Finding a Δ-regular Supergraph of Minimum Order

Hans L. Bodlaender [a], Richard B. Tan [a,b] and Jan van Leeuwen [a]

[a] *Department of Computer Science*
Utrecht University
Padualaan 14, 3584 CH Utrecht
The Netherlands
{hansb,rbtan,jan}@cs.uu.nl

[b] *Department of Computer Science*
University of Sciences & Arts of Oklahoma
Chickasha, OK 73018
U.S.A.

Abstract

Akiyama, Era and Harary [1] proved that every graph of maximum degree Δ is a subgraph of a Δ-regular graph that has at most $\Delta + 2$ additional vertices. We show that, given a graph of maximum degree Δ, a Δ-regular supergraph of it of minimum order can be computed in $O(min\{\Delta^{1.5}|V|^{2.5}, \Delta^6 + \Delta|V|\})$ time.

1 Introduction

Various algorithmic problems in graph theory can be reduced to the case of regular graphs. It is thus of interest to know whether arbitrary, non-regular graphs can be extended to regular graphs with little computational effort, e.g. by adding only a small number of vertices.

König [6] already showed that every graph G of maximum degree Δ is the *induced* subgraph of some Δ-regular graph. Erdös and Kelly [3] obtained a formula for the minimum number of vertices that have to be added to G to obtain such a Δ-regular supergraph. In this note we consider the variant where we do *not* require that G is an induced subgraph.

Akiyama *et al.* [1] showed the following result for the maximum number of vertices that must be added to G to obtain a Δ-regular supergraph of it, now allowing that edges are added between the original vertices of G.

Theorem 1 ([1]) *Let $G = (V, E)$ be a graph of maximum degree Δ. If Δ is odd (even), then G is a subgraph of a Δ-regular graph $H = (V', E')$ with $|V' - V| \leq \Delta + 2$ (respectively $|V' - V| \leq \Delta + 1$).*

Akiyama *et al.* also showed that the result is sharp: for some graphs G the $\Delta + 2$ (respectively $\Delta + 1$) additional vertices are necessary.

In this note we are interested in the problem of determining the *minimum* number of vertices that must be added to a graph G of maximum degree Δ in order to make it Δ-regular (i.e. to obtain a Δ-regular graph H of which it is a subgraph). In Section 3 we show that this problem is tractable, by giving an algorithm for solving this problem that uses $O(\Delta^{1.5}|V|^{2.5})$ time. If Δ is small, a better running time of $O(\Delta^6 + \Delta|V|)$ can be obtained.

In Section 4 we show that a Δ-regular supergraph H of G that is not necessarily of minimum order but satisfies the bounds of Theorem 1 can be computed in $O(\Delta|V|)$ time. The result is based on an algorithmic proof of Theorem 1.

2 Preliminaries

All graphs considered in this paper are assumed to be simple, i.e., there are no parallel edges or self-loops. The degree of a vertex v in a graph G is denoted by $d_G(v)$. If G is clear from the context, we drop the subscript G. The order of a graph $G = (V, E)$ is $|V|$. The complement of a graph G is denoted by $\overline{G}$.

A graph $G = (V, E)$ is Δ-*regular* if every vertex $v \in V$ has degree Δ in G. A graph $G = (V, E)$ is Δ-*regularizable* if G is a subgraph of a Δ-regular graph $H = (V, E')$ with the same vertex set.

Let $f : V \to \mathbf{N}$ be a function, assigning to each vertex a non-negative integer. An f-*factor* of a graph $G = (V, E)$ is a subset $F \subseteq E$ of the edges such that every vertex $v \in V$ is incident to exactly $f(v)$ edges from F. f-factors are also known in the literature as perfect b-matchings [2,5].

The following two observations are obvious but useful.

