Approximability of Edge Matching Puzzles

Antonios Antoniadis[1] and Andrzej Lingas[2]

[1] Institut für Informatik, Humboldt-Universität zu Berlin, 10099 Berlin, Germany
antoniad@informatik.hu-berlin.de
[2] Department of Computer Science, Lund University, S-22100 Lund, Sweden
Andrzej.Lingas@cs.lth.se

Abstract. This paper deals with the (in)approximability of *Edge Matching Puzzles*. The interest in EdgeMatching Puzzles has been raised in the last few years with the release of the *Eternity II*™ puzzle, with a $2 million prize for the first submitted correct solution. It is known [1] that it is NP-hard to obtain an exact solution to Edge Matching Puzzles. We extend on that result by showing an approximation-preserving reduction from Max-3DM-B and thus proving that Edge Matching Puzzles do not admit polynomial-time approximation schemes unless P=NP. We then show that the problem is APX-complete, and study the difficulty of finding an approximate solution for several other optimisation variants of the problem.

1 Introduction

Informally, an *Edge Matching Puzzle* is a puzzle where the goal is to arrange a given set of square tiles with coloured edges into a given rectangle so that colours match along the edges of adjacent tiles. Edge Matching Puzzles first appeared in the 1890s. They are more challenging than the classical jigsaw puzzles we are all familiar with; mainly because there is no global image that can provide guidance. Additionally, there are usually more pieces that match together, but one cannot be sure they should be placed next to each other before attaining

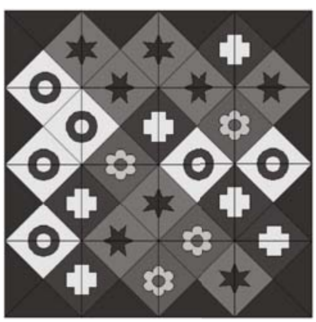


Fig. 1. A solved Edge Matching Puzzle instance with no edges broken

J. van Leeuwen et al. (Eds.): SOFSEM 2010, LNCS 5901, pp. 153–164, 2010.

the complete solution. In other words, a local solution does not generally lead to a global solution.

The computational problem of filling plane areas with rectangular tiles has been extensively studied. Berger [2] has shown that a generalisation of our problem where an infinite number of copies for each tile is given and the goal is to fill the entire plane, is undecidable. The problem of wether a bounded area can be filled by a subset of the given tiles is NP-complete [3]. Additionally, it has been shown that bounded-tiling (having infinite copies of each tile and filling a bounded area) is a viable alternative to the satisfiability problem as a foundation of NP-completeness [4]. The complexity of the game variant of the problem has also been studied: It is PSPACE-complete and EXPTIME-complete when the tiles are to be placed into a square and a rectangle respectively [5]. Unlike the variant we study, all the above do not allow rotations of the tiles given.

Demaine and Demaine [1] established the NP-completeness of edge matching puzzles and some of its variants (a species of jigsaw puzzles, signed edge matching puzzles, and polyonimo packing puzzles) by a reduction from *3-partition*. Their result confirmed the difficulties that people have had in trying to solve this puzzle, and justified the exhaustive search that seemed necessary for the puzzle to be solved by a computer. Benoist and Bourreau [6] studied Edge Matching Puzzles using constraint programming, and Ansótegui et al. [7] worked on the generation of EMP instances of varied hardness, and the application of SAT/CSP solving techniques to the problem.

We show that the maximisation version of a variant of the problem is APX-complete by presenting an approximation-preserving reduction from a problem that is known to be APX-complete (namely Max-3DM-B, defined later) to our problem, and providing a constant-factor approximation algorithm for the problem. The APX-hardness result is then used to show some equivalent results for some other optimisation variants of Edge Matching Puzzles.

1.1 Outline

In the next section some definitions that will be used later on are presented. In Sect. 3, we present and analyse the actual reduction. Finally, Sect. 4 adds some results regarding the minimisation version.

2 Preliminaries

We adhere to the definitions of *approximation ratio*, *relative error* and *absolute error* from [8].

