Discrete Applied Mathematics 157 (2009) 3593–3600

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Approximating the maximum 2- and 3-edge-colorable subgraph problems ☆

Adrian Kosowski *

Department of Algorithms and System Modeling, Gdańsk University of Technology, Narutowicza 11/12, 80952 Gdańsk, Poland

ARTICLE INFO

Article history:
Received 1 December 2007
Received in revised form 23 January 2009
Accepted 2 April 2009
Available online 6 May 2009

Keywords:
Edge coloring of graphs
k-edge-colorable subgraph problem
Approximation algorithm

ABSTRACT

For a fixed value of a parameter $k \geq 2$, the Maximum k-Edge-Colorable Subgraph Problem consists in finding k edge-disjoint matchings in a simple graph, with the goal of maximising the total number of edges used. The problem is known to be *APX*-hard for all k, but there exist polynomial time approximation algorithms with approximation ratios tending to 1 as k tends to infinity. Herein we propose improved approximation algorithms for the cases of $k = 2$ and $k = 3$, having approximation ratios of 5/6 and 4/5, respectively.

1. Introduction

The Maximum k-Edge-Colorable Subgraph Problem (k-ECS) is a well-studied variant of classical edge coloring in which the goal is to assign colors from the range $\{1, \ldots, k\}$ to as many edges of the input graph as possible, preserving the constraint that adjacent edges must receive different colors. Formally, the problem may be stated as follows.

Maximum k-Edge-Colorable Subgraph Problem (k-ECS)	
***Input*:**	A simple graph G.
***Solution*:**	A subgraph $SOL \subseteq G$ together with a legal k-edge-coloring of SOL.
***Goal*:**	Maximise the number of edges of SOL.

Edge coloring techniques are used to model many real-world problems of resource allocation, such as scheduling of tasks requiring the cooperation of two processors, file transfer operations, and assignment of channels for satellite communication [7,13,16]. The optimisation criterion of k-ECS describes situations in which there exists a fixed limit of k on the number of available resources (e.g. time slots or communication channels) and the quality of the solution is expressed in terms of the proportion of input which is handled within these constraints. Some specific cases of k-ECS also have other applications; for example, the 2-ECS problem is closely related to the so-called 2-Factor Packing Problem which is used to model certain geometrical guarding problems in grids [1].

☆ The research was partially funded by the State Committee for Scientific Research (Poland) Grant 4 T11C 047 25.
* Tel.: +48 583471956; fax: +48 583471766.
E-mail address: adrian@kaims.pl.

0166-218X/$ – see front matter
doi:10.1016/j.dam.2009.04.002

1.1. Notation and preliminaries

We will assume that all considered graphs are simple (without multiple edges or loops). For a given graph H, let $V(H)$ denote its vertex set and $E(H)$ its edge set; for the input instance G we simply use the symbols $V = V(G)$ and $E = E(G)$. The number of neighbours of a vertex $v \in V(H)$ is the *degree* $\deg_H v$ of the vertex; the maximum vertex degree in graph H is denoted $\Delta(H)$. Graph H is called *subcubic* if $\Delta(H) \leq 3$. Unless otherwise stated, a *maximum* element in a family of graphs is a graph from this family having the maximum number of edges. For a set $S \subseteq V(H)$, the maximum subgraph of H having vertex set S is said to be *induced* by S and is denoted $H[S]$.

Let $f, g : V \to \mathbb{N}$ be integer functions on the vertex set of G fulfilling the relation $f(v) \leq g(v)$ for all $v \in V$. A subgraph $F \subseteq G$ is called an $[f, g]$*-factor* in G if $V(F) = V$ and $f(v) \leq \deg_F v \leq g(v)$ for all $v \in V$. The term k*-matching*, for some constant $k \in \mathbb{N}$, is equivalent to a $[0, k]$-factor, and a 1-matching is simply known as a matching. We note that the problem of finding a maximum $[f, g]$-factor in a graph can be solved in polynomial time (cf. [20] for a survey of approaches).

When considering the k-ECS problem, the symbol OPT is used to denote some arbitrarily chosen (but fixed) k-colorable subgraph of G which constitutes an optimal solution to k-ECS for G, and SOL denotes some k-colorable subgraph which may be returned as a solution by the currently considered algorithm. An algorithm is said to be an α*-approximation algorithm* ($\alpha \leq 1$) if $\frac{|E(SOL)|}{|E(OPT)|} \geq \alpha$ for all possible input instances and all possible executions of the algorithm.

1.2. State-of-the-art results

The k-ECS problem is known to be *APX*-hard for all $k \geq 2$ [5,11,14] (cf. [7] for a detailed discussion). In view of this, several simple approximation algorithms, working for all k, have recently been proposed; two of these are briefly outlined below.

