolve this question for all cases, up to $\mathcal{P}$ vs. $\mathcal{NP}$.

Theorem 1 *The ρ-IS problem is $\mathcal{NP}$-complete if there is a positive integer $k \notin \rho$ with $k + 1 \in \rho$, and otherwise it is solvable in polynomial time.*

Approximation algorithms Even for the cases when the decision problem is solvable in polynomial time, the corresponding optimization problem, finding a minimum or maximum size ρ-IS, is hard. In Section 3 we give on the one hand approximation algorithms for these optimization problems, and on the other hand strong inapproximability results.

The class of problems that we can approximate is that of finding an independent set where vertices outside the set are adjacent to at most a given number k vertices inside. We obtain performance ratios of $O(\sqrt{n})$ for the maximization versions of these problems. This is significantly better than what is known for the ordinary Independent Set problem, where the best performance ratio known is $O(n/\log^2 n)$ [1], a mere $\log^2 n$ factor from trivial. In fact, it is known that obtaining a performance ratio that is any fixed root of n factor better than trivial is highly unlikely [4].

We find that the same algorithmic technique extends to a number of related independence problems, for which no non-trivial bounds had been given before. Given a base set with m elements and a collection of n subsets of the base set, the Set Packing problem is to find the largest number of disjoint sets from the collection. There is a standard reduction from Independent Set to Set Packing [2] where the number of sets n equals the number of vertices of the graph and the number of base elements m equals the number of edges of the graph. Thus, the hardness results of [4] translates to a $n^{1-\epsilon}$ lower bound for Set Packing, as a function of n, but only a $m^{1/2-\epsilon}$ lower bound in terms of m. The only previous upper bound in terms of m (to our best of knowledge) was the trivial bound m. This left a considerable gap in our understanding of the approximability of the problem, e.g. when m is linear in n.

We resolve this issue by showing that a simple and practical greedy algorithm yields a performance ratio of $\sqrt{m}$. It also yields an $O(\sqrt{m})$ performance ratio for the Maximum k-Matching of a set system (see definition in Section 3), and a $\sqrt{n}$ ratio for the maximum collection of vertices of a graph of mutual distance at least t, for odd t. In all of these cases, the bounds are essentially best possible.

2 Decision Problems

In this section we prove Theorem 1. The polynomial cases are summarized in the following result:

Lemma 2. *The ρ-IS problem is solvable in polynomial time if $\rho = \emptyset$, $\rho = \mathbb{N}^+$ or $\rho = \{0, 1, ..., k\}$ for some $k \in \mathbb{N}$.*

Proof. The cases $\rho = \emptyset$ and $\rho = \mathbb{N}^+$ are trivial. When $\rho = \{0, 1, ..., k\}$ for some $k \in \mathbb{N}$, we are asking if the input graph G has an independent set S of at least $k+1$ vertices such that every vertex not in S has at most k neighbors in S. The algorithm simply tries all subsets S of size $k+1$, and if none of them satisfy the conditions the answer is negative. ∎

We remark that when restricted to chordal graphs the ρ-IS problem is solvable in polynomial time whenever $\min\{k : k \in \rho\} \geq 2$ [7]. We turn to the $\mathcal{NP}$-complete cases, and first state two earlier results. When $\rho = \{1\}$, a ρ-IS set is also known as a perfect code.

Theorem 3 *[6] Deciding if a 3-regular graph has a perfect code is $\mathcal{NP}$-complete.*

Theorem 4 *[10] The ρ-IS problem is $\mathcal{NP}$-complete whenever ρ is a finite non-empty subset of positive integers or when $\rho = \{k, k+1, ...\}$ for some $k \geq 2$.*

Our first result, whose proof is omitted from this extended abstract, is an $\mathcal{NP}$-completeness reduction from the above problem.

Lemma 5. *The $\{0, k+1, k+2, \ldots\}$-IS problem is $\mathcal{NP}$-complete for $k \geq 1$.*

Let EVEN be the set of all even and ODD be the set of all odd non-negative integers. As is often the case with parity problems, e.g. Chromatic Index of 3-regular graphs, the cases of EVEN-IS and ODD-IS require a special reduction for their $\mathcal{NP}$-completeness. These reductions, from the problem EXACT 3-COVER [2], are again left out of this extended abstract.

