Hardness of approximation of the Balanced Complete Bipartite Subgraph problem

Uriel Feige Shimon Kogan

Department of Computer Science and Applied Mathematics
Weizmann Institute, Rehovot 76100, Israel
{uriel.feige,shimon.kogan}@weizmann.ac.il

May 6, 2004

Abstract

We prove that the Maximum Balanced Complete Bipartite Subgraph (BCBS) problem is hard to approximate within a factor of $2^{(\log n)^{\delta}}$ for some $\delta > 0$ under the plausible assumption that 3-SAT $\notin DTIME\left(2^{n^{3/4+\epsilon}}\right)$ for some $\epsilon > 0$. We also show that it is NP-hard to approximate the BCBS problem within a constant factor under the assumption that it is NP-hard to approximate the maximum clique problem within a factor of $n/2^{c\sqrt{\lg n}}$ for some small enough $c > 0$. Furthermore we show that the same hardness of approximation results holds for the Maximum Edge Biclique problem.

1 Introduction and definitions

A balanced bipartite graph is a bipartite graph in which both partite sets are of the same cardinality. Let $G(U, V, E)$ be a balanced bipartite graph. A vertex set C in G is called a *biclique* if $uv \in E$ for all $u \in C \cap U, v \in C \cap V$. A vertex set $C \subseteq G$ is called a *balanced biclique* if C is a biclique and $|C \cap U| = |C \cap V|$. The size of a balanced biclique C is defined as $|C \cap U|$.

In this paper we consider the following problems.

- The *maximum balanced complete bipartite subgraph (BCBS) problem* is the problem of finding a maximum balanced biclique in a balanced bipartite graph.
- The *maximum edge biclique problem* is the problem of finding a biclique with maximum number of edges in balanced bipartite graph.

The maximum BCBS problem already appears in the book of Gary and Johnson [GJ79] (problem GT24) where it is stated that this problem is NP-hard, while the exact reduction from Clique is described in [Joh87]. Another nice NP-hardness proof for this problem is given in [ADL$^+$94]. The problem of finding a biclique which contains the maximum number of nodes in a bipartite graph can be solved in polynomial time via the matching algorithm on the bipartite edge complement of the graph. The maximum BCBS problem is one of the few problems still remaining for which we have neither a hardness of approximation result, nor a 'good' approximation algorithm. The maximum edge biclique problem was shown to be NP-hard in [Pee00].

The maximum BCBS problem and the maximum edge biclique problem have applications in computational biology. Cheng and Church in [CC00] apply those problems for biclustering of expression data of genes. Specifically they create a bipartite graph in which one side represent genes and the other side their properties. The goal is to find a maximum balanced biclique or a maximum edge biclique in such a graph. The maximum BCBS problem is applied in VLSI theory for PLA-folding [RL88], where PLA-folding is a process used to reduce the size of programmable logical arrays. For a full description of PLA-folding and it's connection to the maximum BCBS problem see for example [RL88, AM99]. An interesting application of the maximum edge biclique problem to conjunctive clustering is described in [MRS03]. Applications of the maximum edge biclique problem to manufacturing optimization can be found in [DKST01].

While it is unknown whether the problems defined above are hard to approximate, we think that this is indeed the case.

Conjecture 1.1. The maximum BCBS and maximum edge biclique problems are hard to approximate within a factor of $O(n^\epsilon)$ for some $\epsilon > 0$.

In [Fei02b] it is shown that conjecture 1.1 is valid under the following assumption

Assumption 1.2. *Let Δ be a sufficiently large constant independent of n. There is no polynomial time algorithm that refutes most $3CNF$ formulas with n variables and Δn clauses, and never wrongly refutes a satisfiable formula.*

We will prove that the maximum BCBS and the maximum edge biclique problems are hard to approximate under the plausible assumption that 3-SAT has no subexponential algorithm.

Theorem 1.3. *If maximum BCBS can be approximated within a factor of $2^{(\log n)^\delta}$ for every $\delta > 0$, then 3-SAT can be solved in time $2^{n^{3/4+\epsilon}}$ for every $\epsilon > 0$.*

Theorem 1.4. *If maximum edge biclique can be approximated within a factor of* $2^{(\log n)^\delta}$ *for every constant* $\delta > 0$*, then* 3*-SAT can be solved in time* $2^{n^{3/4+\epsilon}}$ *for every constant* $\epsilon > 0$*.*

In section 2 we show some preliminary results which will be used to prove the theorems above.

2 Preliminary results

2.1 A generalization of Turan's Theorem

Definition 2.1. Given a graph $G(V, E)$, we denote by $C_k(G)$ the set of k-cliques in G (for example $C_2(G)$ is the set of edges of G).

The following generalization of the Turan theorem was first proved in [Zyk] and later ,independently, by [Sau71].