- The *greedy strategy* (cf. e.g. [7,10]) runs in k steps. In the ith step, a maximum matching M_i is selected in graph G and its edges are colored with color i, added to the solution SOL, and removed from graph G. The approximation ratio of this method is $1 - (1 - k^{-1})^k$, and this result holds even when G is allowed to be a multigraph.
- The k*-matching-based algorithm* [3,7] first finds a maximum k-matching F in the input graph G. Subgraph F is then edge-colored with at most $\Delta(F) + 1 = k + 1$ colors using Vizing's algorithm [15], and all the edges of F, except for those which in the coloring of F received the least often used color, form the solution SOL. The algorithm achieves an approximation ratio of $\frac{k}{k+1}$; note that this ratio tends to 1 as k tends to infinity.

In the special case of $k = 2$, the 2-ECS problem can be equivalently defined as the problem of finding a maximum subgraph of G whose connected components are isolated vertices, paths, and cycles of even length. 2-ECS is known to be *APX*-hard even for the restricted class of 2-connected subcubic graphs, and it remains *NP*-hard even for 2-connected subcubic planar graphs [1]. In general, a $\frac{3}{4}$-approximate solution is achieved by the greedy strategy (putting $k = 2$). However, a slight modification of the k-matching-based algorithm is a $\frac{4}{5}$-approximation, and this ratio has recently been improved to $\frac{468}{575} \approx 0.814$ by Chen and Tanahashi [3] using an extension of the same approach. An algorithm with a ratio of $\frac{28\Delta(G)-12}{35\Delta(G)-21}$, providing a further improvement for graphs of maximum degree $\Delta(G) \leq 10$ only, has also been put forward [1].

Several other variants of the k-ECS problem have been studied as well. Algorithms with an approximation ratio which tends to 1 as k tends to infinity have also been proposed for the case when G is allowed to be a multigraph [7,19]. Approximation results for k-ECS in the on-line model of computation were presented in [6].

Contribution and outline of the paper. In Section 2 we propose a new polynomial time approximation algorithm for 2-ECS with an approximation ratio of $\frac{5}{6}$, thus improving the earlier $\frac{468}{575}$-approximation approach from [3]. In Section 3 we propose a simple $\frac{4}{5}$-approximation algorithm for 3-ECS; to the best of our knowledge, this is the first result better than the trivial $\frac{3}{4}$-approximation approach to this problem. The closing section contains some remarks on possible extensions of the applied techniques.

2. Approximation algorithm for 2-ECS

The proposed approach to 2-ECS is stated in the form of Algorithm 1, and an illustration of its execution is presented in Fig. 1. In the following subsections we proceed to show that Algorithm 1 achieves an approximation ratio of $\frac{p}{p+1}$, and that all the steps of the algorithm can be implemented to run in polynomial time when we set the value of parameter $p = 5$.

2.1. Analysis of approximation ratio

In Step 1 of the algorithm we compute a 2-matching F such that all of its connected components are paths, even cycles, or odd cycles of length at least p. Observe that the additional assumption given in Step 1 can easily be fulfilled for any graph G. Initially, we find any maximum 2-matching F, and then perform a local modification of F for each pair of vertices $\{v, w\}$ which violates the assumption, namely, edge $\{v, w\}$ is added to F and one of the two edges incident to v in the odd cycle is removed from F. Since each such modification decreases the number of odd cycles in F, a suitable 2-matching F is obtained after a finite number of iterations.

Algorithm 1 Approximation algorithm for 2-ECS in input graph G

1. Find a maximum 2-matching $F \subseteq G$ having no odd cycles of length less than p (parameter p should be chosen so that a polynomial-time subroutine exists for this step). Select F in such a way that for any two vertices $v, w \in V$, such that v belongs to an odd cycle of F and $\deg_F w < 2$, we have $\{v, w\} \notin E(G)$.
2. Find a matching $R \subseteq G$ which fulfills the following conditions:
   - all edges of R have either exactly one end-vertex belonging to some odd cycle of F, or two end-vertices belonging to two different odd cycles of F.
   - R is incident to the maximum possible number of odd cycles of F.

   Subject to the above conditions, matching R should be minimal (in terms of inclusion).
3. Define subgraph $H \subseteq G$ as $H = (V, E(F) \cup E(R))$.
4. Return a maximum 2-edge-colorable subgraph of H as the solution SOL.