Lemma 6. *The EVEN-IS and ODD-IS problems are $\mathcal{NP}$-complete.*

We now prove the remaining cases, completing the proof of Theorem 1

Lemma 7. *The ρ-IS problem is $\mathcal{NP}$-complete if there is a positive integer $k \notin \rho$ with $k+1 \in \rho$.*

Proof. Let $t = \min\{x : (x \geq 1) \wedge (x \in \rho) \wedge (x+1 \notin \rho)\}$. If such t does not exist then either $\rho = \{k+1, k+2, ...\}$ and ρ-IS problem is $\mathcal{NP}$-complete by Theorem 4, or $\rho = \{0, k+1, k+2, ...\}$ and is $\mathcal{NP}$-complete by Lemma 5. Let $z = \min\{x : (x > t) \wedge (x \notin \rho) \wedge (x+1 \in \rho)\}$. If such z does not exist then $\rho = \{1, 2, ..., k\}$ and is $\mathcal{NP}$-complete by Theorem 4.

For any 3-regular graph G we construct a graph G' which has a ρ-IS if and only if G has a perfect code. We shall be assuming that G is sufficiently large, e.g. contain at least z^2 vertices.

Let $V(G) = \{v_1, ..., v_n\}$. The derived graph G' will consist of $z+1$ copies $G^1, ..., G^{z+1}$ of G, with vertices $V(G^k) = \{v_1^k, ..., v_n^k\}$, along with a large collection of nodes connected into a clique. For each edge $v_iv_j \in E(G)$ add edges $v_i^k v_j^{k'}$ for $1 \leq k, k' \leq z+1$. This ensures that for any independent set S in G', its projection S_G ($u_i \in S_G$ iff $\exists k : u_i^k \in S$) onto G is also an independent set.

A *claw* is a set of four vertices, consisting of a *center* vertex v_i^k and its three neighbors in a particular copy of G. Thus, G' contains $n(z+1)$ claws. Note that an independent set contains at most three vertices of a claw, and if the center vertex is in the independent set then the other three are not. Our construction will ensure that for any ρ-IS S of G', each claw contains exactly one vertex of S. This will imply that for each $v_i \in V(G)$, either all copies of v_i or no copies of v_i are in S, as all copies have the same neighbors. Moreover, it will imply that the projection S_G of S onto the 3-regular graph G is a perfect code, since a subset of vertices containing exactly one vertex from the closed neighborhood of each vertex is a perfect code. Henceforth, when we refer to claws, we always mean claws as described above.

There is a clique node for every group of $z+1$ vertex-disjoint claws in G' and also one clique node for every group of t vertex-disjoint claws in G'. These clique nodes are connected to all the vertices of those claws in G', and to no other vertex in the copies of G. Note that both $t \in \rho$ and $z+1 \in \rho$, but $\{t+1, ..., z\} \cap \rho = \emptyset$ and $t+1 \leq z$.

It remains to show that for any ρ-IS S of G', each claw contains exactly one vertex of S. To ease the presentation, we first prove a weaker property, and then complete the specification of G' by adding some more vertices to the clique, which will allow us to prove the main property.

Claim. Any ρ-IS S in G' contains either one or three vertices from each claw.

Proof. Recall that by definition S must contain at least $t+1$ nodes, and at most one of these could be a clique node. But if S contains a clique node y, we could arbitrarily pick t other vertices of S, and some clique node $x \neq y$ would be adjacent to exactly t vertex-disjoint claws having these t vertices from S as centers. We ensure that the claws are vertex-disjoint by choosing the neighbors of the centers from separate copies of G. The clique node x would have a total of $t+1$ neighbors in S, but $t+1 \notin \rho$. Thus, S contains no clique node.

Moreover, if $t+1 \leq |S| \leq z$, then we can find $|S|$ vertex-disjoint claws with vertices of S as centers, chosen as above, and some clique node will be adjacent to these $|S|$ vertices, but $\{t+1, ..., z\} \cap \rho = \emptyset$. Thus $|S| \geq z+1$.