Lemma 2 *Let $G = (V, E)$ be a graph with maximum degree Δ. Define f by $f(v) = \Delta - d(v)$, for all $v \in V$. Then G is Δ-regularizable if and only if $\overline{G}$ has an f-factor.*

Lemma 3 *Suppose $G = (V, E)$ is Δ-regularizable. Then $|V|$ is even or Δ is even.*

We use the following completion operation on graphs G with maximum degree Δ: as long as there are non-adjacent nodes of degree less than Δ, take two such nodes v and w, add $\{v, w\}$ to the set of edges and repeat. Any supergraph of G that can be obtained this way is called a *degree completion* of G.

Degree completions are not necessarily fully Δ-regular, even if the graph G is Δ-regularizable.

Lemma 4 *Let $G = (V, E)$ be a graph with maximum degree Δ.*
(i) If $H = (V, E')$ is a degree completion of G, then $\sum_{v \in V}(\Delta - d_H(v)) \leq \Delta^2$.
(ii) A degree completion of G can be computed in $O(\Delta|V|)$ time.

PROOF.

(i) Let H be a degree completion of G. Note that the vertices with degree at most $\Delta - 1$ in H form a clique of size at most Δ. This implies the bound on $\sum_{v \in V}(\Delta - d_H(v))$.

(ii) Assume that G is given in a normal adjacency list representation. In $O(\Delta|V|)$ time one can pass through the nodes of G, compute their degrees, and link the nodes of degree less than Δ in a doubly linked list L. As long as L is non-empty, repeat the following step. Pick the leading node v, delete it from L, go through its adjacency list and mark all the nodes that appear on it as its neighbors, and do the following by going through the consecutive nodes w of L one after the other *until the degree of v has become Δ or the end of the list is reached*:

(1) if w is marked, then skip (v and w are already connected).
(2) if w is not marked, then add $\{v, w\}$ to the set of edges:
  (a) add w to the adjacency list of v,
  (b) add v to the adjacency list of w,
  (c) increase the degree counters of v and w by 1, and
  (d) if the degree of w has become Δ, then delete it from L.

When this is done, undo the marking of the original neighbors of v and repeat unless the stop criterion is satisfied. As L gets shorter by at least one node in every step, the algorithm terminates in finitely many steps. The resulting graph H clearly is a degree completion of G, in adjacency list representation.

Note that after picking node v from the head of the list, the algorithm takes at most $O(\Delta)$ time to handle it: it can run into marked nodes w at most Δ times, and each time a non-marked node w is encountered the degree of v increases by 1 and thus this can happen at most Δ times as well. Undoing the marking to prepare for a next step takes another $O(\Delta)$ time. It follows that the running time of the algorithm is bounded by $O(\Delta|V|)$. □

3 Determining a Δ-regular supergraph of minimum order

In this section we show how to compute the minimum number of vertices that must be added to an input graph G to make it Δ-regular. We give two algorithms for the problem, both running in time polynomial in Δ and $|V|$.

We look for the smallest number of vertices that must be added to G so the resulting graph is Δ-regularizable. By Lemma 2, it follows that the following procedure computes the desired information. It has a graph $G = (V, E)$ of maximum degree Δ as input, and it outputs a Δ-regular graph H of minimum order that contains G as a subgraph.

Algorithm **A**:

(1) **while** $\overline{G}$ has no f-factor, with f the function defined by $f(v) = \Delta - d_G(v)$ for all $v \in V$
do
(a) Add a new isolated vertex to G.
(2) Let F be the set of edges in a f-factor of $\overline{G}$.
(3) Output $H = (V, E \cup F)$.

The algorithm immediately leads to the following result.

Theorem 5 *The problem of determining a Δ-regular supergraph $H = (V', E')$ of minimum order of a given graph $G = (V, E)$ of maximum degree Δ can be solved in $O(\Delta^{1.5}|V|^{2.5})$ time.*

PROOF.

Gabow [4] has shown that the problem of determining whether a graph G contains a f-factor, and computing one if it exists, can be solved in

$$O(\sqrt{\textstyle\sum_{v \in V} f(v)} \cdot |E|)$$

time. Using this test in step (1) of the algorithm, Theorem 1 shows that it is applied to graphs that have up to $\Delta + 2 = O(|V|)$ more vertices than the input graph G. Thus the loop in Algorithm **A** can be implemented to run in $O(\sqrt{\Delta|V|}|V|^2)$ time per iteration. As the number of iterations of the loop is at most $\Delta + 2$ (cf. Theorem 1), the time complexity of Algorithm **A** is bounded by $O(\Delta^{1.5}|V|^{2.5})$. □

Algorithm **A** needs $O(\Delta)$ runs of an f-factor algorithm, hence its time complexity is a factor of $O(\Delta)$ larger than that of the best f-factoring algorithm.