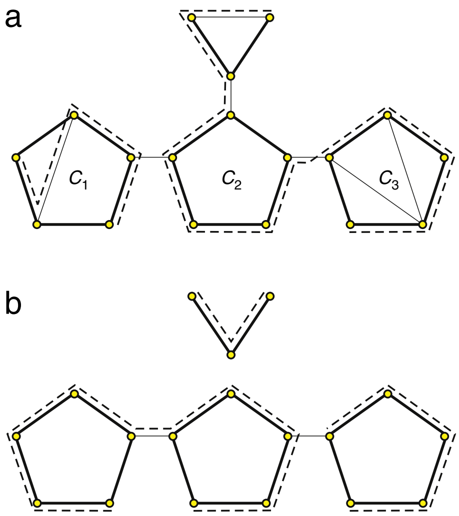

Fig. 1. Exemplary execution of Algorithm 1 for $p = 5$: (a) input graph G (all edges), maximum 2-matching F (bold edges, $|E(F)| = 17$), an optimal solution OPT to the 2-ECS problem for G (dashed edges, $|E(OPT)| = 16$); (b) maximum 2-matching F (bold edges), matching R (thin edges), subgraph H (all edges; $H = (V, E(F) \cup E(R))$ in this case), final solution SOL (dashed edges, $|E(SOL)| = 15$).

For an odd cycle component $C \subseteq F$ we define the set $N(C)$ of *outer-neighbouring edges of cycle* C as the set of edges of G which have exactly one end-vertex in the vertex set of C. Let Γ denote the set of odd cycle components of F. For a subgraph $G' \subseteq G$, we denote by $\Gamma(G') \subseteq \Gamma$ the set of all odd cycles $C \in \Gamma$ such that G' is incident to C, i.e. $E(G') \cap N(C) \neq \emptyset$; the set of all the remaining odd cycles is denoted $\overline{\Gamma}(G') = \Gamma \setminus \Gamma(G')$. In the example in Fig. 1(a) we have $\Gamma = \{C_1, C_2, C_3\}$, $\Gamma(OPT) = \{C_2, C_3\}$, $\overline{\Gamma}(OPT) = \{C_1\}$.

Lemma 2.1. *The size of an optimal solution to 2-ECS in G can be bounded as:*

$$|E(OPT)| \leq |E(F)| - |\overline{\Gamma}(OPT)|. \tag{1}$$

Proof. When $\overline{\Gamma}(OPT) = \emptyset$ the claim is obviously true since OPT is a 2-matching in G and F is a maximum 2-matching in G. Now, consider an arbitrary cycle $C \in \overline{\Gamma}(OPT)$; let $|V(C)| = 2a + 1$. Since OPT does not contain any edges from $N(C)$, and a 2-edge-colorable graph on $2a + 1$ vertices has at most $2a$ edges, it follows that $|E(OPT)| = |E(OPT[V \setminus V(C)])| + |E(OPT[V(C)])| \leq |E(OPT[V \setminus V(C)])| + 2a = |E(OPT[V \setminus V(C)])| + |V(C)| - 1$. By repeating the same procedure for all cycles $C \in \overline{\Gamma}(OPT)$, we eventually obtain:

$$|E(OPT)| \leq |E(OPT[V'])| + \left(\sum_{C \in \overline{\Gamma}(OPT)} |V(C)| \right) - |\overline{\Gamma}(OPT)|, \tag{2}$$

where $V' = V \setminus \bigcup_{C \in \overline{\Gamma}(OPT)} V(C)$. For factor F we may write $|E(F[C])| = |V(C)|$ for all $C \in \overline{\Gamma}(OPT)$, and consequently:

$$|E(F)| = |E(F[V'])| + \sum_{C \in \overline{\Gamma}(OPT)} |V(C)|. \tag{3}$$

Observe that $OPT\,[V']$ is a 2-matching in $G[V']$, and $F[V']$ is a maximum 2-matching in $G[V']$. Hence $|E(OPT\,[V'])| \leq |E(F[V'])|$ and the claim follows directly from this bound and relations (2) and (3). □

Lemma 2.2. *For the matching R constructed in Step 2 of Algorithm 1 we have $|\Gamma(R)| \geq \frac{1}{2}|\Gamma(OPT\,)|$.*

Proof. Consider the subset A of edges of OPT which are outer-neighbouring for at least one odd cycle, $A = OPT \cap \bigcup_{C \in \Gamma} N(C)$. First, observe that $\Gamma(A) = \Gamma(OPT\,)$. Since A is 2-edge colorable, it can be decomposed into two matchings A_1 and A_2. Clearly, $|\Gamma(A_1)| + |\Gamma(A_2)| \geq |\Gamma(A)| = |\Gamma(OPT\,)|$, so without loss of generality we can assume that $|\Gamma(A_1)| \geq \frac{1}{2}|\Gamma(OPT\,)|$. Moreover, all edges of A_1 either have exactly one end-vertex in an odd cycle component of F, or both end-vertices in different odd cycle components of F. Thus by the definition and the maximality of matching R, we obtain $|\Gamma(R)| \geq |\Gamma(A_1)| \geq \frac{1}{2}|\Gamma(OPT\,)|$. □

When considering Steps 3 and 4 of the algorithm it is convenient to introduce a contraction operation with respect to F, defined as follows: for a subgraph $G' \subseteq G$, its *contraction graph* G'_F has a vertex set corresponding to the set of connected components of F, and two vertices C_1, C_2 are connected by an edge if for some two vertices $v_1 \in V(C_1)$ and $v_2 \in V(C_2)$ of G' there exists an edge $\{v_1, v_2\} \in E(G')$. Vertices of G'_F corresponding to odd cycle components of F are called *odd cycle vertices*. In this notation, Lemma 2.2 may be restated as follows.