If some claw X has $X \cap S = \emptyset$, we can take z vertices from S, cover them by z vertex-disjoint claws centered at these vertices, as above, and a clique node x will be adjacent to these claws and to X. But then x would have z neighbors in S, and $z \notin \rho$. Thus X has at least one vertex in S.

Moreover, X cannot have two vertices in S, since we can pick $t-1$ vertices from S and cover them, as above, by $t-1$ vertex-disjoint claws that do not intersect the neighborhood of X. A clique node x is adjacent to these claws and to X and it would have $t+1$ neighbors in S if X had two vertices in S. However, $t+1 \notin \rho$. ∎

Claim 2 already establishes that either all or none of the copies of a vertex $v_i \in V(G)$ must be in a ρ-IS S, since any pair v_i^k and $v_i^{k'}$ are centers of distinct claws sharing the three other claw vertices. When $v_i^k \in S$ the three other claw vertices are not in S so that $v_i^{k'} \in S$ also, and vice-versa.

We complete the construction of G' in three different manners depending on which of the following three cases holds:

- (i) 0 and 1 are in ρ, but 2 is not.
- (ii) For some $w \geq 3$, $w-2$ is in ρ, but w is not.
- (iii) For some $w \geq 2$, w is in ρ but $w-2$ is not.

If none of these cases hold, then for each $w \in \mathbb{N}$ either both or none of w and $w+2$ would have to be in ρ, and $\rho \subset \mathbb{N}$ would be equal to EVEN or ODD. Note that if any pair of non-center vertices of a claw are adjacent, then by Claim 2 we already know the claw has exactly one vertex in any ρ-IS set.

In case (i) we add a node to the clique for each pair of vertices in G^1 which are copies of vertices at distance 2 in G, and make the node adjacent to the pair.

In case (ii) we add a node to the clique for each group of $w-2$ vertex disjoint claws, and make the node adjacent to these claws.

In case (iii) we add for each set Y of $w-2$ vertex-disjoint claws a new clique node Y_i for each $v_i \in G$ whose neighbors form an independent set. We make Y_i adjacent to v_i^1 and to v_i^2 and to all copies of vertices in G^1 at distance two from v_i. There are between three and six such vertices in G^1, since if only two then this would be the whole graph G. Let v_i have neighbors v_a, v_b, v_c and let these latter three have additional neighbors a' and a'', b' and b'', c' and c'', respectively. We make Y_i adjacent to the copy in G^2 of some of these vertices, depending on the common identities of this multiset of six vertices (see Figure 1):

- A: six singletons- adjacent to no further vertices,
- B: one triple and three singletons- adjacent to no further vertices,
- C: one pair and four singletons- adjacent to copy in G^2 of the pair,
- D: two pairs and two singletons- adjacent to copies in G^2 of both pairs,
- E: three pairs- adjacent to copies in G^2 of all three pairs,
- F: one triple, pair and singleton- adjacent to copy in G^2 of the pair.

Claim. Any ρ-IS S in G' contains exactly one vertex from each claw.

Proof. Let X be any claw in G^1. We show that in none of the cases (i),(ii) or (iii) does X contain three vertices in S. The claim then follows for all claws in G', since either all or no copies of a vertex must be in S.

In case (i) we have $2 \notin \rho$. No two vertices in G^1 at distance two in G can both be in S since then the corresponding newly added clique node would have exactly two neighbors in S. Hence no claw in G^1 can contain more than one vertex in S.

