Approximation Algorithms for B_1-EPG Graphs

Dror Epstein[1,2], Martin Charles Golumbic[1,2], and Gila Morgenstern[2]

[1] Department of Computer Science, University of Haifa
[2] Caesarea Rothschild Institute (CRI), University of Haifa

Abstract. The edge intersection graphs of paths on a grid (or EPG graphs) are graphs whose vertices can be represented as simple paths on a rectangular grid such that two vertices are adjacent if and only if the corresponding paths share at least one edge of the grid. We consider the case of single-bend paths, namely, the class known as B_1-EPG graphs. The motivation for studying these graphs comes from the context of circuit layout problems. It is known that recognizing B_1-EPG graphs is NP-complete, nevertheless, optimization problems when given a set of paths in the grid are of considerable practical interest.

In this paper, we show that the coloring problem and the maximum independent set problem are both NP-complete for B_1-EPG graphs, even when the EPG representation is given. We then provide efficient 4-approximation algorithms for both of these problems, assuming the EPG representation is given. We conclude by noting that the maximum clique problem can be optimally solved in polynomial time for B_1-EPG graphs, even when the EPG representation is not given.

1 Introduction

Edge intersection graphs of paths on a grid (or for short EPG graphs) were first introduced by Golumbic, Lipshteyn and Stern in [9]. This is the class of graphs whose vertices can be represented as simple paths on a rectangular grid so that two vertices are adjacent if and only if the corresponding paths share at least one edge of the grid.

EPG graphs have a practical use, e.g., in the context of circuit layout setting, which may be modeled as paths (wires) on a grid. In the knock-knee layout model, two wires may either cross or bend (turn) at a common grid point, but are not allowed to share a grid edge; that is, overlap of wires is not allowed. In this context, some of the classical optimization graph problems are relevant, e.g., maximum independent set and coloring. More precisely, the layout of a circuit may have multiple layers, each of which contains no overlapping paths. Referring to a corresponding EPG graph, then each layer is an Independent Set and a valid partitioning into layers corresponds to a proper coloring.

In [9], the authors show that every graph is an EPG graph. That is, for every graph $G = (V, E)$ there exists an EPG representation $\langle \mathcal{P}, \mathcal{G} \rangle$ where $\mathcal{P} = \{P_v : v \in V\}$ is a collection of paths on a grid $\mathcal{G}$, corresponding to the vertices of V and satisfying: paths $P_v, P_u \in \mathcal{P}$ share a grid edge of $\mathcal{G}$ if and only if $(v, u) \in E$.

F. Dehne, R. Solis-Oba, and J.-R. Sack (Eds.): WADS 2013, LNCS 8037, pp. 328–340, 2013.

Moreover, they showed that if G has n vertices and m edges, then there exists an EPG representation $\langle \mathcal{P}, \mathcal{G} \rangle$ of G in which $\mathcal{G}$ is a grid of size $n \times (n+m)$ and the paths in $\mathcal{P}$ are monotonic. As such, much of the current research today focuses on subclasses of EPG graphs, and, in particular, limiting the type of paths allowed.

A turn of a path at a grid point is called a *bend* and a graph is called a *k-bend EPG graph* (denoted B_k-EPG) if it has an EPG representation in which each path has at most k bends. It is both interesting mathematically, and justified by the circuit layout application described above, to consider subclasses of graphs, e.g., by bounding the number of bends allowed in each path.

A number of mathematical results on B_k-EPG graphs have been shown recently. In [2], the authors show that for any k, only a small fraction of all labeled graphs on n vertices are B_k-EPG. Improving a result of [3], it was shown in [12] that every planar graph is a B_4-EPG graph. It is still open whether $k = 4$ is best possible. So far it is only known that there are planar graphs that are B_3-EPG graphs and not B_2-EPG graphs. The authors in [12] also show that all outerplanar graphs are B_2-EPG graphs thus proving a conjecture of [3]. For the case of B_1-EPG graphs, Golumbic, Lipshteyn and Stern [9] showed that every tree is a B_1-EPG graph, and Asinowski and Ries [1] showed that every B_1-EPG graph on n vertices contains either a clique or a stable set of size at least $n^{1/3}$. In [1], the authors also give a characterization of the B_1-EPG graphs among some subclasses of chordal graphs, namely, chordal bull-free graphs, chordal claw-free graphs, chordal diamond-free graphs, and special cases of split graphs. In [5], a characterization of the subfamily of cographs that are B_1-EPG graphs is given by a complete family of minimal forbidden induced subgraphs.

The simplest case, B_0-EPG graphs, where all paths a straight line segments, are exactly the well studied class of *interval graphs* (the intersection graphs of intervals on a line), and it is well-known that these can be colored optimally with the exact minimum number of colors $\chi(G)$ in polynomial time (see [8]). This is no longer the case when $k > 0$.