Corollary 2.1. *The contraction graph $H_F = (V, E(F) \cup E(R))_F$ has $|\Gamma|$ odd cycle vertices, and exactly $|\Gamma(R)| \geq \frac{1}{2}|\Gamma(OPT\,)|$ of these vertices are not isolated vertices.*

By the minimality of matching R, we conclude that removing any edge from H_F would violate the property expressed by the above corollary, which means that all components of H_F with more than one vertex have to be stars, of order at least 2, in which all non-central vertices are odd cycle vertices (for the star of order 2 the center is chosen so as to meet this condition). Therefore graph H may be represented in the form of a union of connected components, $H = \bigcup_{C_i \in \mathcal{C}} C_i \cup \bigcup_{S_i \in \mathcal{S}} S_i$, where:

- each connected component belonging to set $\mathcal{C}$ is an odd cycle component of F,
- each connected component S belonging to set $\mathcal{S}$ must be a support graph, where a *support graph* is by definition one of the following graphs:
  - an even cycle or path which appears as a subgraph of F ($\Delta(S) \leq 2$),
  - a star-of-cycles graph formed by connecting a connected subgraph of F (known as the *center* of S) with edges of R to some non-zero number of odd cycle components of F ($\Delta(S) = 3$).

In the example (Fig. 1(b)) both of the connected components of H are support graphs from $\mathcal{S}$, and $\mathcal{C} = \emptyset$.

In the contraction graph of H, odd cycle vertices are isolated if and only if they correspond to a component in set $\mathcal{C}$. For any component $S \in \mathcal{C} \cup \mathcal{S}$, denote by $o(S)$ the number of odd cycle vertices contained in S. Taking into account Corollary 2.1 we immediately obtain:

$$\sum_{S_i \in \mathcal{S}} o(S_i) = |\Gamma(R)| \geq \frac{1}{2}|\Gamma(OPT\,)|, \tag{4}$$

$$|\mathcal{C}| = \sum_{C_i \in \mathcal{C}} o(C_i) \leq |\Gamma| - \frac{1}{2}|\Gamma(OPT\,)| = |\overline{\Gamma}(OPT\,)| + \frac{1}{2}|\Gamma(OPT\,)|, \tag{5}$$

where we used the obvious fact that $o(C) = 1$ for all $C \in \mathcal{C}$. For any component $S \in \mathcal{C} \cup \mathcal{S}$, denote by $f(S)$ the number of edges of F contained in S; we have:

$$f(C) \geq p, \quad \text{for all } C \in \mathcal{C}. \tag{6}$$

Next, let $s(S)$ be the size of a maximum 2-edge-colorable subgraph of S. It is clear that for any $C \in \mathcal{C}$ we can write:

$$s(C) = f(C) - 1. \tag{7}$$

A similar expression now needs to be derived for components $S \in \mathcal{S}$, taking into account their simple structural properties as support graphs.

Lemma 2.3. *For any support graph S we have:*

$$s(S) \geq \frac{p \cdot f(S) + o(S)}{p + 1}. \tag{8}$$

Proof. To simplify notation, we will use the plain symbols f, o, and s when referring to graph S. Notice that the sought condition may be equivalently rewritten as $(p + 1)(f - s) \leq f - o$; we will concentrate on proving this version by induction with respect to o. The claim clearly holds when S contains no odd cycles, since S is then an even cycle or path of F ($o = 0, f = s$). Now, let $S_C \subseteq S$ be the connected subgraph of F forming the center of S, let $l = |E(S_C)|$, and let $t = |E(S[V(S) \setminus V(S_C)])| = f - l$ be the total number of edges in all odd cycles attached to the center of S. We need to consider two cases.

First, suppose that no two edges of R which belong to S have end-vertices which are neighbours in S_C. Depending on whether S_C is an even cycle, a path, or an odd cycle, we have the following possibilities.