Theorem 2.1. *For all* $n \geq p \geq k$*, if* $G(V, E)$ *is a graph on* n *vertices, without a* $p+1$*-clique, then*

$$|C_k(G)| \leq \binom{p}{k} \cdot \left(\frac{n}{p}\right)^k$$

The special case of theorem 2.1 with $k = 2$ is the Turan Theorem ([Tur41]).

Corollary 2.2. *Let* $\alpha > 0$ *be a constant. Let* $p = \alpha n$ *and* $k = o(n^{2/3})$*. If* $G(V, E)$ *is a graph on* n *vertices, without a* $p+1$*-clique, then for large enough* n

$$|C_k(G)| \leq 2\binom{n}{k} \cdot e^{\frac{k^2}{2n}(1-\frac{1}{\alpha})}$$

Proof: In [FS96] (pp. 189) it was proven that as $n \to \infty$ the function $Q(n, k) = \frac{n!}{(n-k)!n^k}$ where $k = o(n^{2/3})$, satisfies

$$Q(n, k) = (1 + o(1))e^{-\frac{k^2}{2n}}$$

Thus

$$\begin{aligned}
|C_k(G)| &\leq \binom{p}{k} \cdot \left(\frac{n}{p}\right)^k && \text{by theorem 2.1} \\
&= \binom{n}{k} \cdot \frac{Q(p,k)}{Q(n,k)} \\
&= (1+o(1))\binom{n}{k} e^{\frac{k^2}{2n} - \frac{k^2}{2p}} \\
&= (1+o(1))\binom{n}{k} \cdot e^{\frac{k^2}{2n}\left(1-\frac{1}{\alpha}\right)}
\end{aligned}$$

■

2.2 A variant of the sparsification lemma

The following sparsification lemma was proven in [IPZ01].

Lemma 2.3. *For all $\epsilon > 0$, there is a constant C such that any 3-SAT formula Ψ_i with n variables, can be expressed as $\Phi = \bigvee_{i=1}^{t} \Psi_i$, where $t \leq 2^{\epsilon n}$ and each Ψ_i is a 3-SAT formula with at most Cn clauses. Moreover this disjunction can be computed by an algorithm running in time $poly(n)2^{\epsilon n}$.*

We will need the following variant of the lemma above.

Lemma 2.4. *For all $\epsilon > 0$, any 3-SAT formula Ψ_i with n variables, can be expressed as $\Phi = \bigvee_{i=1}^{t} \Psi_i$, where $t \leq 2^{n^{3/4+\epsilon}}$ and each Ψ_i is a 3-SAT formula with $O(n^{3/2})$ clauses. Moreover this disjunction can be computed by an algorithm running in time $poly(n)2^{n^{3/4+\epsilon}}$.*

Proof: The proof of lemma 2.4 is almost identical to the proof the lemma 2.3. Indeed it follows from setting the following values to the parameters θ_1, θ_2 which are used by the algorithm stated in the proof of theorem 1 in [IPZ01]: $\theta_1 = n^{\frac{1}{4}}$, $\theta_2 = n^{\frac{1}{2}}$. ■

3 Proof of Theorem 1.3

We will need the following bounds on binomial coefficients.

Lemma 3.1. *For all $d, k, n \in \mathbb{N}$, $\binom{n-d}{k} \leq \binom{n}{k} e^{-\frac{dk}{n}}$*

Proof:

$$\begin{aligned}\binom{n-d}{k} &= \binom{n}{k} \cdot \frac{(n-k)(n-k-1)\dots(n-k-d+1)}{n(n-1)\dots(n-d+1)} \\ &= \binom{n}{k} \cdot \left(1-\frac{k}{n}\right)\left(1-\frac{k}{n-1}\right)\dots\left(1-\frac{k}{n-d+1}\right) \\ &\leq \binom{n}{k}\left(1-\frac{k}{n}\right)^d \\ &\leq \binom{n}{k} e^{-\frac{dk}{n}}\end{aligned}$$

■

Lemma 3.2. *For all* $d, k, n \in \mathbb{N}$, $\binom{n+d}{k} \leq \binom{n}{k} e^{\frac{dk}{n-k}}$

Proof:

$$\begin{aligned}\binom{n+d}{k} &= \binom{n}{k} \cdot \frac{(n+1)(n+2)\dots(n+d)}{(n-k+1)(n-k+2)\dots(n-k+d)} \\ &= \binom{n}{k} \cdot \left(1+\frac{k}{n-k+1}\right)\left(1+\frac{k}{n-k+2}\right)\dots\left(1+\frac{k}{n-k+d}\right) \\ &\leq \binom{n}{k}\left(1+\frac{k}{n-k}\right)^d \\ &\leq \binom{n}{k} e^{\frac{dk}{n-k}}\end{aligned}$$

■

Assumption 1: Maximum BCBS can be approximated in polynomial time within a factor of $2^{(\log n)^\alpha}$ for every constant $\alpha > 0$.

Definition 3.1. Let $\mathcal{G}_{1/2}$ be the family of graphs on n vertices which satisfy $\omega(G) \geq \frac{n}{2}$ where $\omega(G)$ is the size of the maximum clique in G. Let $\mathcal{G}_{1/2-\gamma}$ the family of graphs which satisfy $\omega(G) \leq (\frac{1}{2}-\gamma)n$, where $\gamma > 0$.