However, one can note that the graphs in the separate calls to the f-factor algorithm have great similarity, so it seems likely that an incremental construction could lower the runtime somewhat further.

When Δ is small, a better running time can be obtained. We need the following lemma that is of interest in its own right.

Lemma 6 *Let $G = (V, E)$ be a graph with maximum degree Δ. Suppose $\sum_{v \in V}(\Delta - d(v)) \geq 5\Delta^2$. Then G is Δ-regularizable if and only if $|V|$ is even or Δ is even.*

PROOF.

The 'only if' part follows from Lemma 3. To show the 'if' part, suppose that $|V|$ is even or Δ is even.

Let W be the set of vertices in G with degree less than Δ. Now build a Δ-regular supergraph $H = (V, E')$ of G as follows. As an invariant we maintain that (V, E') is a supergraph of G of maximum degree Δ. Start with $E' = E$.

First, we repeatedly add edges $\{v, w\}$ to E', with v and w non-adjacent vertices of degree less than Δ. The resulting graph H is a degree completion of G and thus has $\sum_{v \in V}(\Delta - d_H(v)) \leq \Delta^2$ (Lemma 4). Because $\sum_{v \in V}(\Delta - d(v)) \geq 5\Delta^2$ at the start, it means that the total degree deficiency in G has decreased by at least $4\Delta^2$. Thus at least $2\Delta^2$ new edges were added in the completion process.

Now, as long as there are at least two (adjacent) vertices of degree at most $\Delta - 1$, repeatedly apply the following step. Take two vertices v and w of degree at most $\Delta - 1$. Note that necessarily $\{v, w\} \in E'$, and that $|E' - E| \geq 2\Delta^2$. The number of edges that have at least one endpoint equal to or adjacent to v or w is at most $2\Delta^2 - 1$. Hence, there is an edge $\{x, y\} \in E' - E$, with x and y not equal to or adjacent to v or w. Now, take such an edge $\{x, y\}$, and replace it by the edges $\{v, x\}$, $\{w, y\}$, i.e., change E' to $E' - \{\{x, y\}\} \cup \{\{v, x\}, \{w, y\}\}$. Note that, by the choice of x, the edges $\{v, x\}$ and $\{w, y\}$ did not belong to E' before the operation, hence we increased the size of E' by one. It means that this step can be repeated again. The invariant that G is a subgraph of $H = (V, E')$ is maintained throughout.

After this step, we may still have one vertex v of degree less than Δ left, but all other vertices have degree Δ in the current graph H. As $|V|$ is even or Δ is even, we must have that $d_H(v) \leq \Delta - 2$. By a similar argument as above, there must be an edge $\{x, y\} \in E' - E$ with x and y both not equal or adjacent to v, and we can replace $\{x, y\}$ by $\{v, x\}$ and $\{v, y\}$. Again the invariant is maintained. We can repeat this step until the degree of v, and hence of all vertices in V, finally equals Δ.

This proves that G is Δ-regularizable. $\square$

The bound in Lemma 6 might be improved with respect to the constant factor but not asymptotically: there are graphs with maximum degree Δ and $\sum_{v \in V}(\Delta - d(v)) = \Theta(\Delta^2)$ that are not Δ-regularizable.

As an example, consider the graph G that is the disjoint union of a clique of $\Delta/2+1$ vertices, a clique of $\Delta+1$ vertices, and $\Delta/4$ cliques of Δ vertices. To Δ-regularize G, one must add $(\Delta/2)(\Delta/2+1)$ edges with exactly one endpoint in the first clique. But there are only $\Delta^2/4$ vertices in G that can be the endpoint of any one such edge, and each one of them can be endpoint of at most one such edge. Thus G is not Δ-regularizable.

