Completeness in standard and differential approximation Poly-(D)APX- and (D)PTAS-completeness

Cristina Bazgan Bruno Escoffier Vangelis Th. Paschos

LAMSADE, Université Paris-Dauphine
Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France
{bazgan,escoffier,paschos}@lamsade.dauphine.fr

17th April 2004

Abstract

Several problems are known to be **APX**-, **DAPX**-, **PTAS**-, or **Poly-APX-PB**-complete under suitably defined approximation-preseving reductions. But, to our knowledge, no natural problem is known to be **PTAS**-complete and no problem at all is known to be **Poly-APX**-complete. On the other hand, **DPTAS**- and **Poly-DAPX**-completeness have not been studied until now. We first prove in this paper the existence of natural **Poly-APX**- and **Poly-DAPX**-complete problems under the well known PTAS-reduction and under the DPTAS-reduction (defined in "G. Ausiello, C. Bazgan, M. Demange, and V. Th. Paschos, *Completeness in differential approximation classes*, MFCS'03"), respectively. Next, we deal with **PTAS**- and **DPTAS**-copmpleteness. We introduce approximation-preserving reductions, called FT and DFT, respectively, and prove that, under these new reductions, natural problems are **PTAS**-complete, or **DPTAS**-complete. Then, we deal with the existence of intermediate problems under our reductions and we partially answer this question showing that the existence of **NPO**-intermediate problems under Turing-reduction is a sufficient condition for the existence of intermediate problems under both FT- and DFT-reductions. Finally, we show that is **DAPX**-complete under the DPTAS-reduction.

Keywords: Approximation algorithm; Approximation schema; Combinatorial optimization; Completeness; Complexity; Reduction

1 Introduction

Many **NP**-complete problems are decision versions of natural optimization problems. Since, unless **P** $=$ **NP**, such problems cannot be solved in polynomial time, a major question is to find polynomial algorithms producing solutions "close to the optimum" (in some prespecified sense). Here, we deal with polynomial approximation of **NPO** problems, i.e., for optimization problems the decision versions of which are in **NP**. A polynomial approximation algorithm A for an optimization problem Π is a polynomial time algorithm that produces, for any instance x of Π, a feasible solution $y = \mathtt{A}(x)$. The quality of y is estimated by computing the so-called approximation ratio. Two approximation ratios are commonly used in order to evaluate the approximation capacity of an algorithm: the standard ratio and the differential ratio.

By means of these ratios, **NPO** problems are then classified with respect to their approximability properties. Particularly interesting approximation classes are, for the standard approximation paradigm, the classes **Poly-APX** (the class of the problems approximated within a ratio that is a polynomial of the size of the instance), **APX** (the class of constant-approximable problems), **PTAS** (the class of problems admitting a polynomial time approximation schemata) and **FPTAS** (the class of problems admitting a fully polynomial time approximation schemata). Analogous classes can be defined under the differential

approximation paradigm: **Poly-DAPX**, **DAPX**, **DPTAS** and **DFPTAS** (see section 2 for formal definitions), are the di erential counterparts of **Poly-APX**, **APX**, **PTAS** and **FPTAS**, respectively. Note that **FPTAS** $\subsetneq$ **PTAS** $\subsetneq$ **APX** $\subsetneq$ **Poly-APX**, and **DFPTAS** $\subsetneq$ **DPTAS** $\subsetneq$ **DAPX** $\subsetneq$ **Poly-DAPX**; these inclusions are strict unless $\mathbf{P} = \mathbf{NP}$.

During last two decades, several approximation-preserving reductions have been introduced. Via them, hardness results in several approximability classes have been studied. Consider two classes $\mathbf{C_1}$ and $\mathbf{C_2}$ with $\mathbf{C_1} \subseteq \mathbf{C_2}$, and assume a reduction preserving membership in $\mathbf{C_1}$ (i.e., if Π reduces to Π' and $\Pi' \in \mathbf{C_1}$, then $\Pi \in \mathbf{C_1}$). A problem $\mathbf{C_2}$-complete under this reduction is in $\mathbf{C_1}$ i $\mathbf{C_2} = \mathbf{C_1}$ (as in the case of **NP**-completeness).