1. S_C is an even cycle. Then clearly $l \geq 2o$. We can define a 2-edge-colorable subgraph in S by taking all edges of F belonging to S and discarding exactly one edge belonging to each odd cycle, hence $s \geq f - o$. We may therefore write:
$$(p+1)(f-s) \leq (p+1)o \leq p \cdot o + l - o \leq t + l - o = f - o,$$
which completes the proof in this case.
2. S_C is a path. Recall that by the assumption made in Step 1 of the algorithm, no edge of G can connect an odd cycle of F with an end-vertex of S_C, hence as before we have $l \geq 2o$. All the relations follow as in the previously considered case.
3. S_C is an odd cycle. Then $l \geq 2o - 1$ (recall that S_C is also counted when calculating o). We can define a 2-edge-colorable subgraph in S by taking all edges of F belonging to S, adding one edge connecting S_C with an arbitrarily chosen odd cycle in S, and then appropriately discarding exactly one edge from each odd cycle, hence $s \geq f + 1 - o$. We therefore once again obtain the sought inequality:
$$(p+1)(f-s) \leq (p+1)(o-1) \leq p(o-1) + l - o \leq t + l - o = f - o.$$

Next, suppose that there exists an edge $e \in E(S_C)$ such that some two edges $e_1 \in E(R) \cap E(S)$ and $e_2 \in E(R) \cap E(S)$ are incident to different end-vertices of e. Let $C_1, C_2 \subseteq S$ be the odd cycle components of F attached to the other end-vertices of e_1 and e_2, respectively. Consider the graph $S' = S[V(S) \setminus (V(C_1) \cup V(C_2))]$ formed by removing the edges e, e_1, e_2, and the cycles C_1 and C_2 from S. Depending on whether S_C was a cycle or a path, S' is either a support graph or a disjoint union of two support graphs. Thus, denoting $f' = f(S')$, $o' = o(S')$, and $s' = s(S')$, by the induction hypothesis we immediately have $(p+1)(f'-s') \leq f' - o'$. Comparing the parameters of S' with S we obtain relations $f' = f - 1 - |E(C_1)| - |E(C_2)|$, $o' \geq o - 3$. Moreover, $s \geq s' + |E(C_1)| + |E(C_2)|$, since any 2-edge-colorable subgraph of S' can be extended to a 2-edge-colorable subgraph of S by augmenting it with edges e_1, e_2, all but one edges from the cycle C_1 and all but one edges from the cycle C_2. We therefore have:
$$\begin{aligned}(p+1)(f-s) &\leq (p+1)(f' + 1 + |E(C_1)| + |E(C_2)| - (s' + |E(C_1)| + |E(C_2)|)) \\ &= (p+1)(f'-s') + p + 1 \leq f' - o' + p + 1 \\ &\leq f - 1 - |E(C_1)| - |E(C_2)| - (o-3) + p + 1 \leq f - o - (p-3) \leq f - o,\end{aligned}$$
which completes the proof. □

Taking into account relations (1) and (4)–(8) we finally obtain the sought bound on the approximation ratio:
$$\begin{aligned}\frac{|E(SOL)|}{|E(OPT)|} &\geq^{(1)} \frac{|E(SOL)|}{|E(F)| - |\overline{\Gamma}(OPT)|} = \frac{\sum_{S_i \in \mathcal{S}} s(S_i) + \sum_{C_i \in \mathcal{C}} s(C_i)}{\sum_{S_i \in \mathcal{S}} f(S_i) + \sum_{C_i \in \mathcal{C}} f(C_i) - |\overline{\Gamma}(OPT)|} \\ &\geq^{(7) \text{ and } (8)} \frac{\frac{p \sum_{S_i \in \mathcal{S}} f(S_i) + \sum_{S_i \in \mathcal{S}} o(S_i)}{p+1} + \sum_{C_i \in \mathcal{C}} f(C_i) - |\mathcal{C}|}{\sum_{S_i \in \mathcal{S}} f(S_i) + \sum_{C_i \in \mathcal{C}} f(C_i) - |\overline{\Gamma}(OPT)|} \geq^{(6)} \frac{\frac{p \sum_{S_i \in \mathcal{S}} f(S_i) + \sum_{S_i \in \mathcal{S}} o(S_i)}{p+1} + |\mathcal{C}|p - |\mathcal{C}|}{\sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}|p - |\overline{\Gamma}(OPT)|} \\ &\geq^{(4)} \frac{\frac{p \sum_{S_i \in \mathcal{S}} f(S_i) + \frac{1}{2}|\Gamma(OPT)|}{p+1} + |\mathcal{C}|p - |\mathcal{C}|}{\sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}|p - |\overline{\Gamma}(OPT)|} \geq^{(5)} \frac{\frac{p \sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}| - |\overline{\Gamma}(OPT)|}{p+1} + |\mathcal{C}|(p-1)}{\sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}|p - |\overline{\Gamma}(OPT)|} \\ &= \frac{p}{p+1} + |\overline{\Gamma}(OPT)| \cdot \frac{p-1}{p+1} \cdot \frac{1}{\sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}|p - |\overline{\Gamma}(OPT)|} \geq \frac{p}{p+1},\end{aligned}$$
where the last bound holds because $\sum_{S_i \in \mathcal{S}} f(S_i) + |\mathcal{C}|p > |\Gamma| \geq |\overline{\Gamma}(OPT)|$.