Theorem 7 *The problem of determining a Δ-regular supergraph $H = (V', E')$ of minimum order of a given graph $G = (V, E)$ of maximum degree Δ can be solved in $O(\Delta^6 + \Delta|V|)$ time.*

PROOF.

First compute $f(G) = \sum_{v \in V}(\Delta - d(v))$.

If $f(G) < 5\Delta^2$, then compute the set W of vertices in G of degree less than Δ. For all $v \in W$, compute $f(v) = \Delta - d_G(v)$. Using these values of $f(v)$, we can ignore all vertices of degree Δ, and proceed as in Algorithm **A**: we call the f-factor algorithm $O(\Delta)$ times on a graph with $O(\Delta^2)$ vertices, and hence with $O(\Delta^4)$ edges, and with $\sum_{v \in W} f(v) = O(\Delta^2)$. Using the f-factoring algorithm of [4], this costs $O(\Delta^6)$ time.

If $f(G) \geq 5\Delta^2$, we know that G is Δ-regularizable. To find the corresponding supergraph, first add enough edges between non-adjacent vertices of degree less than Δ so as to obtain a supergraph H of G with $f(H)$ equal to $5\Delta^2+1$ or $5\Delta^2$. This can be done in a greedy manner as in the proof of Lemma 4 (ii), while keeping track of the decreasing value of $f(H) = \sum_{v \in V}(\Delta - d(v))$ at every step. Because a full degree completion brings the value of $f(H)$ under Δ^2 (and the addition of every edge decreases $f(H)$ by 2), this can certainly be done. As in Lemma 4 (ii) the graph H can be computed in $O(\Delta \cdot |V|)$ time.

Now we have a graph H with $f(H) = 5\Delta^2(+1)$. H still is Δ-regularizable, and has at most $5\Delta^2(+1)$ vertices of degree at most $\Delta - 1$. Running an f-factoring algorithm on the subgraph with the vertices in H of degree less than Δ gives the set of edges that can be added to make the graph Δ-regular: this costs $O(\Delta^5)$ time.

The total running time of the algorithm is thus bounded by $O(\Delta^6 + \Delta|V|)$. $\square$

4 An algorithmic proof of the Akiyama-Era-Harary theorem

In this section, we give an algorithmic proof of Theorem 1. In minor details the proof differs from the proof of Akiyama, Era, and Harary [1]. It shows that the construction can be carried out by an algorithm that runs in $O(\Delta \cdot |V|)$ time.

The algorithm consists of a number of steps. Suppose a graph $G = (V, E)$ of maximum degree Δ is given. In each step, we can add vertices and/or edges to the graph. The graph that develops is denoted by $H = (V', E')$, with $H = G$ at the start. We now describe the consecutive steps.

Step 1: Verify evenness

When Δ is odd and $|V|$ is odd, then add a new vertex of degree 0 to V'. This step ensures that $\Delta \cdot |V'|$ is even.

Step 2: Degree completion

While there are vertices v, $w \in V'$, with $d(v) < \Delta$, $d(w) < \Delta$, $v \neq w$, and $\{v, w\} \notin E'$, then add an edge $\{v, w\}$ to E'.

By Lemma 4 (ii) this step can be done in $O(\Delta \cdot |V|)$ time.

After this step, let W be the set of vertices of H that have degree less than Δ. If W is empty, then H is Δ-regular and we are done. If W is not empty, then by the argument in the proof of Lemma 4 (i) it follows that the nodes of W form a clique of size at most Δ in H.

Let $W = \{w_0, \ldots, w_{|W|-1}\}$.

Step 3: Add $\Delta + 1$ new vertices to H and give the vertices in W degree Δ

In this step, we add a set of $\Delta + 1$ new vertices $N = \{n_0, n_1, \ldots, n_\Delta\}$ to V' and we add as many edges between vertices w_i and vertices n_j as are needed to give all all vertices in W degree Δ and such that the vertices in N differ in degree by at most 1. This is easily implemented in $O(\Delta^2)$ time by 'filling' the nodes w_i one after the other and cyclically going through the vertices n_j to create the necessary edges.