Definition 1. APX *is the complexity class of all optimisation problems Q such that the decission version of Q is in NP, and for some $r \geq 1$ there exists a polynomial-time r-approximation algorithm for Q.*

Definition 2. *A* Polynomial-Time Approximation Scheme *(PTAS) [8] for a problem is a set of algorithms A such that for each $\epsilon > 0$, there is an approximation algorithm in A with ratio $1+\epsilon$ for the problem, running in polynomial time (under the assumption that ϵ is fixed).*

Note 1. Since under the P $\neq$ NP conjecture, there exist problems in *APX* that do not admit a PTAS, if P$\neq$NP then no *APX-hard* problem can admit a PTAS.

Definition 3. *An* L-reduction *[9] with constant parameters* $\alpha, \beta > 0$*, from a problem* A *to a problem* B*, with cost functions* c_A *and* c_B *respectively (where* $c_X(v, w)$ *denotes the cost of solution* w *to a problem* X *on instance* v*), is a pair of polynomial time computable functions* f *and* g *such that the following hold.*

- f *transforms instances of* A *to instances of* B*.*
- *If* y *is a solution to* $f(x)$*, then* $g(y)$ *is a solution to* x*.*
- *For every instance* x *of* A*:* $OPT_B(f(x)) \leq \alpha OPT_A(x)$*.*
- *For every solution* y *to* $f(x)$*:* $|OPT_A(x) - c_A(x, g(y))| \leq \beta |OPT_B(f(x)) - c_B(f(x), y)|$*.*

Theorem 1. *([9]) If there is an L-reduction with parameters* α *and* β*, from a problem* A *to a problem* B*, and there exists a polynomial-time approximation algorithm for* B *with relative error* c*, then there also exists a polynomial-time approximation algorithm for* A *with relative error* $\delta = \alpha\beta c$*.*

Definition 4. *Formally, an* Edge Matching Puzzle (EMP) *is a puzzle where the goal is to arrange a given collection of* n *square-shaped and edge-coloured tiles (of area* a *each), into a given rectangle of area* $n \cdot a$ *such that adjacent tiles are coloured identically along their common edge.*

We say that an edge e *in a solution of an EMP is* broken *if the two adjacent tiles in that solution sharing* e *have different colours along it. In the* maximisation version *of EMP, which we consider in the following section, we are looking for a solution maximizing the number of edges that match (are not broken).*

Definition 5. *In* Maximum Three-Dimensional Matching (Max-3DM) *we are given a set of triples* $T \subseteq X \times Y \times Z$ *from pairwise disjoint sets* X, Y *and* Z*, and we are looking for a subset* M *of the triples* T *of maximum size, such that no two triples of* M *agree on any coordinate.*

In the bounded version of Max-3DM, *Maximum Bounded 3-Dimensional Matching* (Max-3DM-B) the number of occurrences of every element in X, Y or Z is bounded by the constant B. Kann showed in 1991 [10] that Max-3DM-B is *APX-complete* for $B \geq 3$. More recently, Chlebìk and Chlebìkovà [11] improved that result by showing that Max-3DM-B is *APX-complete* for $B \geq 2$. Specifically, it is NP-hard to approximate the solution within $\frac{141}{140}$ even on instances with exactly two occurrences of each element. They later improved that bound to $\frac{95}{94}$ [12].

3 The Reduction and Its Analysis

Here we present an L-reduction of Max-3DM-B (with every element appearing exactly twice) to EMP. Whenever an edge of a tile has colour u it should be interpreted as a unique colour, thus it does not match to any other edge, including other u's. All other symbols that appear on edges of tiles represent a specific colour, and can be matched with edges of other tiles where the same symbol occurs.

3.1 Constructing the EMP Instance

This subsection describes how given a Max-3DM-B instance, a corresponding EMP instance is produced. An informal description on why the EMP instance is produced this way and what the purpose of every tile is, is provided in Subsect. 3.2; followed by a formal proof in Subsect. 3.3 and 3.4.

We define f as the function that given any instance of the Max-3DM-B problem with every element occuring exactly two times, will produce an instance of EMP as follows:

1. For each triple (x, y, z) with $x \in X$, $y \in Y$ and $z \in Z$, f produces the tiles seen in Fig. 2(a). We can call these tiles of *Type 1*, *Type 2* and *Type 3* respectively (from the lowest to the highest).
2. For each element $x \in X$, f produces the tiles seen in Fig. 2(b).
3. For each element $z \in Z$, f produces the tiles seen in Fig. 2(c).
4. For each element $y \in Y$, f produces the tiles seen in Fig. 2(d).
5. We are given a rectangle in which we want to arrange the tiles with the fewest possible edges being broken. The rectangle has height 2 and length half the number of tiles produced.