Proposition 2.1. *Algorithm* 1 *finds a* $\frac{p}{p+1}$*-approximate solution to the* 2-ECS *problem.*

2.2. Algorithm complexity

Proposition 2.2. *Algorithm* 1 *with parameter* $p = 5$ *can be implemented with polynomial running time.*

Proof. By applying the algorithm of Hartvigsen [9] as a subroutine, we can in polynomial time find a maximum 2-matching which has no cycles of length 3. This clearly allows us to perform Step 1 of the algorithm with parameter $p = 5$ also in polynomial time.

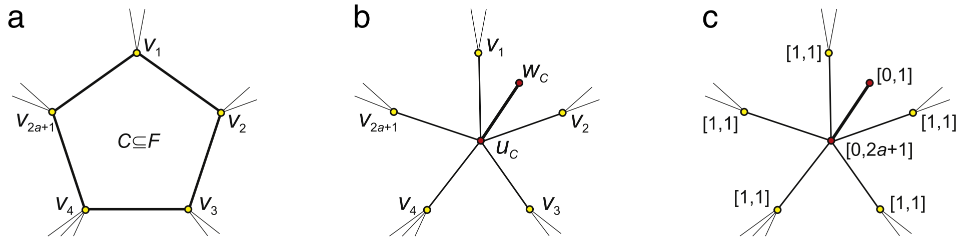

Fig. 2. Example of construction of auxiliary graph G': (a) odd cycle $C \subseteq F$ on $2a+1 = 5$ vertices, (b) local replacement of C in G', (c) values of vertex functions f and g for the $[f, g]$-factor.

Step 2 of the algorithm can be implemented by solving the maximum weighted $[f, g]$-factor problem in a graph $G' = (V', E')$, where sets V' and E', functions $f, g : V' \to \mathbb{N}$, and edge weights $c : E' \to \mathbb{N}$ are chosen as follows. First, we put $V' = V$ and assign to E' the set of all edges from which matching R is to be selected by the algorithm, i.e., the set of all edges of G having either exactly one end-vertex belonging to an odd cycle of F, or two end-vertices belonging to two different odd cycles of F. For all $v \in V$ we put $f(v) = g(v) = 1$. Next, for each odd cycle $C \subseteq F$, $V(C) = \{v_1, \dots, v_{2a+1}\}$, we add to V' two new vertices u_C and w_C, and to E' edge $e_C = \{u_C, w_C\}$ (known as the *counting edge*) and edges $\{v_i, u_C\}$ for all $1 \le i \le 2a+1$ (known as *star edges*), see Fig. 2 for an illustration. For vertex u_C we put $f(u_C) = 0$ and $g(u_C) = 2a+1$, while for vertex w_C we put $f(w_C) = 0$ and $g(w_C) = 1$. Finally, weights are assigned to edges from E' as follows: $c(e_C) = 1$ for counting edges, and $c(e) = 0$ for all other edges.

Observe that any matching within $G'[V]$ can be extended to an $[f, g]$-factor on G' simply by adding certain star edges and counting edges to the matching; conversely, any $[f, g]$-factor on G' restricted to vertex set V is a matching in $G'[V]$. Now, let R' be a maximum weight $[f, g]$-factor in G' (note that some $[f, g]$-factor in G' always exists since the set of all star edges in the graph constitutes a trivial solution to the problem). Since graph G' has $O(|V(G)|)$ vertices, the maximum weighted factor computed in this step can be found in polynomial time [17]. It is easy to see that for any odd cycle component $C \subseteq F$, R' contains the counting edge e_C if and only if R' contains at least one edge with one end-vertex in $V(C)$ and the other end-vertex in some other component of F. The weight of R' is thus equal to the number of odd cycle components of F incident to $R'[V]$. By the maximality of the weight of R', we have obtained that $R'[V]$ is a matching in $G'[V]$ incident to the maximum possible number of odd cycle components of F. So, Step 2 is complete when we choose R as any subset of $R'[V]$, minimal in terms of inclusion, which is adjacent to the same number of odd cycle components of F as $R'[V]$.

Steps 3 and 4 of the algorithm can be performed in linear time, taking into account the discussion from Section 2.1. In Step 4 we can easily find an optimal 2-edge-colorable subgraph of H taking advantage of the special structure of the connected components of H which are paths, cycles, or stars-of-cycles. A 2-edge-colorable subgraph in H, having a sufficient number of edges to guarantee the overall $\frac{p}{p+1}$-approximation ratio of Algorithm 1, can also be obtained simply by applying the procedure implied by the proof of Lemma 2.3. □

Combining Propositions 2.1 and 2.2 we obtain the final result of this section.