Theorem 3.3. *Let* $G(V, E)$ *be a graph on* n *vertices. If assumption 1 holds then for all* $\delta > 0, \gamma > 0$ *one can distinguish in time* $2^{n^{\frac{1}{2}+\delta}}$ *whether* $G \in \mathcal{G}_{1/2}$ *or* $G \in \mathcal{G}_{1/2-\gamma}$.

Proof: Create a balanced bipartite graph $G'(X, Y, E')$ with $|X| = |Y| = \binom{n}{k}$ where $k = t^{\frac{1}{2}+\delta}$ and $t = \frac{n}{2}$ in the following way.

- each vertex $x \in X$ corresponds to a different set of k vertices in G.
- each vertex $y \in Y$ corresponds to a different set of k vertices in G.
- $(x, y) \in E'$ iff the sets corresponding to x and y are disjoint and the set corresponding to y induces a clique in G.

If $G \in \mathcal{G}_{1/2}$ then G' contains a balanced biclique of size $\binom{t}{k}$, since if C is a clique of size t in G then G' contains a balanced biclique with all vertices in partite set X which correspond to sets in $V \backslash C$ and all the vertices in partite set Y which correspond to sets in C. Thus by assumption 1 we can find in G' some biclique $B(B_x, B_y)$ of size at least $\epsilon\binom{t}{k}$ where $\epsilon = e^{-t^\delta}$. The time it takes to find such a biclique is polynomial in the size of G' and that is at most $2^{n^{\frac{1}{2}+2\delta}}$ for large enough n.

Now we will show that if G' contains a balanced biclique $B(B_x, B_y)$ of size $\epsilon\binom{t}{k}$ where $t = \frac{n}{2}$ and $\epsilon = e^{-t^\delta}$ then G contains a clique of size $(\frac{1}{2} - \gamma)n$ for any constant $\gamma > 0$ and thus $G \notin \mathcal{G}_{1/2-\gamma}$ for any $\gamma > 0$. Let $V_x \subseteq V$ be the union of the sets corresponding to the vertices in B_x and $V_y \subseteq V$ be the union of the sets corresponding to the vertices in B_y. Notice that $V_x \cap V_y = \varnothing$. Since

$$\begin{aligned} |B_x| \geq \epsilon\binom{t}{k} &= e^{(\frac{t\ln\epsilon}{k})\frac{k}{t}}\binom{t}{k} \\ &\geq \binom{t + \frac{t\ln\epsilon}{k}}{k} && \text{by lemma 3.1} \end{aligned}$$

we have that $|V_x| \geq t(1 + \frac{\ln\epsilon}{k})$. Furthermore by the same argument we have that $|V_y| \geq t(1 + \frac{\ln\epsilon}{k})$ and thus $t(1 + \frac{\ln\epsilon}{k}) \leq |V_y| \leq t(1 - \frac{\ln\epsilon}{k})$. Let $G_y(V_y, E_y)$ be the subgraph of G induced by V_y. This graph contains at least $|B_y| = \epsilon\binom{t}{k}$ cliques of size k. if $\omega(G_y) < \beta|V_y|$ for some constant $\beta < 1$ then by corollary 2.2 we will have that

$$\begin{aligned} |B_y| &\leq 2\binom{|V_y|}{k} \cdot e^{\frac{k^2}{2|V_y|}(1-\frac{1}{\beta})} \\ &\leq 2\binom{t(1 - \frac{\ln\epsilon}{k})}{k} \cdot e^{\frac{k^2}{2t(1-\frac{\ln\epsilon}{k})}(1-\frac{1}{\beta})} \\ &\leq 2\binom{t}{k} \cdot e^{-\frac{t\ln\epsilon}{t-k} + \frac{k^2}{2t(1-\frac{\ln\epsilon}{k})}(1-\frac{1}{\beta})} && \text{by lemma 3.2} \\ &\leq 2\binom{t}{k} \cdot e^{2t^\delta + n^{2\delta}(1-\frac{1}{\beta})/2} && \text{as } k = t^{\frac{1}{2}+\delta} \text{ and } \epsilon = e^{-t^\delta} \\ &< \epsilon\binom{t}{k} && \text{for large enough } n \end{aligned}$$

we got a contradiction and thus G contains a clique of size $\beta t(1+\frac{\ln\epsilon}{k}) \geq \beta(\frac{n}{2}-2\sqrt{n})$ for all $\beta<1$ and we may conclude that G contains a clique of size $\beta\frac{n}{2}$ for all $\beta<0$. We have shown the if $G\in\mathcal{G}_{1/2}$ then G' contains a balanced biclique of size $\binom{t}{k}$ and if $G\in\mathcal{G}_{1/2-\gamma}$ for some $\gamma>0$ then G' does not contain a balanced biclique of size $\epsilon\binom{t}{k}$ and thus we're done. ■

Definition 3.2. Language $L\in PCP_{1,\beta}[r,q]$ if there is a randomized polynomial time algorithm V which gets as an input a string x, and has access to a witness w, with the following properties:

- **Completeness.** For every $x\in L$ there is a witness w such that $V^w(x)$ accepts (with probability 1).
- **Soundness.** For every $x\notin L$ and every w, $Pr[V^w(x) \text{ accepts }]\leq\beta$.

the verifier V uses up to r random bits in order to list at most q bit locations. Then it queries w at these q locations and gets back the bit values. Finally, based on the values received, it decides whether to accept or reject.