In this paper, we consider approximation algorithms for B_1-EPG which are the edge intersection graphs of (at most) single bend paths on a rectangular grid. Heldt et al. [11] have proved that the recognition problem for B_1-EPG is NP-complete. Moreover, Cameron, Chaplick and Hoang [4] proved that even the recognition of a subclass of B_1-EPG know as $\llcorner$-EPG is NP-complete; we define this subclass in Section 2 below. Thus, for all of the algorithms that we will later present, an EPG representation $\langle \mathcal{P}, \mathcal{G} \rangle$ of G is assumed to be given as part of the input.

Maximum Independent Set, Minimum Coloring, and Maximum Clique are fundamental optimization problems in graph-theory. These problems arises naturally in many scenarios involving resource allocation in the presence of interference. The graph coloring problem deals with assigning colors to the vertices of a graph such that no two adjacent vertices share the same color, and the number of colors used is minimized. A coloring using at most c colors is called a (proper) c-coloring. The smallest number of colors needed to color a graph G is called its *chromatic number*, and is denoted $\chi(G)$. The graph coloring problem is known to be NP-hard. The current best known approximation ratio for the graph coloring problem is $O(n\frac{(\log\log n)^2}{(\log n)^3})$, where n is the number of vertices in the graph; see [10]. In graph theory, an *Independent Set* (*Stable Set*) is a set of vertices in a graph, where no two of which are adjacent. This corresponds to a *Clique* in the graphs complement. The size of a maximum independent set of a graph G is denoted by $\alpha(G)$, and the size of a maximum clique is denoted by $\omega(G)$. The problem of finding a largest independent set for a given graph G is called the *Maximum Independent Set Problem* (MIS) which is NP-hard. Even for graphs whose maximum degree is bounded by b, the current best known approximation ratio for the MIS problem are a fraction of b, see references in [13].

The paper is organized as follows. We begin with preliminary definitions in Section 2. In Section 3, we first prove that coloring B_1-EPG graphs is NP-complete, and then we present a 4-approximation algorithm for coloring B_1-EPG graphs in polynomial time. Similarly, in Section 4, we prove that finding a maximum independent set in B_1-EPG graphs is NP-complete, and then present a 4-approximation algorithm for the problem. Conclusions and open problems are given in Section 5 where we note that the maximum clique problem can be optimally solved in polynomial time for B_1-EPG graphs.

2 Preliminaries

Let $\langle \mathcal{P}, \mathcal{G} \rangle$ be a B_1-EPG representation of a graph $G = (V, E)$. We say that paths P_v and P_u are adjacent paths if v and u are adjacent vertices in G, i.e., P_v and P_u share a common grid edge of $\mathcal{G}$. We also say that $G = EPG(\langle \mathcal{P}, \mathcal{G} \rangle)$. In B_1-EPG graphs, each vertex corresponds to a path of one of the following shapes: ⌞, ⌜, ⌟ or ⌝, allowing horizontal or vertical segments as well. We refer to a path of shape $\tau \in \{⌞, ⌜, ⌝, ⌟, |, -\}$ as an τ-path. We denote by $\mathcal{P}_{⌞}$ the collection of ⌞-paths in $\mathcal{P}$, and similarly we use the notations $\mathcal{P}_{⌟}$, $\mathcal{P}_{⌜}$and $\mathcal{P}_{⌝}$. For no-bend paths we complete the definition by referring them as ⌞-paths. Sometimes, it is of interest to consider even finer, more restrictive subclasses of B_1-EPG by limiting the type of bends that are allowed, namely, the subclasses formed by the subsets of the four single bend shapes (i.e., $\{⌞\}$, $\{⌟, ⌞\}$, $\{⌞, ⌝\}$, $\{⌞, ⌜, ⌝\}$, where all other subsets are isomorphic to these up to $90°$ rotation), allowing paths with no-bend as well. We denote these classes by ⌞-EPG, ⌟⌞-EPG, ⌞⌝-EPG and ⌞ ⌜ ⌝-EPG respectively.

Let G be a ⌟⌞-EPG with grid representation $\langle \mathcal{P}, \mathcal{G} \rangle$. We define the lexicographic (LEX) order $\prec$ on the paths in $\mathcal{P}$ as follows; see Figure 1. For path $P_v \in \mathcal{P}$ we denote by ∂P_v the bottommost-leftmost grid point that is contained in P_v, that is, $\partial P_v = \min_y \{\min_x \{(x, y) \in P_v\}\}$. We say that $P_v \prec P_u$ if ∂P_v lies below ∂P_u or they both lie in the same row and ∂P_v is left of ∂P_u. We complete $\prec$ to a total order by arbitrarily breaking ties.