Consider, for instance, the P-reduction defined in [5]; this reduction, extended in [3, 6] (and renamed PTAS-reduction), preserves membership in **PTAS**. Natural problems, such as max independent set in bounded degree graphs (called -B in what follows), or , are **APX**-complete under the PTAS-reduction (see, respectively, [13, 14]). This implies that such problems are not in **PTAS** unless $\mathbf{P} = \mathbf{NP}$ (since, as we have mentioned previously, provided that $\mathbf{P} \neq \mathbf{NP}$, **PTAS** $\subsetneq$ **APX**).

In di erential approximation, analogous results have been obtained in [1] for di erential approximation; there, a DPTAS-reduction, preserving membership in **DPTAS**, is defined and natural problems such as are shown to be **DAPX**-complete.

In the same way, the F-reduction of [5], preserves membership in **FPTAS**. Under this reduction, only one (not very natural) problem (derived from -) is known to be **PTAS**-complete (see Appendix A). Despite some restrictive notions of **DPTAS**-hardness presented in [1], no systematic study of **DPTAS**-completeness has been performed until now.

Finally, another well known reduction (and very interesting in what follows in this paper) is the E-reduction ([11]). It preserves inclusion in **FPTAS** and, using it, the existence of **Poly-APX-PB**-complete problems has been shown in [11] (informally, **Poly-APX-PB** is the class of problems of **Poly-APX**the optimal solution-values of which are bounded above by a polynomial of the size of their instances).

Reductions provide a structure in approximation classes, and are very useful in obtaining hardness approximability results. As in the case of **NP**-completeness with the result of [12], one can try to refine the study of this structure by determining if there exist intermediate problems. For two complexity classes $\mathbf{C}$ and $\mathbf{C}'$, $\mathbf{C}' \subseteq \mathbf{C}$, and a reduction R preserving membership in $\mathbf{C}'$, a problem is called $\mathbf{C}$-intermediate, if it is neither $\mathbf{C}$-complete under R, nor in $\mathbf{C}'$. In [5] is proved the existence of **APX**- and **PTAS**-intermediate problems under P- and F-reductions, respectively.

The main results of this paper deal with the existence of complete problems for the following standard and di erential approximation classes:

- **Poly-APX** and **Poly-DAPX** under the PTAS- and DPTAS-reductions, respectively (the first one is defined in [6] while the second one in [1]);
- **FPTAS** and **DFPTAS** under two new reductions called FT and DFT, respectively;

Finally, for reductions FT and DFT, we try to apprehend if they allow existence of intermediate problems and we partially answer this question by proving that such problems do exist provided that there exist intermediate problems in **NPO** under the seminal Turing-reduction.

Let us note that no problem was known to be **Poly-APX**-complete until now, since the results in [11] prove the existence of **Poly-APX-PB**-complete problems. On the other hand, the question about the existence of **Poly-DAPX**-complete problems is not, to our knowledge, posed until now. The existence of **PTAS**-complete problems is proved, as we have just mentioned, by means of a **FPTAS**-preserving reduction (called FT-reduction). It is somewhat weaker than the F-reduction of [5], but it has the merit that natural problems are shown to be **PTAS**-complete under it (while this seems to be not true for the F-reduction). Indeed, we show that, under FT-reduction, any polynomially bounded **NP**-hard problem of **PTAS** is **PTAS**-complete. Next, we propose a reduction preserving membership in **DFPTAS** and show that, under it, natural

problems as , or , both in planar graphs, are **DPTAS**-complete. Here also, we use another notion of polynomial boundness, called diameter polynomial boundness, and show that any diameter polynomially bounded **NP**-hard problem of **DPTAS** is **DPTAS**-complete.

The paper is organized as follows: in Section 2, we recall some basic definitions and present the two new reductions. In Sections 3 and 4, we show **Poly-APX** and **Poly-DAPX**-completeness, respectively. In Sections 5 and 6, we present our completeness results for **PTAS** and **DPTAS**. The results on intermediate problems are given in Section 7. Finally, in Section 8, it is proved that is **DAPX**-complete under DPTAS-reduction. This is the first problem that is **DAPX**-complete but not **APX**-complete. Definitions of problems used and/or discussed in the paper, together with specifications of their worst solutions are given in Appendix A.