3.2 Informal Description

The purpose of this subsection is to provide an insight on the reduction and this way make the material and the proofs in the rest of this section easier to follow.

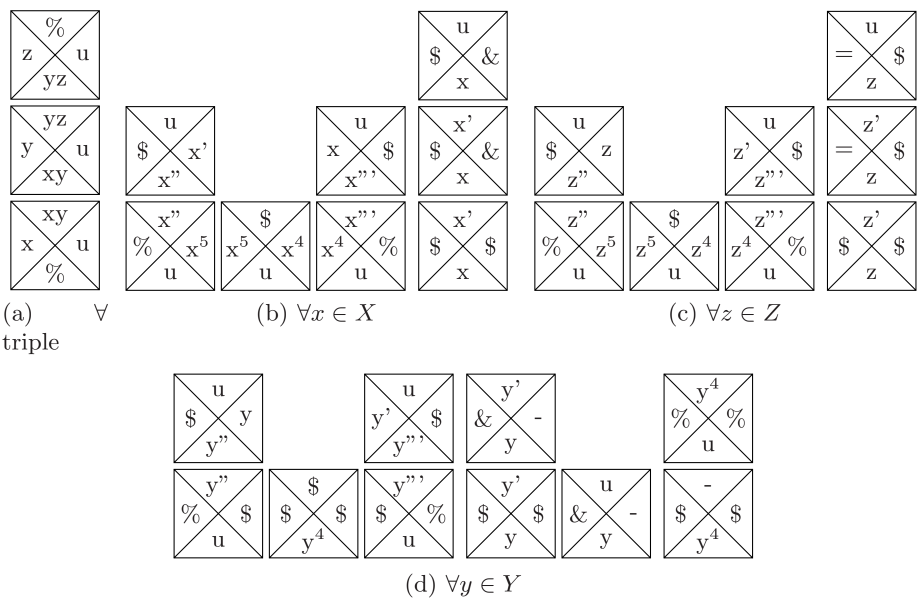
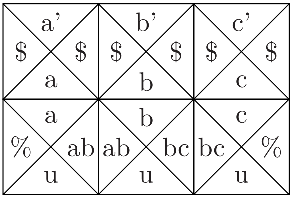
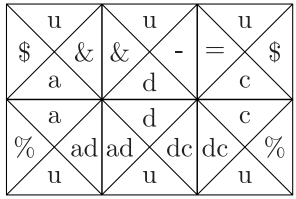


Fig. 2. The tiles constructed in Subsect. 3.1



(a) a set corresponding to a picked triple, with no edges broken

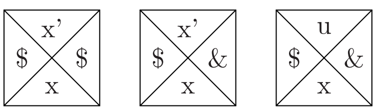


(b) a set corresponding to an not picked triple, with exactly one edge broken

Fig. 3. A "good" and a non "good" set

Suppose that we have the two triples (a, b, c) and (a, d, c) among others in our given Max-3DM-B instance. Clearly in an optimal solution to that instance at most one of these triples can be picked, assume that it is (a, b, c). The main idea is that after the construction of the EMP instance in the way described in Subsect. 3.1, its optimal solution will contain the two sets of tiles seen in Fig. 3.

This way whenever a triple is picked we have no broken edge in the corresponding set, and whenever one is not picked we have exactly one edge that is broken in the corresponding set. Also note that no two triples that have a common element can be picked. To make the two sets of Fig. 3 always possible we have additionally produced some excess tiles. These, are not being discarded (the EMP definition does not allow this) but will be placed somewhere else in the rectangle. We are aware that from the first three tiles made for each element exactly one tile will be excess. Taking as an example a tile corresponding to an element $x \in X$, out of the tiles



two will be used, and one will be excess. By the pigeonhole principle either the second or the third tile **has** to be used, and when used these tiles are equivalent because of differing only in the colour placed on the border. Thus we may assume that always the third tile is used and either the first or the second tile are unused. Less formally, an element can either appear in two not picked triples (the second and the third tile are used), or in one picked and one not picked triple (the first and third tile are used). Note here that no matter which of the first two tiles is the excess one it can be matched with the other tiles constructed for that element, producing a set (call it "dumping set"), with cost 0. For a detailed example of an arrangement, including the placement of excess tiles, see Fig. 4.