We show a reduction from 3-SAT to BCBS. Let Φ be an instance of 3-SAT on n variables. The reduction has 3 steps, and uses the following theorem which was proven in [BSSVW02].

Theorem 3.4. *(Short PCPs). There exist constants $\beta<1, q<\infty$, and a function $r(n)=\log n+O(\sqrt{\log n}\log\log n)$ such that $SAT\in PCP_{1,\beta}[r,q]$.*

Let Ψ be an arbitrary 3-SAT formula.

1. Using lemma 2.4 express formula Φ as $\Phi=\bigvee_{i=1}^{t}\Psi_i$, where $t\leq 2^{n^{3/4+\epsilon}}$ and each Ψ_i is a 3-SAT formula with $O(n^{3/2})$ clauses.
2. Using theorem 3.4 obtain for each Ψ_i a PCP verifier VER_i which uses $r=(\frac{3}{2}+\epsilon)\log n$ random bits and queries q bits. Given the PCP verifier VER_i define the corresponding FGLSS graph $G_i(V_i,E_i)$ (see [FGL+96]) as follows: The vertices of this graph are all accepting patterns $\tau=(S,\nu)$. There is an edge between two accepting patterns (S,ν) and (S',ν') if ν,ν' assign the same value to the bits common to S and S'. Thus the graph G_i contains at most $2^{r+q}=2^q n^{\frac{3}{2}+\epsilon}=O(n^{\frac{3}{2}+\epsilon})$ vertices. Add a clique C of size $O(n^{\frac{3}{2}+\epsilon})$ to G_i and connect each vertex in C to each vertex in V_i, call the resulting graph $G_i'(V_i',E_i')$. It's easy to see that one can choose the size of C in such a manner that $\omega(G_i')\geq\frac{1}{2}|V_i'|$ if Ψ_i is satisfied and $\omega(G_i')\leq(\frac{1}{2}-\delta)|V_i'|$ for some constant δ if Ψ_i is unsatisfiable.
3. Now by theorem 3.3 we can decide in time $2^{n^{\frac{3}{4}+\epsilon}}$ whether $\omega(G_i')\geq\frac{1}{2}|V_i'|$ or $\omega(G_i')\leq(\frac{1}{2}-\delta)|V_i'|$ and thus we are done.

A linear 3-SAT formula is a 3-SAT formula on n variables which contains $O(n)$ clauses. We note that if maximum BCBS can be approximated within a factor of $2^{(\log n)^\delta}$ for every $\delta > 0$, then linear 3-SAT can be solved in time $2^{n^{1/2+\epsilon}}$ for every $\epsilon > 0$. Thus maximum BCBS is hard to approximate within a factor of $2^{(\log n)^\delta}$ for some small enough $\delta > 0$ under the assumption that linear 3-SAT $\notin DTIME\left(2^{n^{1/2+\epsilon}}\right)$ for some $\epsilon > 0$.

4 Proof of Theorem 1.4

We will need the following bound on binomial coefficients.

Lemma 4.1. *For all $d, k, n \in \mathbb{N}$, $\binom{n-d}{k}\binom{n+d}{k} \leq \binom{n}{k}^2$*

Proof:

$$\begin{aligned}
\binom{n-d}{k}\binom{n+d}{k} &= \binom{n}{k}^2 \cdot \frac{(n-k)(n-k-1)\dots(n-k-d+1)}{n(n-1)\dots(n-d+1)} \cdot \frac{(n+1)(n+2)\dots(n+d)}{(n-k+1)(n-k+2)\dots(n-k+d)} \\
&= \binom{n}{k}^2 \cdot \prod_{i=1}^{d} \frac{n-k-d+i}{n-k+i} \cdot \frac{n+i}{n-d+i} \\
&\leq \binom{n}{k}^2
\end{aligned}$$

The last inequality follows from the fact that for each i

$$\begin{aligned}
(n-k-d+i)(n+i) &= (n-k+i)(n+i) - d(n+i) \\
&\leq (n-k+i)(n+i) - d(n+i) + kd \\
&= (n-k+i)(n-d+i)
\end{aligned}$$