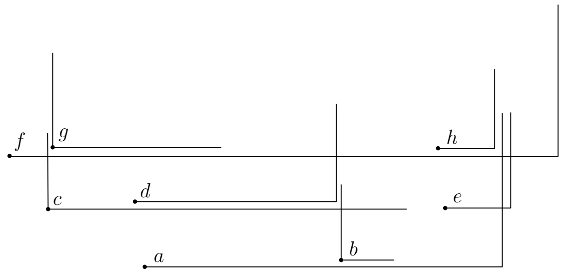


Fig. 1. The LEX ordering of a $\lrcorner\llcorner$-EPG representation: $a \prec b \prec c \prec d \prec e \prec f \prec g \prec h$

3 Coloring B_1-EPG Graphs

3.1 Hardness Result for Coloring B_1-EPG Graphs

In this section, we prove that coloring problem on B_1-EPG graphs is NP-complete by a reduction from the problem of coloring circle graphs which was shown to be NP-complete in [7].

We start by defining circle graphs. A *circle graph* is the intersection graph of a set of chords of a circle. That is, it is an undirected graph whose vertices can be associated with chords of a circle such that two vertices are adjacent if and only if the corresponding chords cross each other. We may assume without loss of generality that no two chords in the diagram of chords of the circle share a common endpoint. Coloring circle graphs remains NP-complete even if the graph is given by its chord model [7].

Theorem 1. *Let G be a B_1-EPG graph. Coloring G with the exact number of colors $\chi(G)$ is NP-complete.*

Proof. Let G be a circle graph. We construct a B_1-EPG representation for a graph G' so that G is c-colorable if and only if G' is. The construction is as follows; see Figures 2 and 3 for an illustration. We slide all the endpoints of the chords to the upper right quadrant of the circle, while preserving their order on the circle (thus, intersections are not changed under these transformations). Now, we replace each chord by an $\llcorner$-shape bend path, where every vertex v in G corresponds to a path P_v with the same endpoints on the circle. Note that since we assumed that all endpoints are distinct, the horizontal segment of each path lies on a unique horizontal line, and the vertical segment lies on a unique vertical line. Moreover, the intersection points of pairs of paths are in one-to-one correspondence with the edges of the graph.

Consider an intersection point between two paths P_v and P_u in the representation, where the horizontal section of P_v intersects with the vertical segment of P_u. We split P_v at the intersection point into two disjoint parts; the left part is a $\llcorner$-path, and the right one is a $-$-path. We complete the latter to a $\llcorner$-path by joining it to a vertical segment that overlaps only P_u. We also add $(c-1)$ short

Fig. 2. (a) A circle diagram. (b) Each chord is replaced by a single-bend path on the grid.

—-paths overlapping only these two segments of the former path P_v. Perform this transformation for every intersection point, and let G' be the B_1-EPG-graph of this transformed representation. This, of course, may have split P_v into several segments, $P_{v_1}, P_{v_2}, \ldots, P_{v_k}$, with consecutive segments P_{v_i} and $P_{v_{i+1}}$ being joined by such a set of $(c-1)$ short horizontal paths: a $(c-1)$-clique in G' overlapping only P_{v_i} and $P_{v_{i+1}}$. See Figure 3 for an illustration.

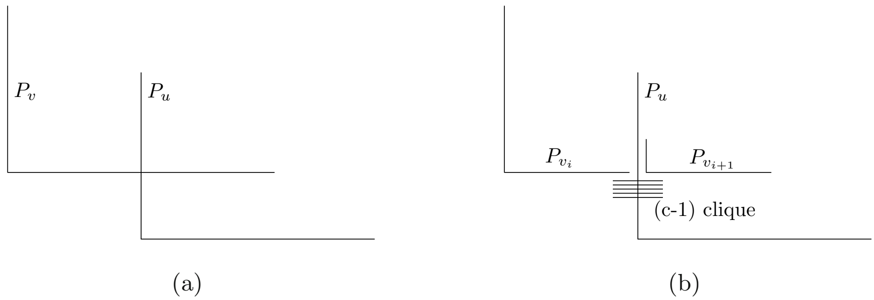


Fig. 3. (a) Intersecting paths. (b) The horizontal is "split" and "glued" using a $(c-1)$-clique.

It is clear from the transformation that the obtained graph G' is indeed a B_1-EPG graph. Moreover, the transformation can be performed in polynomial time and the size of G' is polynomial in the size of G, since $|V(G')| = n + ce \leq n + n^3$, where G has n vertices and e edges.