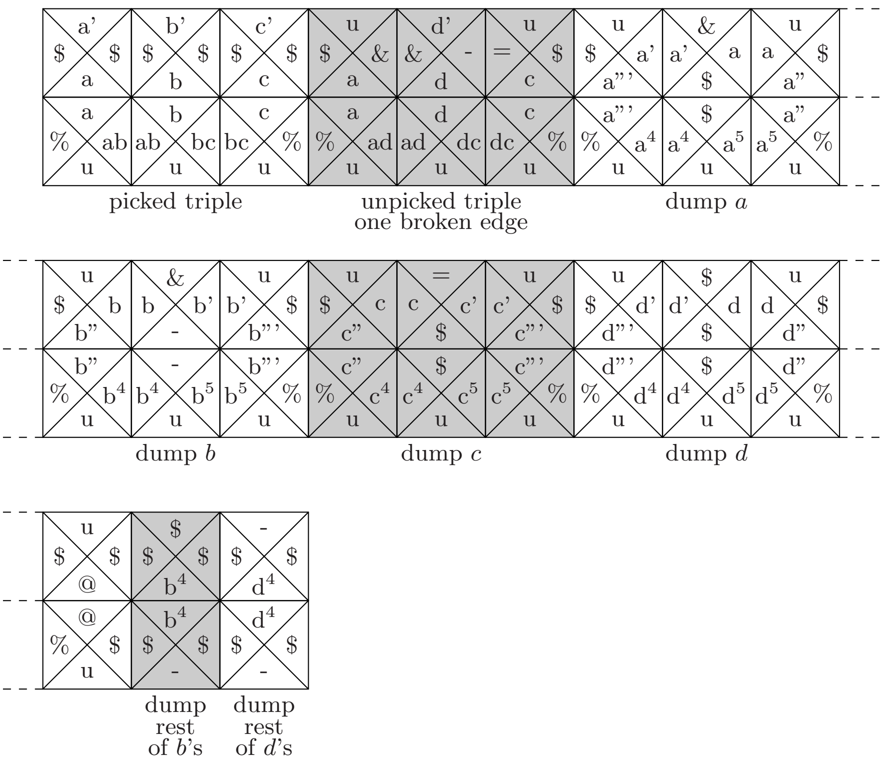


Fig. 4. A solution of the EMP instance, split into three pieces, corresponding to the Max-3DM-B instance consisting of choosing triple (a,b,c) but not (a,d,c)

3.3 Some Useful Lemmas

We say that a *good set* is a formation of six tiles arranged as seen in Fig. 3(a). Such a set in a solution of our instance of EMP always corresponds to a triple (a, b, c) that we want to pick in the Max-3DM-B instance.

A transformation of a solution to the EMP instance to a solution to the Max-3DM-B instance can easily be done in polynomial time by selecting the good sets, and picking the corresponding triples. By the construction of the instance of EMP there is a solution to it of the form seen in Fig. 4, which corresponds to the optimal solution of the initial Max 3-DM-B instance.

Lemma 1. *Given an instance C of Max-3DM-B containing n triples, an instance D of EMP can be constructed as described above. Any solution to D that breaks k edges yields a solution to C consisting of at least $n - k$ triples.*

Proof. The core idea of this proof is to show that if k edges are broken in the solution to D, then this solution must contain at least $n - k$ good sets.

Assume that our instance D has at least k_1 tiles of Type 2 *that are adjacent to a broken edge in the solution (note that $k_1 \leq k$). This means that the form of set seen in Fig. 5(a) will appear $n - k_1$ times.*

The symbol '?' in Fig. 5 is a placeholder indicating that the colour of that edge is still unknown. Assume now that out of these $n - k_1$ sets, k_2 have the upper edge of the bottom right tile broken, or the right edge of the top tile broken. This means that there are at least $n - k_1 - k_2$ sets where these edges are not broken. The only way for this to be the case, is when the sets look like in Fig. 5(b).