In case (ii) we can find a set of $w-3$ vertex-disjoint claws in G whose centers are all in S. We ensure that such vertex-disjoint claws can always be found by assuming, without loss of generality, that G is large, say with at least w^2 vertices, so that by Claim 2 the center vertices can be chosen to be copies of vertices in

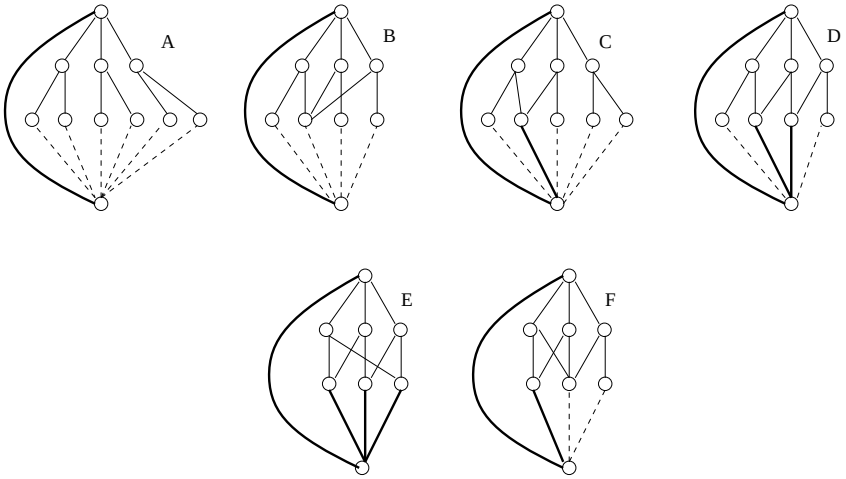

Fig. 1. The six cases, showing the center vertex of the claw on top, the clique vertex on bottom, with a thick edge indicating that the clique vertex is adjacent to copies in both G^1 and G^2 and a dotted edge indicating adjacency only to the copy in G^1. In each case, if the top claw has three vertices in S the clique vertex has no S-neighbors in the figure, whereas if each claw has exactly one vertex in S the clique vertex has exactly two S-neighbors in the figure (counting thick edges twice).

G whose pairwise distance in G is at least three. If X had three neighbors in S, the clique node adjacent to X and these claws would have exactly w neighbors in S. However, $w \notin \rho$.

In case (iii) a set Y of $w-2$ vertex-disjoint claws has the central vertex chosen. Let X have center node v_i^1. The clique node Y_i added for v_i^1 and these $w-2$ claws has at least $w-2$ neighbors in S. If the claw X has three vertices in S then these are all the three neighbors of v_i^1 and none of the remaining neighbors of Y_i is in S. On the other hand, if X and all other claws all have one vertex in S, then it is easy to check, in each of the separate cases of common identities above, that exactly two of the remaining neighbors of Y_i is in S. For example, if Y_i has an extra neighbor v_j^2 in G^2 then in each case v_j^2 is adjacent to exactly two (a pair) of the neighbors of v_i^1 and the third neighbor of v_i^1 must be in S whenever $v_j^2 \in S$ so that the remaining neighbors of Y_i could then not be in S. We conclude that, since $w-2 \notin \rho$ but $w \in \rho$, the claw X must have exactly one vertex in S. ∎

A perfect code in G gives rise to a ρ-IS in G' consisting of all copies of nodes in the perfect code. For every ρ-IS S in G', either all or no copies of a vertex from G must be in S and no clique node is in S. Hence it follows from Claim 2 that the projection of S onto G is a perfect code. ∎

3 Optimization

Let us consider the complexity of ρ-IS optimization problems. Clearly optimization is no easier than the corresponding decision problem, thus we are interested in the problems where the decision version is polynomial solvable. When an optimization problem turns out to be hard to compute, we would further like to know how hard it is to compute approximate solutions by polynomial-time algorithms.

We say that an algorithm *approximates* a problem *within* r if the solution computed on any instance never strays from the optimal by more than a multiplicative factor r. The algorithm then has *performance ratio* r. Note that the factor r may be a function of the size of the input. When a better approximation algorithm cannot be found, we naturally try to show that no better algorithm can be found given some natural complexity-theoretic assumption.

Approximation is not well defined when the corresponding decision problem is not polynomial solvable. If an algorithm cannot produce a feasible value for a solvable problem, the approximation ratio for that problem is not defined. Attempts to deal with this by modifying the definition of a performance ratio seldom meet with success. Thus, we consider only the approximation of the ρ-IS optimization problems, either minimization or maximization, whose decision version is in P, namely: $\rho = \mathbb{N}^+$, $\rho = \{0\}$, and $\rho = \{0, 1, \ldots, k\}$, for some $k \in \mathbb{N}^+$.