We now claim that G is c-colorable if and only if G' is c-colorable. Let $\varphi : V \mapsto \{1, \cdots, c\}$ be a valid assignment of colors for G. Then to color G' it suffices to (1) color each vertex from G' that came from an original path P_v (including its vertical segment and all of its horizontal split segments $P_{v_1}, \ldots, P_{v_k}$) with the color used in G, and (2) for each newly added $(c-1)$-clique (the short segments overlapping only P_{v_i} and $P_{v_{i+1}}$ which have the same color in the construction), we can use the $(c-1)$ remaining colors. This clearly colors G' in c colors.

We now show that if G' is c-colorable then G is c-colorable. Assume we have a c-coloring of the graph G'. Since the $(c-1)$-clique connecting any P_{v_i} and $P_{v_{i+1}}$ requires $(c-1)$ colors, consequently, P_{v_i} and $P_{v_{i+1}}$ have the same remaining c^{th} color. Moreover, let P_u be the path that intersects P_v in G and whose intersection point with P_v is the split point between P_{v_i} and $P_{v_{i+1}}$, then P_u and $P_{v_{i+1}}$ are adjacent in G', thus get distinct colors. Since the coloring of G' is proper, it also gives a proper coloring of G: color the path representing v in G with the same (common) color of its split segments $P_{v_1}, \ldots, P_{v_k}$ in G'. This concludes the proof of the theorem. □

Observe that by our construction, the paths in G' are either $\llcorner$-paths or $-$-paths, we thus conclude:

Corollary 2. *Let G be a $\llcorner$-EPG graph. Coloring G with the exact number of colors $\chi(G)$ is NP-complete.*

3.2 A 4-Approximation Algorithm for Coloring B_1-EPG Graphs

We start by presenting a "subroutine" in Algorithm 3.1 that computes an approximation solution for a $\lrcorner\llcorner$-EPG representation. We then apply it more generally to an arbitrary B_1-EPG representation. It is a greedy First-Fit algorithm using the LEX ordering $\prec$, defined in Section 2 so clearly, it produces a proper coloring. Lemma 1 will show that when used for a $\lrcorner\llcorner$-EPG graph, Algorithm 3.1 achieves a 2-approximation. We will use the notation $c(v)$ for the color assigned to vertex v.

Algorithm 3.1 Greedy-$\lrcorner\llcorner$-EPG-Coloring (Input: $\mathcal{P} = \mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\llcorner}$)

```
for each P_v ∈ 𝒫 (in increasing order ≺) do
    c(v) ← least color not in use among v's neighbors
return total number k of distinct colors used and the coloring
```

Applying Algorithm 3.1 to the representation in Figure 1 gives the coloring: $c(a) = c(c) = c(f) = 1$; $c(b) = c(e) = c(g) = 2$; $c(d) = c(h) = 3$.

For every path $P_v \in \mathcal{P}$ we denote by $\widetilde{\Gamma}(P_v)$ the collection of paths adjacent to P_v that have been colored by Algorithm 3.1 prior to P_v. When convenient, we refer to $\widetilde{\Gamma}(P_v)$ as a set of vertices. The color assigned to P_v by Algorithm 3.1 is dependent only on the colors assigned to paths in $\widetilde{\Gamma}(P_v)$, thus we have Observation 3.

Observation 3. *Let $\langle \mathcal{P}, \mathcal{G} \rangle$ be a $\lrcorner\llcorner$-EPG representation of a graph $G = (V, E)$, and let P_v and P_u be adjacent paths. If $P_u \prec P_v$, then P_v and P_u share at least one of two grid edges e_1 and e_2 as follows:*

- *If P_v is a* $\llcorner$*-path, then e_1 and e_2 are respectively the horizontal and vertical grid edges contained in P_v and attached to its bend point.*
- *If P_v is a* $\lrcorner$*-path, then e_1 is the left-most horizontal grid edge contained in P_v and e_2 is the vertical grid edge attached to its bend point.*
- *If P_v is a $|$-path, then e_1 is the bottom-most vertical grid edge contained in P_v (e_2 in this case is undefined).*
- *If P_v is a $-$-path, then e_1 is the left-most horizontal grid edge contained in P_v (e_2 in this case is undefined).*

Lemma 1. *Let G be a* $\lrcorner\llcorner$*-EPG graph, then Algorithm 3.1 uses at most $2\chi(G)$ colors.*