■

Lemma 4.2. *For all $d, k, n \in \mathbb{N}$,*

$$\binom{n-d}{k}\binom{n+d}{k} \leq \frac{n^{2k}}{k!^2} \cdot e^{-kd^2/n^2}$$

Proof:

$$\begin{aligned}\binom{n-d}{k}\binom{n+d}{k} &= \frac{(n-d)!}{(n-d-k)!k!}\cdot\frac{(n+d)!}{(n+d-k)!k!}\\ &= \frac{(n-d)!}{(n-d-k)!(n-d)^k}\cdot\frac{(n+d)!}{(n+d-k)!(n+d)^k}\cdot\frac{n^{2k}}{k!^2}\left(1-\frac{d^2}{n^2}\right)^k\\ &\leq \frac{n^{2k}}{k!^2}\left(1-\frac{d^2}{n^2}\right)^k\\ &\leq \frac{n^{2k}}{k!^2}e^{-kd^2/n^2}\end{aligned}$$

■

Lemma 4.3. *For large enough n and $k = o(n^{2/3})$ we have*

$$\binom{n}{k} \geq \frac{n^k}{2k!}\cdot e^{-\frac{k^2}{2n}}$$

Proof:

$$\begin{aligned}\binom{n}{k} &= \frac{n!}{(n-k)!k!}\\ &= \frac{n!}{(n-k)!n^k}\cdot\frac{n^k}{k!}\\ &\geq \frac{1}{2}e^{-\frac{k^2}{2n}}\cdot\frac{n^k}{k!}\end{aligned} \tag{4.1}$$

Where the last inequality follow from the fact ([FS96], pp. 189) that as $n\to\infty$ the function $Q(n,k) = \frac{n!}{(n-k)!n^k}$ where $k = o(n^{2/3})$, satisfies

$$Q(n,k) = (1+o(1))e^{-\frac{k^2}{2n}}$$

■

Assumption 2: Maximum edge biclique can be approximated in polynomial time within a factor of $2^{(\log n)^\alpha}$ for every constant $\alpha > 0$.

Recall definition 3.1. Let $\mathcal{G}_{1/2}$ be the family of graphs on n vertices which satisfy $\omega(G) \geq \frac{n}{2}$ where $\omega(G)$ is the size of the maximum clique in G. Let $\mathcal{G}_{1/2-\gamma}$ the family of graphs which satisfy $\omega(G) \leq (\frac{1}{2}-\gamma)n$, where $\gamma > 0$.

Theorem 4.4. *Let $G(V,E)$ be a graph on n vertices. If assumption 2 holds then for all $\delta > 0, \gamma > 0$ one can distinguish in time $2^{n^{\frac{1}{2}+\delta}}$ whether $G \in \mathcal{G}_{1/2}$ or $G \in \mathcal{G}_{1/2-\gamma}$.*

Proof: Create a balanced bipartite graph $G'(X,Y,E')$ with $|X| = |Y| = \binom{n}{k}$ where $k = t^{\frac{1}{2}+\delta}$ and $t = \frac{n}{2}$ in the following way.

- each vertex $x \in X$ corresponds to a different set of k vertices in G.
- each vertex $y \in Y$ corresponds to a different set of k vertices in G.
- $(x,y) \in E'$ iff the sets corresponding to x and y are disjoint and the set corresponding to y induces a clique in G.

Notice that the construction above is the same construction that we used in theorem 3.3. If $G \in \mathcal{G}_{1/2}$ then G' contains a biclique with at least $\binom{t}{k}^2$ edges, since if C is a clique of size t in G then G' contains a balanced biclique with all vertices in partite set X which correspond to sets in $V \backslash C$ and all the vertices in partite set Y which correspond to sets in C. Thus by assumption 2 we can find in G' some biclique $B(B_x, B_y)$ with at least $\epsilon\binom{t}{k}^2$ edges where $\epsilon = e^{-t^\delta}$. The time it takes to find such a biclique is polynomial in the size of G' and that is at most $2^{n^{\frac{1}{2}+2\delta}}$ for large enough n.

Now we will show that if G' contains a biclique $B(B_x, B_y)$ with at least $\epsilon\binom{t}{k}^2$ edges where $t = \frac{n}{2}$ and $\epsilon = e^{-t^\delta}$ then G contains a clique of size $(\frac{1}{2} - \gamma)n$ for any constant $\gamma > 0$ and thus $G \notin \mathcal{G}_{1/2-\gamma}$ for any $\gamma > 0$. Let $V_x \subseteq V$ be the union of the sets corresponding to the vertices in B_x and $V_y \subseteq V$ be the union of the sets corresponding to the vertices in B_y. Notice that $V_x \cap V_y = \varnothing$. Suppose that $|V_y| = t + r$ for some integer r s.t. $-t < r < t$. First we shall show that $|r| < t^{4/5}$. By lemma 4.3 the biclique B contains at least $\frac{\epsilon}{4} \cdot \frac{t^{2k}}{k!^2} \cdot e^{-\frac{k^2}{t}}$ edges. On the other hand if $|V_y| = t + r$ then $|V_x| \leq t - r$ so the number of edges in B is at most $\binom{t-r}{k}\binom{t+r}{k} \leq \frac{t^{2k}}{k!^2} \cdot e^{-kr^2/t^2}$ (by lemma 4.2). Thus the following inequality should hold