2 Preliminaries

2.1 Polynomial approximation

We firstly recall some useful definitions about basic concepts of polynomial approximation.

Definition 1. A problem Π in **NPO** is a quadruple $(\mathcal{I}, \mathrm{Sol}, m, \mathrm{opt})$ where:

- $\mathcal{I}$ is the set of instances (and can be recognized in polynomial time);
- given $x \in \mathcal{I}$, $\mathrm{Sol}(x)$ is the set of feasible solutions of x; the size of a feasible solution of x is polynomial in the size $|x|$ of the instance; moreover, one can determine in polynomial time if a solution is feasible or not;
- Given $x \in \mathcal{I}$ and $y \in \mathrm{Sol}(x)$, $m(x, y)$ denotes the value of the solution y of the instance x; m is called the objective function, and is computable in polynomial time; we suppose here that $m(x, y) \in \mathbb{N}$;
- $\mathrm{opt} \in \{\min, \max\}$; in what follows, we will use notations $\mathrm{opt}(\Pi) = \max$, or $\min$ to denote that Π is a maximization, or a minimization problem, respectively. ∎

Given a problem Π in **NPO**, we distinguish the following three di erent versions of it:

- the constructive version denoted also by Π, where the goal is to determine a solution $y^* \in \mathrm{Sol}(x)$ satisfying $m(x, y^*) = \mathrm{opt}\{v(x, y), y \in \mathrm{Sol}(x)\}$;
- the evaluation problem Π_e, where we are only interested in determining the value of an optimal solution;
- the decision version Π_d of Π where, given an instance x of Π and an integer k, we wish to answer the following question: "does there exist a feasible solution y of x such that $m(x, y) \geqslant k$ if $\mathrm{opt} = \max$, or $m(x, y) \leqslant k$ if $\mathrm{opt} = \min$?".

Given an instance x of an optimization problem Π, let $\mathrm{opt}(x)$ be the value of an optimal solution, and $\omega(x)$ be the value of a worst feasible solution. In other terms, $\omega(x)$ is the optimal value of the same optimization problem with the opposite objective (minimize instead of maximize, and vice-versa) with respect to Π. We now define the two ratios the most commonly used for the analysis of approximation algorithms, called standard and di erential in the sequel.

Let x be an instance of an **NPO** problem Π and let $y \in \mathrm{Sol}(x)$. The *standard approximation ratio* of y is $r(x, y) = m(x, y)/\mathrm{opt}(x)$. The *differential approximation ratio* of y is $\delta(x, y) = |m(x, y) - \omega(x)|/|\mathrm{opt}(x) - \omega(x)|$.

Following the above, standard approximation ratios for minimization problems are greater than, or equal to, 1, while for maximization problems these ratios are smaller than, or equal to 1. On the other hand, di erential approximation ratio is always at most 1 for any problem.

Let λ be a function mapping the instances of a problem Π to $[0,1]$, or to $[1,+\infty)$. An algorithm $\mathtt{A}$ guarantees standard (resp., di erential) ratio g i , for any instance x of Π, $r(x,A(x)) \geqslant \lambda(x)$, or $r(x,A(x)) \leqslant \lambda(x)$, depending whether Π is a maximization or a minimization problem (resp., $\delta(x,A(x)) \geqslant \lambda(x)$). A problem Π is standard (resp., di erential) λ-approximable i there exists a polynomial algorithm that guarantees standard (resp., the di erential) ratio λ.

We can now formally define the approximation classes **Poly-APX**, **APX**, **PTAS** and **FPTAS** with which we deal in this paper. An **NPO** problem Π is in the class:

- **Poly-APX**, i it is approximable within ratios $O(|x|^{\eta})$, for some $\eta \geqslant 0$, if $\mathrm{opt}(\Pi) = \min$, or $\eta \leqslant 0$, if $\mathrm{opt}(\Pi) = \max$;
- **APX**, i it is constant-approximable, i.e., i there exists a polynomial algorithm which guarantees g where g does not depend on the instance;
- **PTAS**, i it admits a polynomial time approximation schema; such a schema is a family of polynomial algorithms $\mathtt{A}_{\varepsilon}$, $\varepsilon \in]0,1]$, any of them guaranteeing approximation ratio $1-\varepsilon$ (if if $\mathrm{opt}(\Pi) = \max$), or $1+\varepsilon$ (if $\mathrm{opt}(\Pi) = \min$);
- **FPTAS**, i it admits a fully polynomial time approximation schema; such a schema is a polynomial time approximation schema $(\mathtt{A}_{\varepsilon})_{\varepsilon \in]0,1]}$, where the complexity of any $\mathtt{A}_{\varepsilon}$ is polynomial in both the size of the instance and in $1/\varepsilon$.