Minimization problems are trivial when ρ contains zero, which leaves only the case $\rho = \mathbb{N}^+$. This is the *Minimum Independent Dominating Set* problem, which is known to be $\mathcal{NP}$-hard to approximate within $n^{1-\epsilon}$, for any $\epsilon > 0$ [3]. The reduction holds even if the graph is sparse, thus it is hard within $m^{1-\epsilon}$. In fact, no sub-linear performance ratio is known for this problem.

The maximization problem with $\rho = \{0\}$ is trivial, whose solution consists of all isolated vertices. When $\rho = \mathbb{N}^+$ we have the *Maximum Independent Set* problem, for which the best performance ratio known is $O(n/\log^2 n)$ [1]. Håstad has recently improved a sequence of deep results to show that this problem is hard to approximate within $n^{1-\epsilon}$, for any $\epsilon > 0$ [4]. This result is modulo the assumption that $\mathcal{NP} \neq \mathcal{ZPP}$, namely that zero-error randomized polynomial algorithms do not exist for all problems in $\mathcal{NP}$. This is highly expected, while slightly weaker hardness results are known under the stronger assumption that $\mathcal{P} \neq \mathcal{NP}$. We shall use this result in this paper, with the knowledge that weaker assumptions will then also transfer to our results. In particular, our reductions do give the $\mathcal{NP}$-hardness of the exact optimization problems considered.

The only remaining maximization problems are when $\rho = \{0, 1, \ldots, k\}$, for some $k \in \mathbb{N}^+$. We focus on these problems for the remainder of this section. We show them to be $\mathcal{NP}$-hard, and obtain nearly tight bounds on their approximabilities. The results are summarized in the following theorem. Let *opt* denote the size of the optimal solution of the instance.

Theorem 8 *The $\{0, 1, \ldots, k\}$-IS maximization problem, for $k \in \mathbb{N}^+$, can be approximated within $O(\sqrt{n})$ in polynomial time, but not within $O(n^{1/(k+1)-\epsilon})$ nor $O(opt^{1-\epsilon})$, for any fixed $\epsilon > 0$, unless $\mathcal{NP} = \mathcal{ZPP}$.*

3.1 Approximation algorithm

We now give an algorithm that approximates some important problems on set systems. These results are interesting in their own right. Simple reductions then imply the same approximation for the $\{0, 1, \ldots, k\}$-IS problems.

Definition 9 *The* Set Packing *problem is the following: Given a base set S and a collection $\mathcal{C}$ of subsets of S, find a collection $\mathcal{C}' \subseteq \mathcal{C}$ of disjoint sets that is of maximum cardinality.*

Set Packing and Maximum Independent Set can be shown to be mutually reducible by approximation-preserving reductions. Given a graph, form a set system with a base element for each edge and a set corresponding to a vertex containing the elements corresponding to incident edges. Then independent sets in the graph are in one-to-one correspondence with packings of the set system. Thus, the $O(n/\log^2 n)$ approximation of Independent Set carries over to Set Packing. This approximation is in terms of n, the number of sets in the set system.

An alternative would be to measure the approximation in terms of m, the size of the base system. For this, there is an obvious upper bound of m, since that is the maximum size of any solution. Another easy upper bound is the maximum cardinality k of a set in the solution, since any maximal solution will find a solution of size at least m/k. However, k can be as large as m, and no better bounds were known in terms of m, to the best of our knowledge.