$$\begin{aligned} \frac{t^{2k}}{k!^2} \cdot e^{-\frac{kr^2}{t^2}} &\geq \frac{\epsilon}{4} \cdot \frac{t^{2k}}{k!^2} \cdot e^{-\frac{k^2}{t}} \\ &\geq \frac{t^{2k}}{k!^2} \cdot e^{-\frac{k^2}{t} - t^\delta - 4} \end{aligned}$$

thus

$$\frac{kr^2}{t^2} \leq \frac{k^2}{t} + t^\delta + 4$$

and we conclude that

$$\begin{aligned}|r| &\leq \sqrt{kt + \frac{t^{2+\delta}}{k} + 4\frac{t^2}{k}} \\ &\leq \sqrt{t^{3/2+\delta} + t^{3/2} + 4t^{3/2}} \\ &< t^{4/5}\end{aligned}$$

Let $G_y(V_y, E_y)$ be the subgraph of G induced by V_y. Thus G_y is a graph on $t+r$ vertices, which contains at least

$$\epsilon \cdot \frac{\binom{t}{k}^2}{\binom{t-r}{k}}$$

cliques of size k. if $\omega(G_y) < \beta|V_y|$ for some constant $\beta < 1$ then by corollary 2.2 we will have that the number of k-cliques in G_y denoted by $|C_k(G_y)|$ satisfies

$$\begin{aligned}|C_k(G_y)| &\leq 2\binom{|V_y|}{k} \cdot e^{\frac{k^2}{2|V_y|}(1-\frac{1}{\beta})} \\ &= 2\binom{t+r}{k} \cdot e^{\frac{k^2}{2(t+r)}(1-\frac{1}{\beta})} \\ &\leq 2\frac{\binom{t+r}{k}\binom{t-r}{k}}{\binom{t-r}{k}} \cdot e^{\frac{k^2}{4t}(1-\frac{1}{\beta})} \\ &\leq 2\frac{\binom{t}{k}^2}{\binom{t-r}{k}} \cdot e^{\frac{k^2}{4t}(1-\frac{1}{\beta})} && \text{by lemma 4.1} \\ &\leq 2\frac{\binom{t}{k}^2}{\binom{t-r}{k}} \cdot e^{\frac{t^{1+2\delta}}{4t}(1-\frac{1}{\beta})} && \text{as } k = t^{\frac{1}{2}+\delta} \\ &\leq 2\frac{\binom{t}{k}^2}{\binom{t-r}{k}} \cdot e^{\frac{t^{2\delta}}{4}(1-\frac{1}{\beta})} \\ &\leq \epsilon\frac{\binom{t}{k}^2}{\binom{t-r}{k}} && \text{as } \epsilon = e^{-t^\delta}\end{aligned}$$

we got a contradiction and thus G contains a clique of size $\beta|V_y| \geq \beta(t - t^{4/5})$ for all $\beta < 1$ and we may conclude that G contains a clique of size $\beta\frac{n}{2}$ for all $\beta < 0$. We have shown the if $G \in \mathcal{G}_{1/2}$ then G' contains a biclique with at least $\binom{t}{k}^2$ edges and if $G \in \mathcal{G}_{1/2-\gamma}$ for some $\gamma > 0$ then G' does not contain a balanced biclique of size $\epsilon\binom{t}{k}^2$ and thus we're done. ■

The rest of the proof follows exactly as in theorem 1.3.

5 Relations between the approximation hardness of the BCBS problem and the maximum clique problem

Srinivasan in [Sri00] conjectured that independent set is hard to approximate within a factor of $n/2^{\Omega(\sqrt{\log n})}$. In [Kho01] it was proven that independent set is hard to approximate within a factor of $n/2^{(\log n)^{1-\gamma}}$ for some $\gamma > 0$. Currently the best approximation algorithm for clique has an approximation ratio of $O\left(n(\log\log n)^2/(\log n)^3\right)$ [Fei02a]. In this section we prove the following theorem.

Theorem 5.1. *Suppose that BCBS can be approximated within a constant factor, then independent set can be approximated within a factor of $n/2^{c\sqrt{\lg n}}$ for some $c > 0$ in polynomial time.*

For notational reasons we shall work henceforth with a problem which is equivalent to the BCBS problems, namely the Balanced Bipartite Independent Set problem (BBIS). Let $G(U, V, E)$ be a balanced bipartite graph. A vertex set I in G is called a *balanced bipartite independent set* if $|I \cap U| = |I \cap V|$ and $uv \notin E$ for all $u \in I \cap U, v \in I \cap V$. The size of a balanced bipartite independent set I is defined as $|I \cap U|$. The *maximum balanced bipartite independent set (BBIS) problem* is the problem of finding a maximum balanced bipartite independent set in a balanced bipartite graph. Notice that the BBIS and BCBS problems are equivalent with respect to approximation ratio, as any balanced bipartite independent set in G corresponds to a balanced biclique in the graph obtained by complementing all the edges and non edges between the partite sets of G.