Classes **Poly-DAPX**, **DAPX**, **DPTAS** and **DFPTAS** for the di erential approximation paradigm can be defined analogously (recall that di erential approximation ratio is always less than, or equal to, 1; so, di erential approximation classes are defined analogously to the standard ones for maximization problems).

We now recall what it is called a *polynomially bounded* problem and introduce a notion of diameter boundness, very useful and intuitive when dealing with the di erential approximation paradigm.

Definition 2. An **NPO** problem Π is *polynomially bounded* i there exists a polynomial q such that, for any instance x and for any feasible solution $y \in \mathrm{Sol}(x)$, $m(x,y) \leqslant q(|x|)$. It is *diameter polynomially bounded* i there exists a polynomial q such that, for any instance x, $|\mathrm{opt}(x) - \omega(x)| \leqslant q(|x|)$. ∎

The class of polynomially bounded **NPO** problems will be denoted by **NPO-PB**, while the class of diameter polynomially bounded **NPO** problems will be denoted by **NPO-DPB**. Analogously, for any (standard or di erential) approximation class **C**, we will denote by **C-PB** (resp., **C-DPB**) the subclass of polynomially bounded (resp., diameter polynomially bounded) problems of **C**.

We also need the following definitions introduced in [11] and by means of which the **Poly-APX**-completeness of is proved.

- A problem $\Pi \in$ **NPO** is said *additive*, if there exist an operator $\oplus$ and a function f, both computable in polynomial time, such that:
  - $\oplus$ associates to any pair $(x_1,x_2) \in \mathcal{I}_{\Pi} \times \mathcal{I}_{\Pi}$ an instance $x_1 \oplus x_2 \in \mathcal{I}_{\Pi}$ with $\mathrm{opt}(x_1 \oplus x_2) = \mathrm{opt}(x_1) + \mathrm{opt}(x_2)$;
  - with any solution $y \in \mathrm{sol}_{\Pi}(x_1 \oplus x_2)$, f associates two solutions $y_1 \in \mathrm{sol}_{\Pi}(x_1)$ and $y_2 \in \mathrm{sol}_{\Pi}(x_2)$ such that $m(x_1 \oplus x_2, y) = m(x_1,y_1) + m(x_2,y_2)$.
- Let **Poly** be the set of functions from $\mathbb{N}$ to $\mathbb{N}$ bounded by a polynomial. A function $F : \mathbb{N} \to \mathbb{N}$ is *hard* for **Poly** i for any $f \in$ **Poly**, there exist two constants k and k' such that, there exists n_0 such that $f(n) \leqslant kF(n^c)$.

- A problem $\Pi \in$ **NPO** is *canonically hard for* ***Poly-APX*** i there exist a transformation T from 3 to Π, two constants n_0 and c and a function F, hard for **Poly** such that, given an instance x of 3 on $n \geqslant n_0$ variables and a number $N \geqslant n_0^c$, instance $x' = T(x, N)$ belongs to $\mathcal{I}_\Pi$ and verifies the following properties:
  - if x is satisfiable, then $\mathrm{opt}(x') = N$;
  - if x is not satisfiable, then $\mathrm{opt}(x') \geqslant N/F(n)$;
  - given a solution $y \in \mathrm{sol}_\Pi(x')$ such that $m(x, y') > N/F(n)$, one can polynomially determine a truth assignment satisfying x.

2.2 Reductions

First, let us recall that, given a reduction R and a set **C** of problems, a problem $\Pi \in$ **C** is **C**-complete under R i any problem in **C** R-reduces to Π. If R preserves membership in $\mathbf{C}' \subseteq$ **C**, Π is **C**-intermediate under R i it is neither **C**-complete nor in $\mathbf{C}'$ (provided that **P** $\neq$ **NP**). Moreover, we will say that a problem in **NPO** is **NP**-hard if its decision version is **NP**-complete.