Proof. Let k be the maximum color used by Algorithm 3.1, we show that $k \leq 2\chi(G)$. Indeed, put $G = (V, E)$ and let $v \in V$ be a vertex for which $c(v) = k$. Notice that whenever Algorithm 3.1 colors a vertex, the assigned color is determined by its previous-colored neighbors $\widetilde{\Gamma}(P_v)$. Notice that if Algorithm 3.1 colored v with color k, then k is the least color that not in use for any vertex $u \in \widetilde{\Gamma}(P_v)$, thus $k \leq \widetilde{\Gamma}(P_v) + 1$. Moreover, by Observation 3, we have that each path in $\widetilde{\Gamma}(P_v)$ shares at least one of two specified grid edges contained in $\mathcal{P}_v$ (denoted e_1 and e_2). We conclude that at least half of the paths in $\widetilde{\Gamma}(P_v)$ contain one of those edges and without loss of generality, we assume it is e_1. Now, observe that any collection of paths containing a common edge corresponds to a clique in G, in particular, those paths in $\widetilde{\Gamma}(P_v)$ that contain e_1 together with v itself, form a clique. We get $\frac{1}{2}\tilde{\Gamma}(v) + 1 \leq \omega(G) \leq \chi(G)$, thus $k \leq \widetilde{\Gamma}(P_v) + 1 < 2\omega(G) \leq 2\chi(G)$, which completes the proof. □

Remark 1. Clearly, by rotating a representation by 180°, Algorithm 3.1 can be "turned" from Greedy-$\lrcorner\llcorner$-EPG-Coloring into Greedy-$\urcorner\ulcorner$-EPG-Coloring.

We now use Algorithm 3.1 as a building block in Algorithm 3.2 in order to colors B_1-EPG graphs.

Algorithm 3.2 B_1-EPG Coloring 4-Approximation (Input: $G = EPG(\langle\mathcal{P}, \mathcal{G}\rangle)$)

1: Let $\mathcal{P} = \mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\urcorner} \cup \mathcal{P}_{\ulcorner}$
2: $k_1 \leftarrow$**Greedy-$\lrcorner\llcorner$-EPG-Coloring**$(\mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner})$
3: $k_2 \leftarrow$**Greedy-$\urcorner\ulcorner$-EPG-Coloring**$(\mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\urcorner})$ // using different color names //
4: **return** total number of distinct colors used and the coloring

---

Algorithm 3.2 partitions the paths in $\mathcal{P}$ into two subsets $\mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner}$ and $\mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\urcorner}$, each induces a subgraph of G, which is a $\lrcorner\llcorner$-EPG graph (denoted G_1 and G_2 respectively). Then, it colors each of these two graphs G_1 and G_2 using Algorithm 3.1, with distinct "palettes" of colors. Clearly, the coloring produced by

Algorithm 3.2 is proper. Notice that in order to color a graph G, one needs at least the maximum of $\chi(G_1), \chi(G_2)$ colors. By Lemma 1, Algorithm 3.2 uses at most $2\chi(G_1) + 2\chi(G_2) \leq 4\chi(G)$ colors, we thus have Theorem 4 below.

Theorem 4. *Let G be a B_1-EPG graph, then Algorithm 3.2 uses at most $4\chi(G)$ colors.*

4 Maximum Independent Set on B_1-EPG Graphs

4.1 Hardness Result for Finding Maximum Independent Set on B_1-EPG Graphs

In this section, we show that the MAXIMUM INDEPENDENT SET on B_1-EPG graphs is NP-complete. We use a reduction from MAXIMUM INDEPENDENT SET on planar graphs with maximum degree four, which is known to be NP-complete [6]; our proof is inspired by [14].

Theorem 5. MAXIMUM INDEPENDENT SET *on B_1-EPG graphs is NP-complete.*

Proof. Let $G = (V, E)$ be a planar graph with maximum degree four; MAXIMUM INDEPENDENT SET on planar graph with maximum degree four is NP-complete [6]. We construct a B_1-EPG representation of a graph $G' = (V', E')$ so that a maximum independent set in G' corresponds to a maximum independent set in G and vice versa.

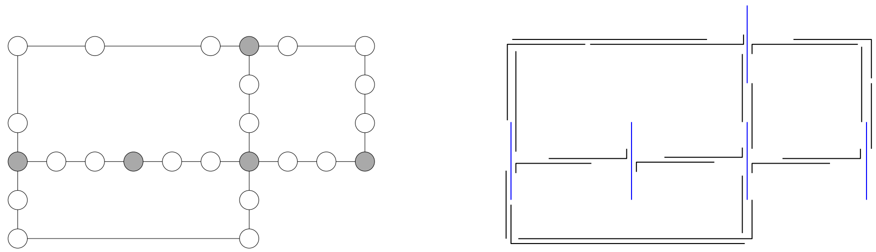
Fix an embedding of G in a grid $\mathcal{G}$ such that edges of G are piecewise linear curves following the grid lines (such an embedding in a linear sized grid always exists and is constructible in polynomial time [16]). Each edge $e \in E$ is thus corresponds to a path π_e in the grid $\mathcal{G}$, and denote by k_e the number of segments (*links*) π_e consists of. Note further, that these paths intersect only at their endpoints, namely, in the vertices of G since the embedding is planar.