5.1 Proof of Theorem 5.1

Let $k = 2^{\sqrt{\lg n}}$. Let $I(G)$ denote a maximum independent of graph G.
Let $BIS(G)$ be an algorithm which approximates BBIS within a constant factor on a bipartite graph G. Notice that as a consequence of theorem 6.1 which is proven in Appendix I we may assume that algorithm $BIS(G)$ approximates BBIS within a constant factor of 2. Recall that we define the size of a balanced bipartite independent set to be the size of one of it's sides.

Definition 5.1. Given a graph $G(V, E)$ where $V = \{v_1, v_2, \ldots, v_n\}$ we denote by $B(G)$ a bipartite graph $G'(X, Y, E')$ with $|X| = |Y| = n$ such that

- for all $1 \leq i < j \leq n$ $(x_i, y_j) \in E' \iff (v_i, v_j) \in E$.
- for all $1 \leq i \leq n$ $(x_i, y_i) \in E'$.

We will show an algorithm $IND(G)$ which approximates independent set within a factor of $O(n/k)$. The first call to the algorithm will be with a graph H on n vertices as an input. We may assume that H contains an independent set of size at least n/k as if $I(H) < n/k$ then we get the required approximation ratio trivially.

Algorithm $IND(G)$

Input: A graph $G(V, E)$ where $|V| = n'$.
Output: An independent set.

1. if $|V| \leq 8k$ return an arbitrary vertex of G.
2. Let (X',Y') be the balanced bipartite independent set returned by $BIS(B(G))$.
3. Set G_1 to be the subgraph of G corresponding to the vertices in X' and G_2 to be the subgraph of G corresponding to the vertices in Y'.
4. if $|G_1| < \frac{n'}{8k}$ set $H = H \setminus G$ and restart the algorithm with graph H as an input.
5. return $IND(G_1) \bigcup IND(G_2)$.

The idea behind the algorithm is very simple. If a graph G contains an independent set of size $n'/2k$ then using a 2-approximation approximating for the BBIS problem we can find two disjoint subgraph G_1, G_2 in G of cardinality $n'/8k$ each, with no edges between them. Applying this method recursively on the subgraphs found we can find a 'large' independent set in G. The only problem we may have is that some subgraph G' of G which we encounter does not contain an independent set of size at least $|G'|/2k$. We will show later in the formal analysis of the algorithm that in this case we may remove this subgraph from the original graph and restart the algorithm with the truncated graph as an input.

Definition 5.2. Let G be a graph with an independent set of size n/t. A vertex induced subgraph T is called poor if it does not contain an independent set of size $|T|/2t$.

The following lemma is from [Fei02a].

Lemma 5.2. *Let G be a graph with an independent set of size n/t. Let $T_1, T_2, \ldots$ be arbitrary disjoint poor subgraphs of G. Let $G'(V', E')$ be the subgraph of G that remains after removing the poor subgraphs. Then $|V'| \geq n/2t$, and G' contains an independent set of size at least $|V'|/t$.*

Let us look at step 4 of algorithm IND. As BIS is a 2-approximation algorithm of BBIS we have that if $|G_1| < n/8k$ then G is a poor graph. Thus each subgraph removed from H in step 4 is a poor graph. We conclude by lemma 5.2 that the inequality $|H| \geq n/2k$ always holds, and furthermore that H always contains an independent set of size at least $n/2k^2$. As $I(H) \geq \frac{n}{2k^2}$ we have that the recursion depth d of algorithm IND on input H satisfies

$$\frac{n}{2k^2(8k)^d} \leq 8k$$

and thus $d \geq \sqrt{\lg n} - O(1)$. Since the algorithm returns an independent set of size 2^d we get an independent set of size $\Omega\left(2^{\sqrt{\lg n}}\right)$ and thus we get the required approximation ratio.

We can prove an identical result for the maximum edge biclique problem.

Theorem 5.3. *Suppose that the maximum edge biclique problem can be approximated within some constant factor, then independent set can be approximated within a factor of $n/2^{c\sqrt{\lg n}}$ for some $c > 0$ in polynomial time.*

Proof: The proof of this theorem is similar to the proof of theorem 5.1 and thus omitted. ■

6 Appendix I

In this appendix we will show that BBIS has some of the self improvement properties that the maximum clique problem has. We will prove this claim by a minor modification of the proofs of self improvement properties of the maximum clique problem (see for example [AL96]).

Definition 6.1. Let $G(X, Y, E)$ be a bipartite graph. Define the k-th graph product G^k in the following way. Let X^k be the k-th Cartesian product of X, i.e. each vertex in X^k is denoted by $(x_1, x_2, \ldots, x_k)$ where $x_i \in X$ for every $1 \leq i \leq k$. Define Y^k in an identical manner. Let the bipartition of G^k be (X^k, Y^k). There is an edge between $(x_1, x_2, \ldots, x_k) \in X^k$ and $(y_1, y_2, \ldots, y_k) \in Y_2^k$ in G^k iff the union of the vertices in both sets is not a bipartite independent set of G.