Theorem 2.1. *The 2-ECS problem admits a polynomial-time $\frac{5}{6}$-approximation algorithm.*

3. Approximation algorithm for 3-ECS

A simple $\frac{4}{5}$-approximation approach to the 3-ECS problem is presented in the form of Algorithm 2. In Step 1 we compute a maximum 3-matching $F \subseteq G$, which provides an upper bound on the size of the optimum solution to the problem, $|E(F)| \ge |E(OPT)|$. We now show that the sought solution SOL to 3-ECS can simply be chosen as an appropriate subgraph of F.

Proposition 3.1. *For any subcubic graph F there exists a 3-edge-colorable subgraph $SOL \subseteq F$ such that $|E(SOL)| \ge \frac{4}{5}|E(F)|$.*

Proof. If F is triangle-free, then the claim holds since by a result of Bondy and Locke [2] any triangle-free subcubic graph has a bipartite subgraph on a $\frac{4}{5}$ part of its edges, and a subcubic bipartite graph is 3-edge-colorable. When F is not triangle-free, consider the slight modification of this approach applied in the construction of SOL in Steps 2-7 of Algorithm 2. We can write $|E(F)| = d + t + |E(F'')|$, where d and t denote the number of edges of F contained in the removed diamonds and contracted triangles in Steps 2 and 3, respectively. Then the size of the 3-edge-colorable subgraph SOL is equal to the sum of the size of subgraph $|E(B)| = |E(B'')| \ge \frac{4}{5}|E(F'')|$, the number t of edges in the triangles, and the number $\frac{4}{5}d$ of edges in the four-vertex cycles of the diamonds; clearly, $|E(SOL)| \ge \frac{4}{5}|E(F)|$. □

Algorithm 2 Approximation algorithm for 3-ECS in input graph G

1. Find a maximum 3-matching $F \subseteq G$.
2. Obtain diamond-free subcubic graph F' from F by iteratively removing all subgraphs isomorphic to diamonds (Fig. 3(a)) from F.
3. Obtain subcubic graph F'' from F' by iteratively contracting all triangles in F' into single vertices (Fig. 3(b)).
4. Find a bipartite subgraph $B'' \subseteq F''$ such that $|E(B'')| \geq \frac{4}{5}|E(F'')|$, using the algorithm of Bondy and Locke [2].
5. Edge color B'' using 3 colors.
6. Map the edges of B'' into the edges of the corresponding subgraph B of F, retaining a correct 3-edge-coloring of B.
7. Extend the 3-edge-coloring of B to all triangles of F which were contracted in Step 3, and to the four-vertex cycles of all the diamonds removed in Step 2 (relevant cases are shown in Fig. 3(c) and (d)). Return the final edge-colored subgraph as the solution SOL.

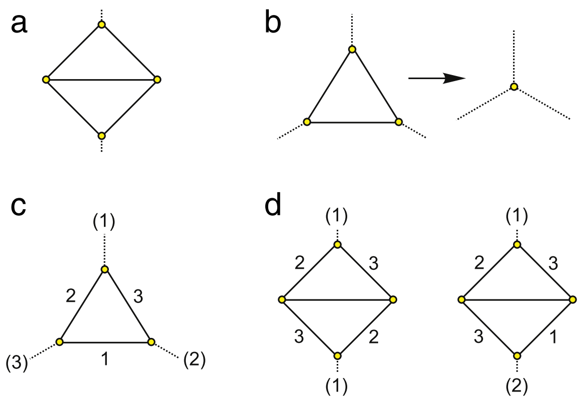

Fig. 3. Handling of triangles in a subcubic graph: (a) a diamond subgraph, (b) contraction of a triangle, (c) extending the coloring to a de-contracted triangle, (d) extending the coloring to the four-vertex cycle of a re-inserted diamond. Dashed edges may but need not appear in the graph.

Table 1
Best known approximation ratios of algorithms for k-ECS.

k-ECS	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k \to \infty$
ratio	5/6 Theorem 2.1	4/5 Theorem 3.1	4/5	5/6	6/7	$\to 1$

The running time of Algorithm 2 is determined by Step 1, which can be solved in $O(|V|^{1/2}|E|^2)$ time using an algorithm for maximum k-matching where k is constant [8]. All the remaining operations are restricted to subcubic graphs and require $O(|V|^2)$ time; in particular Step 4 makes use of a subroutine implied by the proof of Bondy and Locke [2], while Step 5 can be solved by a linear-time algorithm for edge-coloring bipartite graphs of bounded degree [4]. We can therefore write the following theorem.