Let G' be a graph obtained from G by subdividing every edge e with $2\left\lceil \frac{k_e+1}{2} \right\rceil$ new vertices; we denote the set of new vertices corresponding to an edge e by U_e and by U the set of all such new vertices, we thus have $V' = V \cup U$. Notice that since $|U_e|$ is even for each edge e of G, a maximum independent set in G' contains exactly half of the vertices in U_e, and at most one of the vertices corresponding to the "original" endpoints of e. We thus have

$$\alpha(G') = \alpha(G) + \sum_{e \in E} \left\lceil \frac{k_e + 1}{2} \right\rceil$$

and thus to complete the proof it suffices to show that G' is B_1-EPG graph.

Having the grid embedding of G, we construct a B_1-EPG representation $\langle \mathcal{P}, \mathcal{G} \rangle$ of G' as follows; see Figure 4 for an illustration. We start by placing the vertices in U into $\mathcal{G}$. Let e be an edge of G, by definition π_e has $k_e - 1$ bend points. At each such grid point we place one vertex from U_e, we also place one vertex from U_e in the interiors of the first and last links of π_e. Finally, we place the remaining vertices of U_e arbitrarily along π_e (the order in which the vertices are located along π_e preserves adjacencies). When convenient we may refer to vertices of G' as the grid points they are embedded to. We now associate each vertex v of G' with a path P_v (which is either a single-bend path or a segment) so that P_v and P_u share an edge of $\mathcal{G}$ if and only if v and u are adjacent in G'.



(a) (b)

Fig. 4. (a) A rectilinear grid embedding of some graph G'; vertices of V are grayed. (b) A B_1-EPG representation of G'.

For every $v \in V$, set P_v to be a short vertical segment around v. Let $u \in U$, then u has exactly two neighbors, and consider first the case where both are from U. We set P_u to be a path consisting of the two segments connecting u with each of its neighbors. If u is embedded to a bend point of some π_e, then P_u is a single-bend path, otherwise it is just a segment. Finally, let $u \in U$ be a vertex with neighbors $u' \in U$ and $v \in V$ (notice that by construction no vertex in G' has more than one neighbor from V) in this case, u, u', and v are embedded to the same grid row/column and we set P_u as follows, distinguishing between two subcases, according to whether all three vertices are embedded to the same column or row of $\mathcal{G}$. (i) u, u', and v are on the same column: We set P_u to be a vertical segment that begins at u' and almost reaches v (in such a way that it ends close enough to share a grid edge with P_v). (ii) u, u', and v are on the same row: We set P_u to be a ┘-path or a ┌-path that starts at u' and bends at v, sharing its vertical edge with P_v, avoiding other possible neighbors of v.

It is easy to see that indeed for every $u, v \in V'$ the paths P_u and P_v share a grid edge if and only if u and v are adjacent in G', thus the desired result follows. □

Remark 2. The proof of Theorem 5 can be modified so that it uses only two bend shapes; thus Maximum Independent Set is NP-complete already on ┘└-EPG and on └┐-EPG graphs.

4.2 A 4-Approximation Algorithm for Maximum Independent Set on B_1-EPG Graphs

In this section we present a constant-factor approximation algorithm for Maximum Independent Set (Algorithm 4.2 below). In a similar way to Section 3.2, we start by presenting a "subroutine" that computes an approximated solution for a subgraph, and then use the subroutine in order to compute an approximated solution for the whole graph. This subroutine is described in Algorithm 4.1 below, which uses a standard greedy Independent Set algorithm (thus clearly, produces an Independent Set). Note that the order in which it examines the vertices is the *reversed order* of that used in Algorithm 3.1, namely, according to the decreasing order of $\prec$. Lemma 2 claims that when used for a $\lrcorner\llcorner$-EPG graph, Algorithm 4.1 computes a 2-approximation.

Algorithm 4.1 Greedy-$\lrcorner\llcorner$-EPG-Independent-Set (Input: $\mathcal{P} = \mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\llcorner}$)

1: $S \leftarrow \emptyset$
2: **for each** $P_u \in \mathcal{P}$ (in decreasing order by $\prec$) **do**
3:     add u to S and remove P_u from $\mathcal{P}$
4:     remove all paths corresponding to u's neighbors from $\mathcal{P}$
5: **return** S

Applying Algorithm 4.1 to the representation in Figure 1 gives the independent set: $\{h, g, d\}$.

Lemma 2. *Let G be a $\lrcorner\llcorner$-EPG graph, then Algorithm 4.1 finds a maximal independent set of size at least $\frac{1}{2}\alpha(G)$.*

Proof. Let $\langle \mathcal{P}, \mathcal{G} \rangle$ be a $\lrcorner\llcorner$-EPG representation of a graph $G = (V, E)$. Let OPT be a maximum independent set in G and let S be the maximal Independent Set returned by Algorithm 4.1. We claim that $|OPT| \leq 2|S|$.