Theorem 6.1. *If BBIS can be approximated within some constant then it has a polynomial time approximation scheme (PTAS).*

Proof: Let $I(G)$ be the size of the largest balanced bipartite independent set in a bipartite graph $G(X, Y, E)$. It is easy to see that $I(G^k) \geq I(G)^k$. Suppose we have a polynomial time algorithm which approximates BBIS within a factor of $\frac{1}{\epsilon}$ for some $\epsilon > 0$. This algorithm when applied on G^k will return a balanced bipartite independent set I' of size at least $\epsilon I(G)^k$ with bipartition (X', Y'). Let m be the largest integer such that X' contains at least m^k vertices. There must be a coordinate $1 \leq i \leq k$ such that in the vertices of X', written as k-tuples, there are at least m different elements in coordinate i and each such element corresponds to a vertex in X. The same also holds for Y' (with respect to Y) and thus we may extract from (X', Y') a balanced bipartite independent set of size m in G, i.e. a balanced bipartite independent set of size $\epsilon^{1/k} \cdot I(G)$ and so we see that we may achieve an arbitrarily good approximation for our problem. ■

References

[ADL+94] Noga Alon, Richard A. Duke, Hanno Lefmann, Vojtech Rödl, and Raphael Yuster. The algorithmic aspects of the regularity lemma. *J. Algorithms*, 16(1):80–109, 1994.

[AL96] Sanjeev Arora and Carsten Lund. Hardness of approximations. *In Approximation Algorithms for NP-hard Problems, Dorit Hochbaum, Ed. PWS Publishing*, 1996.

[AM99] Claudio Arbib and Raffaele Mosca. Polynomial algorithms for special cases of the balanced complete bipartite subgraph problem. *J. Combin. Math. Combin. Comput.*, 30:3–22, 1999.

[BSSVW02] Eli Ben-Sasson, Madhu Sudan, Salil P. Vadhan, and Avi Wigderson. Randomness-efficient low degree tests and short PCPs via epsilon-biased sets. *Proceedings of the 34th Symposium on the Theory of Computing*, pages 612–621, 2002.

[CC00] Yizong Cheng and George M. Church. Biclustering of expression data. *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 93–103, 2000.

[DKST01] Milind Dawande, Pinar Keskinocak, Jayashankar M. Swaminathan, and Sridhar Tayur. On bipartite and multipartite clique problems. *J. Algorithms*, 41(2):388–403, 2001.

[Fei02a] Uriel Feige. Approximating maximum clique by removing subgraph. *manuscript*, 2002.

[Fei02b] Uriel Feige. Relations between average case complexity and approximation complexity. *Proceedings of the 34th Symposium on the Theory of Computing*, pages 534–543, 2002.

[FGL+96] Uriel Feige, Shafi Goldwasser, László Lovász, Shmuel Safra, and Mario Szegedy. Interactive proofs and the hardness of approximating cliques. *J. ACM*, 43(2):268–292, 1996.

[FS96] Philippe Flajolet and Robert Sedgewick. *Analysis of algorithms*. Addison Wesley, 1996.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A guide to the theory of NP-completeness*. Freeman, San Fransico, 1979.

[IPZ01] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *JCSS*, 63(4):512–530, 2001.

[Joh87] David S. Johnson. The NP-completeness column: An ongoing guide. *Journal of Algorithms*, 8(3):438–448, 1987.

[Kho01] Subhash Khot. Improved inaproximability results for maxclique, chromatic number and approximate graph coloring. *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science*, pages 600–609, 2001.

[MRS03] Nina Mishra, Dana Ron, and Ram Swaminathan. On finding large conjunctive clusters. *Proceedings of the Sixteenth Annual Conference on Learning Theory*, 2003.

[Pee00] René Peeters. The maximum edge biclique problem is NP-complete. *Research Memorandum 789, Faculty of Economics and Business Administration, Tilberg University*, 2000.

[RL88] S. S. Ravi and Errol L. Lloyd. The complexity of near-optimal programmable logic array folding. *SIAM J. Comput*, 17(4):696–710, 1988.

[Sau71] N. Saur. A generalization of turán's theorm. *J. Combin. Theory Ser. B*, 10:109–112, 1971.

[Sri00] Aravind Srinivasan. The value of strong inapproximability results for clique. *Proceedings of the 32th Symposium on the Theory of Computing*, pages 144–152, 2000.

[Tur41] Paul Turán. On an extremal problem in graph theory. *Math. Fiz. Lapok*, 48:436–452, 1941.

[Zyk] A. A. Zykov. On some properties of linear complexes. *Math Sbornik (N.S.) 24 (66) (1949) (in Russian). (English translation: Amer. Math. Soc. Transl. no. 79, 1952).*