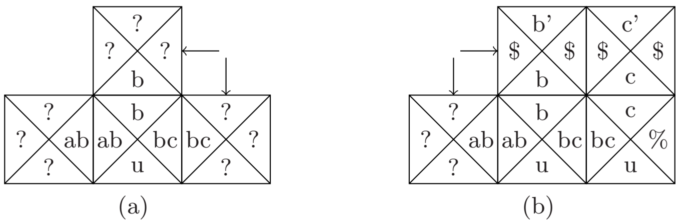


Fig. 5.

Finally, out of these $n - k_1 - k_2$ sets, let k_3 have the top edge of the bottom left tile broken, or the leftmost edge of the top row broken. It follows that we have at least $n - k_1 - k_2 - k_3$ good sets. As $k_1 + k_2 + k_3 \leq k$ (the total number of broken edges is at most k), we conclude that we have at least $n - k$ good sets, and thus select $n - k$ triples in our solution of the Max 3DM-B instance. □

As a natural consequence of the way the EMP instance is constructed and Lemma 1, we get the following Lemma.

Lemma 2. *Given an instance x of Max-3DM-B, let $f(x)$ denote the corresponding instance produced as described in Sect. 3.1. Given an optimal solution to $f(x)$, by picking the triples that correspond to good sets in the optimal solution of $f(x)$, an optimal solution to x is produced.*

3.4 The Reduction Preserves Approximability

Lemma 3. *The function f defined in Sect. 3.1 transforms instances of Max-3DM-B to instances of EMP, and can be computed in polynomial time.*

Lemma 4. *There is a polynomial-time computable function g, such that if y is a solution to $f(x)$, $g(y)$ is a solution to x.*

Proof. *The function g takes as input a solution to an instance $f(x)$ of EMP, and produces a solution to instance x of Max-3DM-B. This is done by selecting the triples corresponding to the good sets. Clearly also g can be computed in polynomial time.* □

Lemma 5. *Let x be a given instance to problem Max-3DM-B, with $B = 2$, $|E|$ representing the number of elements in instance x (that is $|E| = |X| + |Y| + |Z|$) and the number of triples in that instance being n. Then the following hold:*

- $OPT_{\mathrm{Max-3DM-B}}(x) \geq \frac{1}{4}n$
- $OPT_{\mathrm{Max-3DM-B}}(x) \geq \frac{1}{9}|E|$

Proof. *Every element can appear at most 2 times, and in at most 2 triples. Thus, for every selected triple there can be **at most** 6 elements that never get used in another selected triple (2 for every variable that is used in the selected one).*

To make this more clear, if triple (a, b, c) is selected then a can appear in one more triple (two occurrences in total), with two new, unique elements. For example $(a, 1, 2)$. The same is the case for elements b and c. We also notice that for a we can have up to one more triple that is unselected, the same for b and c. Thus, if we select (a, b, c) in the worst case we may not select 3 more triples which in the worst case again, would contain 6 more unique elements.

As an example, consider having the following 4 triples that use 9 elements and only 1 triple can get selected: $\{(a, b, c), (a, 1, 2), (3, b, 4), (5, 6, c)\}$. □

Lemma 6. *Let $\alpha = 150$. For every instance x of Max-3DM-B with $B = 2$ of size $n \geq n_0$ for some constant $n_0 > 0$:*

$$\mathrm{OPT}_{\mathrm{EMP}}(f(x)) \leq \alpha \mathrm{OPT}_{\mathrm{Max-3DM-B}}(x) \ .$$

Proof. *Assume that the optimum of the Max-3DM-B (with $B = 2$) instance has k non-selected triples. Then $OPT_{\mathrm{Max-3DM-B}}(x) \geq n - k$. Now, $f(x)$ will consist of 3 tiles for every triple, and less than or equal to 10 tiles for every element. The most tiles will be produced if the element is in set Y, when we produce $7 + (\lambda + 1)$ tiles given that it appears λ times in total (here, $\lambda = 2$). Thus, the number of tiles is $|T| \leq 3n + 10|E|$. If we consider the optimum of the corresponding EMP instance to be the number of edges in a solution that are not adjacent to the border, and assume that the optimal solution has again k broken edges (we can always achieve this by placing the tiles as seen in Fig. 4), then the optimal solution is at most the number of edges that are not adjacent to the border:*