Theorem 10 *Set Packing can be approximated within $\sqrt{m}$, where m is the size of the base set, in time linear in the input size.*

Proof. A greedy algorithm is given in Fig. 2. In each step, it chooses a smallest set and removes from the collection all sets containing elements from the selected set.

```
Greedy(S,C)
t ← 0
repeat
  t ← t + 1
  X_t ← C ∈ C of minimum cardinality
  Z_t ← {C ∈ C : X ∩ C ≠ ∅ }
  C ← C − Z_t
until |C| = 0
Output {X_1, X_2, ..., X_t}
```

Fig. 2. Greedy set packing algorithm

Let $M = \lfloor \sqrt{m} \rfloor$. Observe that $\{Z_1, \ldots, Z_t\}$ forms a partition of $\mathcal{C}$. Let i be the index of some iteration of the algorithm, i.e. $1 \leq i \leq t$. All sets in Z_i

contain at least one element of X_i, thus the maximum number of disjoint sets in Z_i is at most the cardinality of X_i. On the other hand, every set in Z_i is of size at least X_i, so the maximum number of disjoint sets in Z_i is also at most $\lfloor m/|X_i| \rfloor$. Thus, the optimal solution contains at most $\min(|X_i|, \lfloor m/|X_i| \rfloor) \leq \max_{x \in \mathbb{N}} \min(x, \lfloor m/x \rfloor) = M$ sets from Z_i.

Thus, in total, the optimal solution contains at most tM sets, when the algorithm finds t sets, for a ratio of at most M. ∎

The *Strong Stable Set* problem is the $\{0,1\}$-IS maximization problem. A strong stable set, also known as a *2-packing*, corresponds to a set of vertices of pairwise distance at least three. The Strong Stable Set problem reduces to Set Packing in the following way. Recall that $N[v] = N(v) \cup \{v\}$. Given a graph $G = (V, E)$, construct a set system $(S, \mathcal{C})$ with $S = V$ and $\mathcal{C} = \{N[v] : v \in V\}$. Then, a strong stable set corresponds to a set of nodes whose closed neighborhoods do not overlap, thus forming a set packing of $(S, \mathcal{C})$.

Corollary 11 *Strong Stable Set can be approximated within* $\sqrt{n}$.

The *Distance-t Set* problem is that of finding a maximum cardinality set of vertices of mutual distance at least t in a given graph G. It corresponds to finding a maximum independent set in the power graph G^{t-1}. If A is the adjacency matrix of G and I is the identity matrix, then the adjacency matrix of G^{t-1} is obtained by computing $(A+I)^{t-1}$, replacing non-zero entries by ones, and eliminating self-loops. The Strong Stable Set problem on G is the Distance-3 Set problem, or that of finding a maximum independent set in G^2. Since the Distance-$2q+1$ Set problem is that of finding a maximum independent set in $(G^q)^2$, the odd case is a restricted case of the Strong Stable Set problem.

Corollary 12 *The Distance-t Set problem can be approximated within* $\sqrt{n}$ *for any odd* t.

We now extend the application of the greedy set packing algorithm.

Definition 13 *A* k-matching *of a set system* $(S, \mathcal{C})$ *is a collection* $\mathcal{C}' \subseteq \mathcal{C}$ *such that each element in* S *is contained in at most* k *sets in* $\mathcal{C}'$.

In particular, a 1-matching is precisely a set packing. The *k-Matching* problem is that of finding a k-Matching of maximum cardinality, i.e. containing the greatest number of sets.

Observe that the sizes of maximum set packings and maximum k-matchings can vary widely. Consider the set system that is the dual of a complete graph, namely $S = \{e_{i,j} : 1 \leq i < j \leq n\}$, $\mathcal{C} = \{C_x : 1 \leq x \leq n\}$ and $C_x = \{e_{i,x} : 1 \leq i < x\} \cup \{e_{x,j} : x < j \leq n\}$. Then, the whole system is a 2-matching while any set packing is of unit size. Thus, the ratio between the two can be as much as $\sqrt{m}$. We nevertheless find that the algorithm for Set Packing still yields $O(\sqrt{m})$ approximations for k-Matching.

Theorem 14 *The greedy set packing algorithm approximates the k-Matching problem within* $k\sqrt{m}$.

Proof. The sum of the sizes of sets in a k-matching is at most km. Thus, if each set contains at least q elements, then the matching contains at most $\lfloor \frac{km}{q} \rfloor$ sets.