After Step 3, all vertices in W will have degree Δ, and all vertices in N have 'almost the same' degree: there is an integer s, such that every vertex in N has either degree s or degree $s + 1$. One easily observes that $s + 1 \leq \Delta$. In addition, the vertices with degree $s+1$ appear consecutively in the given order of N.

Step 4: Give vertices in N degree Δ

Note that we have so far not added any edge between vertices in N. This is done in this step in such a way that all vertices in N (and hence all vertices in H) get degree Δ.

We distinguish a few cases which are all handled slightly different.

Case 1: $\Delta - s$ is even and there are no vertices of degree $s+1$

Let $\alpha = (\Delta - s)/2$. Now add edges $\{n_j, n_{(j+\beta) \bmod \Delta}\}$ to E' for all j, $0 \leq j \leq \Delta$, and all β, $1 \leq \beta \leq \alpha$. In words, viewing $n_0, \ldots, n_\Delta$ arranged along a cycle, we add an edge between each pair of vertices of distance at most α on the cycle. This results in a Δ-regular graph.

Case 2: $\Delta - s$ is even and there are vertices of degree $s+1$

Suppose w.l.o.g. that vertices $n_0, \ldots, n_{t-1}$ have degree $s+1$. Now, t is even: all vertices, except those in N have degree Δ, and $|V' - V| \cdot \Delta$ is even; so the sum of all degrees of the vertices in N is even. If Δ is even, then s is even, hence t is even. If Δ is odd, then $(\Delta + 1) \cdot s$ is even, hence t is even.

Now, we use the same construction as in Case 1, except that we do not add the edges $\{n_{2\gamma}, n_{2\gamma+1}\}$ with $0 \leq \gamma < t/2$. This gives a Δ-regular graph.

Case 3: $\Delta - s$ is odd

Again suppose vertices $n_0, \ldots, n_{t-1}$ have degree $s+1$. Let $\alpha = (\Delta - s + 1)/2$. Now, use the same construction as in Case 1, but do not add edges of the form $\{n_j, n_{(j+1) \bmod \Delta}\}$. This gives a graph in which every vertex has degree Δ, except the vertices $n_0, \ldots, n_{t-1}$ which have degree $\Delta - 1$. Again, case analysis shows that t must be even, and we can add the edges $\{n_{2\gamma}, n_{2\gamma+1}\}$, with $0 \leq \gamma < t/2$ to obtain a Δ-regular graph.

The construction described above results in a Δ-regular supergraph H of G and proves the following constructive variant of Theorem 1.

Theorem 8 *There is an algorithm that, given any graph $G = (V, E)$ with maximum degree at most Δ, determines a Δ-regular graph $H = (V', E')$ with $|V' - V| \leq \Delta + 1$ when Δ is even and $|V' - V| \leq \Delta + 2$ when Δ is odd, and that uses $O(\Delta|V|)$ time.*

Acknowledgement

We thank K.I. Aardal and T. Nishizeki for suggesting several useful references.

References

[1] Akiyama, J., Era, H., and Harary, F. Regular graphs containing a given graph. *Elem. Math. 83* (1983), 15–17.

[2] Anstee, R. P. An algorithmic proof of Tutte's f-factor theorem. *J. Algorithms 6* (1985), 112–131.

[3] Erdös, P., and Kelly, P. The minimal regular graph containing a given graph. *Amer. Math. Monthly 70* (1963), 1074–1075.

[4] Gabow, H. N. An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems. In *Proceedings of the 15th Annual Symposium on Theory of Computing* (1983), pp. 448–456.

[5] Gerards, A. M. H. Matching. In *Handbooks in Operations Research and Management Sciences, Volume 7, Network Models*, M. O. Ball et al., Ed. Elsevier Science, Amsterdam, 1995, ch. 3, pp. 135–224.

[6] König, D. *Theorie der endlichen und unendlichen Graphen.* Akademische Verlagsgesellschaft, Leipzig, 1936.