$$\begin{aligned} OPT_{\mathrm{EMP}}(f(x)) \leq |T|/2 + |T| - 2 - k = 1.5|T| - 2 - k \leq \\ 1.5 \cdot (3n + 10|E|) - 2 - k = 4.5n + 15|E| - 2 - k = 4.5n - k + 15|E| - 2 \leq \\ 3.5n + OPT_{\mathrm{Max-3DM-B}}(x) + 15 \cdot 9 \cdot OPT_{\mathrm{Max-3DM-B}}(x) - 2 \leq \\ 3.5 \cdot 4OPT_{\mathrm{Max-3DM-B}}(x) + 136 \cdot OPT_{\mathrm{Max-3DM-B}}(x) - 2 = \\ 150OPT_{\mathrm{Max-3DM-B}}(x) - 2 \ . \end{aligned}$$ □

Lemma 7. *For $\beta = 1$, and for every solution y to $f(x)$ the following inequality holds:*

$$\begin{aligned} |OPT_{\mathrm{Max-3DM-B}}(x) - c_{\mathrm{Max-3DM-B}}(x, g(y))| \leq \\ \beta |OPT_{\mathrm{EMP}}(f(x)) - c_{\mathrm{EMP}}(f(x), y)| \ . \end{aligned}$$

Proof. $OPT_{\text{EMP}}(f(x)) = \left(\frac{3}{2}|T| - 2\right) - k$ *that is, the optimal solution to problem EMP on instance* $f(x)$ *breaks* k *edges, and a solution* y *to* $f(x)$ *breaks* k' *edges (thus* $c_{\text{EMP}}(f(x), y) = \left(\frac{3}{2}|T| - 2\right) - k'$ *(Note that our instance has dimensions* $2 \times \frac{|T|}{2}$*). Because of Lemma 2,* $OPT_{\text{Max-3DM-B}}(x) = n - k$*. Also, obviously* $c_{\text{Max-3DM-B}}(x, g(y)) = n - k'$*. The above means that there is a positive constant* $\beta = 1$*, such that for every solution* y *to* $f(x)$*,* $n - k - (n - k') \leq \beta\left(\frac{3}{2}|T| - k - 2 - \left(\frac{3}{2}|T| - k' - 2\right)\right) \Rightarrow k' - k \leq \beta(k' - k)$*,* □

As a natural consequence of the definition of the L-reduction, and Lemmas 2, 3, 4, 6 and 7,

Lemma 8. *The reduction described above, is an L-reduction from Max-3DM-B to EMP with* $\alpha = 150$ *and* $\beta = 1$*.*

The following theorem follows naturally,

Theorem 2. *Edge Matching Puzzle is APX-hard, and thus under the P≠NP assumption it does not admit a PTAS.*

3.5 APX-Completeness

Theorem 3. *Edge Matching Puzzle is in APX.*

Proof. Suppose that the optimal solution to a given EMP instance is known and has cost OPT*. We then can construct a graph* G *by representing each tile as a node, and for every matched edge in the optimal solution draw an edge between the corresponding tiles/nodes. Then one can proceed from this graph:*

Initialise an empty list M. While there are edges left in G pick one of them, push it into M, and remove both its endpoints and their adjacent edges from G.

As every tile has 4 *edges, the degree of every node in* G *has to be at most* 4*, so there are at most* 8 *edges removed in every step of the algorithm, and* M *has size at least* $OPT/8$*. Now, the following is a constant factor approximation algorithm for EMP:*

Construct a graph G' by creating a node for every tile, and connecting with edges all the pairs of nodes corresponding to two tiles with at least one edge with the same colour. Find a maximum matching M' of G' using a polynomial time algorithm [13]. Now, for every edge uv in M' match the tiles corresponding to vertices u and v into a pair of tiles. Place the pairs of tiles into the given rectange in a snake fashion: fill in row by row, and if the rows have odd size place the last tile so that it takes one place in the current row and one in the next one. If there are single tiles left over place them arbitrarily in the free space of the rectangle.

The algorithm described above is running in polynomial time. As the matching M' *is maximum,* $|M'| \geq |M|$*, the solution returned by the algorithm has cost at least* $1/8$ *times* OPT*, and it is an* $\Theta(1)$*-approximation algorithm for EMP.* □

As a natural consequence of Theorems 2 and 3,

Theorem 4. *Edge Matching Puzzle is APX-complete.*

3.6 Approximation Lower Bound

This subsection, copes with finding an approximation lower bound for the EMP problem using the following approximation lower bound for Max-3DM-B:

Theorem 5 ([12]). *It is NP-hard to approximate the solution of Max-3DM-B, with exactly two occurrences of every element, to within any constant smaller than* $\delta' = \frac{95}{94}$.