Notice that for every $v \in V$ the path P_v is removed from $\mathcal{P}$ at some point (in lines 3 or 4). Moreover, if a path P_v is removed from $\mathcal{P}$ in line 4, then its deletion must occur when the algorithm added to S some vertex u with $v \prec u$. Equivalently, whenever the algorithm adds a vertex u to S, it removes from $\mathcal{P}$ paths P_v adjacent to P_u where $v \prec u$ (in this case, any other vertex v' adjacent to u with $u \prec v'$ has been already removed from S in an earlier stage, necessarily in line 4).

By eliminating vertices in $OPT \cap S$ we may assume that $OPT \cap S = \emptyset$. We therefore assume that the paths corresponding to vertices in OPT were all eliminated from $\mathcal{P}$ in line 4. We define a correspondence $\varphi : OPT \to S$ as follows:

$$\varphi(v) = u \text{ where } P_v \text{ was removed from } \mathcal{P} \text{ in line 4 as a consequence of adding } u \text{ to } S$$

In particular, if $\varphi(v) = u$ then u and v are adjacent and $v \prec u$. We claim that for every $u \in S$ there exist at most two distinct vertices $v_1, v_2 \in OPT$ with $\varphi(v_1) = \varphi(v_2) = u$ and conclude that $|OPT| \leq 2|S|$. Indeed, assume to the contrary that for some $u \in S$, there exist three vertices $v_1, v_2, v_3 \in OPT$ with $\varphi(v_i) = u$ $(i = 1, 2, 3)$. At least two of the three paths share with P_u a grid edge on the same direction; w.l.o.g., assume that P_{v_1} and P_{v_2} share a horizontal edge with P_u. We thus have that P_{v_i} is adjacent to P_u and $v_i \prec u$ $(i = 1, 2)$, and in particular P_u, P_{v_1} and P_{v_2} share a common edge (the leftmost-bottommost grid-edge contained in P_u). However, as v_1 and v_2 are both in OPT, they are nonadjacent. – A contradiction. $\square$

We now use Algorithm 4.1 as a building block in Algorithm 4.2 in order to find a maximal Independent Set in B_1-EPG graphs. Here too, as in Remark 1, by rotating a representation by 180°, Algorithm 4.1 can be "turned" from Greedy-$\lrcorner\llcorner$-EPG-Independent-Set into Greedy-$\urcorner\ulcorner$-EPG-Independent-Set. Theorem 6 claims that when used on a B_1-EPG graph, Algorithm 4.2 achieves a 4-approximation.

Algorithm 4.2 B_1-EPG Independent Set 4-Approximation$(G = \langle \mathcal{P}, \mathcal{G} \rangle)$

1: let $\mathcal{P} = \mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\urcorner} \cup \mathcal{P}_{\ulcorner}$
2: $S_1 \leftarrow$**Greedy-$\lrcorner\llcorner$-EPG-Independent-Set**$(\mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner})$
3: $S_2 \leftarrow$**Greedy-$\urcorner\ulcorner$-EPG-Independent-Set**$(\mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\urcorner})$
4: **return** the largest amongst S_1, S_2

Theorem 6. *Let G be a B_1-EPG graph, then Algorithm 4.2 finds a maximal Independent Set of size at least $\frac{1}{4}\alpha(G)$.*

Proof. Let $\langle \mathcal{P}, \mathcal{G} \rangle$ be a B_1-EPG representation of G. Put $\mathcal{P} = \mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\urcorner} \cup \mathcal{P}_{\ulcorner}$ and let G_1 and G_2 be the $\lrcorner\llcorner$-EPG graphs with representations $\langle \mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner}, \mathcal{G} \rangle$ and $\langle \mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\urcorner}, \mathcal{G} \rangle$, respectively. Clearly, $\alpha(G) \leq \alpha(G_1) + \alpha(G_2)$.

Let S_1 and S_2 be the sets computed in lines 2 and 3 of the algorithm. By Lemma 2, we get

$$\alpha(G) \leq \alpha(G_1) + \alpha(G_2) \leq 2|S_1| + 2|S_2| \leq 4 \max\{|S_1|, |S_2|\}$$

which completes the proof. $\square$

5 Concluding Remarks

We observe that Maximum Clique in B_1-EPG graphs can be optimally solved in polynomial time using a brute-force algorithm. In [9] the authors show that each clique in the graph has one of two forms in the B_1-EPG representation, referred to as "edge clique" and "claw clique". An *edge clique* consists of all paths containing a given grid edge; a *claw clique* consists of all paths sharing two-out-of-three edges of a given claw centered at a given grid point (there are 4 different claws at each grid point.) Consequently, given a grid representation of a B_1-EPG graph G, one can simply examine each grid edge and count the number of paths containing that edge, and for each grid point and four corresponding claws, count the number of path containing two out of three edges of that claw. This can be done in time polynomial in the size of $\mathcal{G}$, which may be assumed to be of size at most $2n \times 2n$ for a B_1-EPG representation. This implies an $O(n^3)$ time algorithm for Maximum Clique given a B_1-EPG representation.