Consider any iteration i. Each set in Z_i is of size at least $|X_i|$. Thus, the optimal k-matching OPT contains at most $\lfloor \frac{km}{|X_i|} \rfloor$ sets from Z_i. On the other hand, OPT never contains more than $k|X_i|$ sets from Z_i, since it contains at most k sets containing a particular element from X_i. Thus,

$$|OPT \cap Z_i| \leq k \min(|X_i|, m/|X_i|) = k\sqrt{m}.$$

Hence, the optimal k-matching contains at most $tk\sqrt{m}$ sets,

$$|OPT| = \sum_{i=1}^{t} |OPT \cap Z_i| \leq tk\sqrt{m}.$$

while the algorithm obtains t sets, for a performance ratio of $k\sqrt{m}$. ∎

This also translates to a similar ratio for the other $\{0, 1, \ldots, k\}$-IS problems. While we can again show that the size of a maximum strong stable set and a maximum $\{0, 1, 2\}$-IS can differ by a factor of as much as $\Omega(\sqrt{n})$, the analysis nevertheless works out.

Corollary 15 *The $\{0, 1, \ldots, k\}$-IS problem, for $k \geq 1$ is approximable within $O(\sqrt{n})$.*

Proof. Given an instance G to $\{0, 1, \ldots, k\}$-IS, form the set system of closed neighborhoods, as in the reduction of Strong Stable Set to Set Packing. Recall that the number of base elements m now equals the number of sets n. Clearly the solution output by the greedy set packing solution is a feasible solution, since it forms a $\{0, 1\}$-IS.

Observe that any solution to the $\{0, 1, \ldots, k\}$-IS problem of G corresponds to a k-matching in the derived set system (while the converse is not true). Hence, by Theorem 14 the size of the algorithm's solution is also within $O(\sqrt{n})$ of the optimal $\{0, 1, \ldots, k\}$-IS solution. ∎

3.2 Approximation lower bound

A set system is also sometimes referred to as a hypergraph, where the hypervertices correspond to the base elements and hyperedges correspond to the sets of the set system. A *t-uniform hypergraph* is a set system where the cardinality of all edges is t. A subset S of V is an *independent set* if no hyperedge is fully contained in S.

Our lower bound rests on the following reduction from the problem of finding an approximately maximum independent set in a hypergraph.

Lemma 16. *If the $\{0, 1, \ldots, k\}$-IS maximization problem can be approximated within $f(n)$, then the Maximum Independent Set problem in $(k+1)$-uniform hypergraphs can be approximated within $O(f(n)^{k+1})$.*

Also, if the former problem can be approximated within $g(opt)$, as a function of the optimal solution value opt, so can the latter.

Proof. Given a hypergraph H, construct a graph G as follows. G contains a vertex for each node and each hyperedge of H. The hyperedge-vertices form a clique, while the node-vertices are independent. A hyperedge-vertex is adjacent precisely to those node-vertices that correspond to nodes incident on the hyperedge.

We first claim that any independent set S in the hypergraph H is a $\{0, 1, \ldots, k\}$-IS in G. Clearly it is an independent set in G since it consists only of node-vertices. Each node-vertex thus has a ρ-value of 0. Hyperedge-vertices have exactly k node-vertices as neighbors and not all of those can be in S given the independence property of S in H. Thus, hyperedge-vertices have a ρ-value of at most $k - 1$.

Any $\{0, 1, \ldots, k\}$-IS S in G can contain at most one hyperedge-vertex, and if we eliminate that possible vertex from S, it can be verified that the remainder corresponds to an independent set in H.

Taken together, any approximate solution to $\{0, 1, \ldots, k\}$-IS gives an equally approximate independent set of H, within an additive one. Hence, ratios in terms of *opt* carry over immediately. For approximations in terms of the input size, we must factor in that $|V(G)| = |V(H)| + |E(H)| = O(|V(H)|^{k+1})$. ∎

To obtain the theorem, we need to show that Maximum Independent Set in hypergraphs is hard to approximate. We sketch here how the $n^{1-\epsilon}$ inapproximability result of [4] translates to the same bound for the case of uniform hypergraphs. Given a graph G, form a hypergraph H on the same vertex set, with hyperedges for any $(k + 1)$-tuples such that some pair of vertices in the tuple form an edge in G. Then, we have a one-to-one correspondence between independent sets (of cardinality at least k) in G and in H.