An approximation lower bound for EMP can now be easily derived:

Theorem 6. *It is NP-hard to approximate the solution of EMP to within any constant smaller than* $c' = \frac{14250}{14249}$.

Proof. *It easily follows from Theorems 1 and 5 by using* $\alpha = 150$ *and* $\beta = 1$*:*

$$c' = \frac{1}{1 - \frac{1-(1/\delta')}{\alpha\beta}} = \frac{14250}{14249} \qquad \square$$

4 The Corresponding Minimisation Problem

In this section we study two other optimisation variants of the problem: the absolute error for both the minimisation and the maximisation version, and the approximation ratio of the minimisation version. For the latter, as the optimum solution could have cost 0, to make the problem interesting we introduce an assumption that changes the problem a bit; namely that the optimum solution has exactly k edges broken.

We define a minimisation and a maximisation problem to be *corresponding* when the following holds: For every instance with a fixed rectangle, the sum of the costs of the maximisation and the minimisation version is constant. Formally, there exists some $M(x)$ such that $c_{\max}(x, y) = M(x) - c_{\min}(x, y)$, and $OPT_{P_{\min}} = M(x) - OPT_{P_{\max}}$. In the EMP case, $M(x)$ is the number of edges that instance x has in total (either broken or non-broken).

4.1 Absolute Error

It has been shown that EMP does not admit a PTAS. Here it will also be shown that it cannot be approximated within an absolute error of size $o(n)$ if P≠NP.

Theorem 7. *Any maximisation problem* $P_{\max}$ *with an optimal solution* $OPT_{P_{\max}} = \Omega(n)$*, which can be approximated within an absolute error of* $o(n)$*, admits a PTAS for large enough instances.*

Proof. *Let* $OPT_{P_{\max}} = \Omega(n)$*, and* $f(n) = o(n)$ *be an absolute error within which we can approximate the problem and* $\epsilon = \frac{f(n)}{OPT_{P_{\max}}}$*. Then, by definition, for* *the feasible solution y returned by the approximation algorithm when run on instance x,*

$$c(x,y) \geq OPT_{P_{\max}} - f(n) = OPT_{P_{\max}}\left(1 - \frac{f(n)}{OPT_{P_{\max}}}\right)$$

$$\Rightarrow c(x,y) \geq OPT_{P_{\max}}(1-\epsilon) \overset{\forall \epsilon \leq \frac{1}{2}}{\geq} \frac{OPT_{P_{\max}}}{1+2\epsilon} \Rightarrow 1+2\epsilon \geq \frac{OPT_{P_{\max}}}{c(x,y)}$$

which – as ϵ can be made arbitrarily small for large enough instances – implies a PTAS for $P_{\max}$. □

Note 2. For $P_{\max}$ to have a PTAS, $f \in o(g)$ would be enough.

Theorem 8. *If a minimisation problem $P_{\min}$ can be approximated within an absolute error of $o(n)$ and it has a corresponding maximisation problem with $OPT_{P_{\max}} = \Omega(n)$ then $P_{\max}$ admits a PTAS for large enough instances.*

Proof. Let again $OPT_{P_{\max}} = \Omega(n)$, and $f(n) = o(n)$ be an absolute error within which we can approximate the problem. By definition, we have that

$$c_{\min}(x,y) \leq (M(x) - OPT_{P_{\max}}) + f(n) \Rightarrow M(x) - c_{\min}(x,y) \geq OPT_{P_{\max}} - f(n)$$
$$\Rightarrow c_{\max}(x,y) \geq OPT_{P_{\max}} - f(n)$$

and the rest of the proof is identical with the proof of Theorem 7. □

Theorem 9. *EMP cannot be approximated within an absolute error of size $o(n)$, neither in the minimisation, nor in the maximisation version.*

Proof. That the maximisation version cannot be approximated within an absolute error of size $o(n)$ directly follows from Theorem 7, as it admits no PTAS and has an optimum of size $\Omega(n)$. Now clearly, because of Theorem 8 the corresponding minimimisation version of EMP also cannot be approximated within an absolute error of $o(n)$. □

4.2 Approximation Ratio

In this subsection a result on the approximation ratio of the minimisation version is presented, namely, the minimisation version cannot be approximated within an approximation ratio of $o(n)$.