A somewhat different approach can solve MAXIMUM CLIQUE for a B_1-EPG graph *without* being given representation based on the fact that the neighborhood of a vertex in B_1-EPG graph is weakly-chordal [1]. It is well known that MAXIMUM CLIQUE in weakly-chordal graphs can be found in $O(n^4)$ time [15]. Since a maximum clique is contained in a closed neighborhood of each of its vertices, then this yields a $O(n^5)$ time algorithm for MAXIMUM CLIQUE given just the B_1-EPG graph and not the representation.

In Algorithms 3.2 and 4.2 we used, respectively, Algorithms 3.1 and 4.1 with subgraphs induced by $\mathcal{P}_{\llcorner} \cup \mathcal{P}_{\lrcorner}$ and $\mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\urcorner}$. Taking into consideration also the two other options (i.e., $\mathcal{P}_{\lrcorner} \cup \mathcal{P}_{\urcorner}$ and $\mathcal{P}_{\ulcorner} \cup \mathcal{P}_{\llcorner}$) has no effect on the asymptotic quality of the solutions. However, as a heuristic, one might wish to apply the algorithm to both and take the better of the two.

Algorithm 3.2 and Algorithm 4.2 are greedy. Both have "bad" instances for which the factors mentioned here are tight. It is possible, of course, that a different approach may lead to better approximation factors.

As open problems, we suggest that it would be interesting to find approximation algorithms to find a minimum dominating set or a maximum weighted independent set for B_1-EPG graphs.

References

1. Asinowski, A., Ries, B.: Some properties of edge intersection graphs of single bend paths on a grid. Discrete Mathematics 312, 427–440 (2012)
2. Asinowski, A., Suk, A.: Edge intersection graphs of systems of grid paths with bounded number of bends. Discrete Applied Mathematics 157, 3174–3180 (2009)
3. Biedl, T., Stern, M.: On edge intersection graphs of k-bend paths in grids. Discrete Mathematics & Theoretical Computer Science (DMTCS) 12, 1–12 (2010)
4. Cameron, K., Chaplick, S., Hoang, C.T.: Recognizing Edge Intersection Graphs of $\llcorner$-Shaped Grid Paths. In: LAGOS 2013 (to appear, 2013)
5. Cohen, E., Golumbic, M.C., Ries, B.: Characterizations of cographs as intersection graphs of paths on a grid (submitted)
6. Garey, M.R., Johnson, D.S.: Computers and Intractability: a Guide to the Theory of NP-completeness. Freeman, San Francisco (1979)
7. Garey, M.R., Johnson, D.S., Miller, G.L., Papadimitriou, C.: The complexity of coloring circular arcs and chords. SIAM. J. on Algebraic and Discrete Methods 1, 216–227 (1980)
8. Golumbic, M.C.: Algorithmic Graph Theory and Perfect Graphs. Academic Press, New York (1980); Annals of Discrete Mathematics, 2nd edn., vol. 57. Elsevier, Amsterdam (2004)
9. Golumbic, M.C., Lipshteyn, M., Stern, M.: Edge intersection graphs of single bend paths on a grid. Networks 54, 130–138 (2009)
10. Halldórsson, M.M.: A still better performance guarantee for approximate graph coloring. Information Processing Letters 45, 19–23 (1993)
11. Heldt, D., Knauer, K., Ueckerdt, T.: Edge-intersection graphs of grid paths: the bend-number, Arxiv preprint arXiv:1009.2861, arxiv.org (September 2010)
12. Heldt, D., Knauer, K., Ueckerdt, T.: On the bend-number of planar and outerplanar graphs. In: Fernández-Baca, D. (ed.) LATIN 2012. LNCS, vol. 7256, pp. 458–469. Springer, Heidelberg (2012)

13. Kako, A., Ono, T., Hirata, T., Halldórsson, M.M.: Approximation algorithms for the weighted independent set problem in sparse graphs. Discrete Applied Mathematics 157, 617–626 (2009)
14. Kratochvíl, J., Nešetřil, J.: Independent set and clique problems in intersection-defined classes of graphs. Commentationes Mathematicae Universitatis Carolinae 31, 85–93 (1990)
15. Spinrad, J.P., Sritharan, R.: Algorithms for weakly triangulated graphs. Discrete Appl. Math. 59, 181–191 (1995)
16. Valiant, L.G.: Universality considerations in VLSI circuits. IEEE Trans. Comput. 30, 135–140 (1981)