Observe that in the case $k = 1$, the Strong Stable Set problem, we obtain a lower bound of $\Omega(n^{1/2-\epsilon})$ which is essentially tight in light of the upper bound given. The lower bound can be generalized for Set Packing to show that the $O(\sqrt{m})$ approximation in terms of the number of base elements is essentially the best possible.

We also obtain tight lower bounds for the Distance-t Set problems defined earlier.

Theorem 17 *For any $\epsilon > 0$, the Distance-t Set problem is hard to approximate within $n^{1-\epsilon}$ when t is even, and within $n^{1/2-\epsilon}$ when t is odd, $t \geq 3$.*

Proof. First consider the even case, $t = 2q + 2$. Given a graph G, construct a graph H that contains a copy of G, a vertex u adjacent to every vertex of G, and a distinct path of q edges attached to each vertex of G. That is, $V(H) = \{v_i, w_{i,j} : v_i \in V(G), 1 \leq j \leq q\} \cup \{u\}$, and $E(H) = E(G) \cup \{uv_i, v_iw_{i,1}, w_{i,j}w_{i,j+1} : v_i \in V(G), 1 \leq j < q\}$. All pairs of vertices in H are of distance at most $2q + 2 = t$. The only vertices of distance t are pairs $w_{i,q}, w_{j,q}$ of leaves on paths where (v_i, v_j) are non-adjacent. Hence, a Distance-t Set in H is in one-to-one correspondence with an independent set in G. Further, the size of H is linear in the size of G. Thus, the Distance-t Set problem, for t even, is no easier to approximate than the IS problem.

For the lower bound for the odd case, we similarly append paths to each vertex of the construction for the Strong Stable Set problem. We invite the reader to verify the details. ∎

4 Conclusion

We have investigated the complexity of decision and optimization problems over independent sets with domination constraints. These problems belong to the framework of (σ, ρ)-problems. Our results constitute a complete complexity classification for the cases when $\sigma = \{0\}$, up to $\mathcal{P}$ vs. $\mathcal{NP}$ for the decision problems, and with tight approximability bounds for the optimization problems. The approximation results extended also to several related independence problems. The complexity of problems for other cases of $\sigma \subseteq \mathbb{N}$ remain to be investigated in detail.

Acknowledgement

A comment by Hiroshi Nagamochi prompted us to greatly improve an early algorithm.

References

1. R. B. Boppana and M.M. Halldórsson, Approximating maximum independent sets by excluding subgraphs, *BIT*, 32 (1992), 180–196.
2. M. R. Garey and D. S. Johnson, *Computers and Intractability* (Freeman, New York, 1979).
3. M.M. Halldórsson, Approximating the minimum maximal independence number, *Information Processing Letters* 46 (1993), 169–172.
4. J. Håstad, Clique is hard to approximate within $n^{1-\epsilon}$, In *Proc. 37th IEEE Symp. on Found. of Comput. Sci.*, (1996), 627–636.
5. S. Khanna, M. Sudan and D. P. Williamson, A complete classification of the approximability of maximization problems derived from boolean constraint satisfaction. in *Proc. 29th ACM Symp. on Theory of Computing*, (1997), 11–20.
6. J. Kratochvíl, Perfect codes in general graphs, monograph, Academia Praha (1991).
7. J. Kratochvíl, P. Manuel and M. Miller, Generalized domination in chordal graphs, *Nordic Journal of Computing* 2 (1995), 41–50
8. T.J. Schaefer, The complexity of satisfiability problems, In *Proc. 10th ACM Symp. on Theory of Computing* (1978), 216–226.
9. J.A. Telle, Characterization of domination-type parameters in graphs, *Proceedings of 24th Southeastern International Conference on Combinatorics, Graph Theory and Computing -Congressus Numerantium* Vol.94 (1993), 9–16.
10. J.A. Telle, Complexity of domination-type problems in graphs, *Nordic Journal of Computing* 1 (1994), 157–171.