Theorem 10. *If a minimisation problem $P_{\min}$ with $OPT_{P_{\min}} = \Omega(1)$, can be approximated within an approximation ratio of $o(n)$, then it also can be approximated within an absolute error of $o(n)$.*

Proof. Assume that $P_{\min}$ is a minimisation problem with $OPT_{P_{\min}} = \Omega(1)$ and that A is an approximation algorithm that can approximate it within an approximation ratio of $o(n)$. Then for some $h(n) = o(n)$ being the approximation ratio,

$$\forall x, c_{\min}(x, A(x)) \leq OPT_{P_{\min}} h(n) \Rightarrow c_{\min}(x, A(x)) \leq o(n)$$
$$\Rightarrow c_{\min}(x, A(x)) \leq OPT_{P_{\min}} + f(n)$$

for some function $f = o(n)$. Naturally that $P_{\min}$ can be approximated within an absolute error of $o(n)$. □

Theorem 11. *The minimisation version of EMP cannot be approximated within an approximation ratio of $o(n)$, assuming that $OPT_{\mathrm{EMP}_{\min}} \neq 0$.*

Proof. *For EMP when $OPT_{\mathrm{EMP}_{\min}} \neq 0$ it holds that $OPT_{\mathrm{EMP}_{\min}} = \Omega(1)$. Thus applying Theorems 9 and 10 we get that the minimisation version of EMP cannot be approximated within an approximation ratio of $o(n)$.* □

References

1. Demaine, E.D., Demaine, M.L.: Jigsaw Puzzles, Edge Matching, and Polyonymo Packing: Connections and Complexity. Graphs and Combinatorics 23, 195–208 (2007)
2. Berger, R.: The Undecidability of the Domino Problem. Memoirs of the American Mathematical Society 66 (1966)
3. Garey, M.R., Johnson, D.S., Papadimitriou, C.H.: Computers and Intractability: A Guide to the Theory of NP-completeness, p. 257. W.H. Freeman & Co., New York (1979)
4. Savelsberg, M.P.W., van Emde Boas, P.: BOUNDED TILING, an Alternative to SATISFIABILITY? In: Proceedings 2nd Frege Conference, vol. 20, pp. 354–363. Akademie Verlag, Schwerin (1984)
5. Chlebus, B.S.: Domino-Tiling Games. Journal of Computer and System Sciences 32, 374–392 (1986)
6. Benoist, T., Bourreau, E.: Fast Global Filtering for Eternity II. Constraint Programming Letters 3, 36–49 (2008)
7. Ansótegui, C., Béjar, R., Fernández, C., Mateau, C.: Edge Matching Puzzles as Hard SAT/CSP Benchmarks. In: Stuckey, P.J. (ed.) CP 2008. LNCS, vol. 5202, pp. 560–565. Springer, Heidelberg (2008)
8. Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., Protasi, M.: Complexity and Approximation: Combinatorial Optimization Problems and their Approximability Properties. Springer, Heidelberg (1999)
9. Papadimitriou, C.H., Yannakakis, M.: Optimization, Approximation, and Complexity Classes. Journal of Computer and System Sciences 43, 425–440 (1991)
10. Kann, V.: Maximum Bounded 3-Dimensional Matching is MAX SNP-Complete. Information Processing Letters 37, 27–35 (1991)
11. Chlebìk, M., Chlebìkovà, J.: Approximation Hardness for Small Occurrence Instances of NP-hard Problems. In: Petreschi, R., Persiano, G., Silvestri, R. (eds.) CIAC 2003. LNCS, vol. 2653. Springer, Heidelberg (2003); (also ECCC Report) (2002)
12. Chlebìk, M., Chlebìkovà, J.: Inapproximability Results for Bounded Variants of Optimization Problems. In: Proocedings of FCT 2003: the 14th International Symposium on Fundamentals of Computation Theory (2003)
13. Cormen, T., Leiserson, C., Rivest, R., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press, Cambridge (2001)