Theorem 3.1. *The 3-ECS problem admits an $O(|V|^{1/2}|E|^2)$-time $\frac{4}{5}$-approximation algorithm.*

4. Final remarks

A comparison of the best known approximation ratios achieved by algorithms for k-ECS for different values of k is given in Table 1. The values for $k = 2$ and $k = 3$ correspond to the results described in the paper, whereas for $k \geq 4$ the ratios are obtained by applying the simple k-matching-based $\frac{k}{k+1}$-approximation algorithm outlined in the introduction.

The k-ECS problem appears to have a fundamentally different nature depending on the parity of k. For odd values of k it may be possible to improve the approximation ratio simply by carefully edge-coloring the maximum k-matching, trying to use color $k + 1$ as seldom as possible. For even values of k such an improvement probably cannot be applied in general (since then only a $\frac{k}{k+1}$ proportion of the edges of the clique K_{k+1} can be edge-colored with k colors). It is interesting to ask whether a generalisation of the techniques used herein for the cases of $k = 2$ and $k = 3$ could be applied for other even and odd values of k, respectively.

Considerations concerning 3-ECS in Section 3 lead naturally to the following extremal problem which is of some interest in its own right: *what is the proportion ϱ of edges of a subcubic graph which can always be colored with 3 colors*? This question was probably first formulated in [12], where a bound useful for graphs of small dominating number was provided. In the general case, the properties of bipartite subgraphs of triangle-free subcubic graphs discussed in Section 3 lead to the lower bound $\varrho \geq \frac{4}{5}$. In follow-up work on the subject available at the time of preparation of these proofs, by substantially extending the techniques proposed in Section 3, Rizzi [18] has established that $\varrho = \frac{6}{7}$.

Acknowledgements

The author would like to express his gratitude to the anonymous referees for their careful reading of the paper and numerous helpful comments.

References

[1] A. Kosowski, M. Małafiejski, P. Żyliński, Packing [1, Δ]-factors in graphs of small degree, Journal of Combinatorial Optimization 14 (2007) 63–86.
[2] J.A. Bondy, S.C. Locke, Largest bipartite subgraphs in triangle-free graphs with maximum degree three, Journal of Graph Theory 10 (1986) 477–504.
[3] Z.-Z. Chen, R. Tanahashi, L. Wang, An improved approximation algorithm for maximum edge 2-coloring in simple graphs, Journal of Discrete Algorithms 6 (2008) 205–215.
[4] R.J. Cole, Two problems in graph theory, Ph.D. Thesis, Cornell University, 1982.
[5] G. Cornuejols, W.R. Pulleyblank, A matching problem with side conditions, Discrete Mathematics 29 (1980) 135–159.
[6] L.M. Favrholdt, M.N. Nielsen, On-line edge-coloring with a fixed number of colors, Algorithmica 35 (2003) 176–191.
[7] U. Feige, E. Ofek, U. Wieder, Approximating maximum edge coloring in multigraphs, Lecture Notes in Computer Science 2462 (2002) 108–121.
[8] H.N. Gabow, An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems, in: Proc. 15th ACM Symposium on Theory of Computing, 1983, pp. 448–456.
[9] D. Hartvigsen, Extensions of matching theory, Ph.D. Thesis, Carnegie-Mellon University, 1984.
[10] D.S. Hochbaum, Approximation Algorithms for NP-hard Problems, PWS Publishing, 1997.
[11] I. Holyer, The NP-completeness of edge-coloring, SIAM Journal on Computing 10 (1981) 718–720.
[12] M. Juvan, B. Mohar, R. Škrekovski, On list edge-colorings of subcubic graphs, Discrete Mathematics 187 (1998) 137–149.
[13] M. Kubale, Introduction to computational complexity and algorithmic graph coloring, GTN, Gdańsk, Poland, 1998.
[14] D. Leven, Z. Galil, NP-completeness of finding the chromatic index of regular graphs, Journal of Algorithms 4 (1983) 35–44.
[15] J. Misra, D. Gries, A constructive proof of Vizing's theorem, Information Processing Letters 41 (1992) 131–133.
[16] S. Nakano, X. Zhou, T. Nishizeki, Edge-coloring algorithms, Lecture Notes in Computer Science 1000 (1995) 172–183.
[17] W.R. Pulleyblank, Faces of matching polyhedra, Ph.D. Thesis, University of Waterloo, 1973.
[18] R. Rizzi, Approximating the maximum 3-edge-colorable subgraph problem, Discrete Mathematics, 2009, in press (doi:10.1016/j.disc.2008.11.017).
[19] P. Sanders, D. Steurer, An asymptotic approximation scheme for multigraph edge coloring, in: Proc. 16th ACM-SIAM Symposium on Discrete Algorithms, 2005, pp. 897–906.
[20] A. Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, Springer, 2003.