Five basic and two new reductions will be used in this paper. Among the former, the first one is the seminal Turing-reduction between optimization problems as it appears in ([9]). It preserves optimality of solutions and hence membership in **PO** (the optimization problems, the decision versions of which are in **P**; obviously, **PO** $\subseteq$ **NPO**). Let Π and Π' be two problems in **NPO**. Then, Π reduces to Π' under Turing-reduction (denoted by $\Pi \leq_{\mathsf{T}} \Pi'$) i , given an oracle $\square$ optimally solving Π', we can devise an algorithm optimally solving Π, in polynomial time if $\square$ is polynomial.

The other four basic reductions, PTAS, E, DPTAS and F that will be discussed or used in what follows. They are defined in [6, 11, 1, 5], respectively, and revisited here for reasons of readability..

Let Π and Π' be two maximization **NPO**-problems (the case of minimization is completely analogous). Then, Π *PTAS-reduces to* Π' (denoted by $\Pi \leq_{\mathsf{PTAS}} \Pi'$), i there exist theree function f, g and c such that:

- for any $x \in \mathcal{I}_\Pi$ and any $\varepsilon \in]0,1[$, $f(x,\varepsilon) \in \mathcal{I}_{\Pi'}$; f is computable in time polynomial with $|x|$;
- for any $x \in \mathcal{I}_\Pi$, any $\varepsilon \in]0,1[$ and any $y \in \mathrm{sol}_{\Pi'}(f(x,\varepsilon))$, $g(x,y,\varepsilon) \in \mathrm{sol}_\Pi(x)$; g is computable in time polynomial with $|x|$ and $|y|$;
- $c :]0,1[\to]0,1[$;
- for any $x \in \mathcal{I}_\Pi$ and any $\varepsilon \in]0,1[$, $r_{\Pi'}(f(x,\varepsilon), S) \geqslant 1 - c(\varepsilon) \Longrightarrow r_\Pi(x, g(x,y,\varepsilon)) \geqslant 1 - \varepsilon$

PTAS-reduction preserves inclusion in **PTAS**. Furthermore, it is the strictest reduction under which natural problems as -B, or -B are shown **APX**-complete.

As we have already mentioned, the E-reduction has been defined in [11] in an attempt to be applied uniformly at all levels of approximability. It is slightly weaker than the L-reduction of [13] and preserves inclusion in **FPTAS**. We say that a *problem* Π *E-reduces to* Π' ($\Pi \leq_{\mathsf{E}} \Pi'$) i there exist two polynomially computable functions f and g and a constant c such that:

- for any $x \in \mathcal{I}_\Pi$, $f(x) \in \mathcal{I}_{\Pi'}$; moreover, there exist a polynomial p such that $\mathrm{opt}(x) \leqslant p(|x|)\,\mathrm{opt}(f(x))$;
- for any $x \in \mathcal{I}_\Pi$ and any $y \in \mathrm{sol}_{\Pi'}(f(x))$, $g(x) \in \mathrm{sol}_\Pi(x)$; furthermore, $\epsilon(x, g(y)) \leqslant \beta\epsilon(f(x), y)$ where for $I \in \mathcal{I}_\Pi$ and $S \in \mathrm{sol}_\Pi(I)$, $\epsilon(I,S) = r(I,S) - 1$, if $\mathrm{opt}(\Pi) = \min$ and $\epsilon(I,S) = 1 - r(I,S)$, if $\mathrm{opt}(\Pi) = \max$.

As it is proved in [11], is additive and canonically hard for **Poly-APX**. Furthermore, if a problem Π is additive and canonically hard for **Poly-APX**, then any problem in **Poly-APX-PB** E-reduces to Π. Consequently, *is* ***Poly-APX-PB****-complete, under the E-reduction.*

The DPTAS-reduction has been introduced in [1] in order to provide **DAPX**-completeness results. It preserves membership in **DPTAS**. Consider two **NPO** problems Π and Π'. Then, $\Pi \leq_{\mathsf{DPTAS}} \Pi'$ if there exist three functions f, g and c, computable in polynomial time, such that:

- $\forall x \in \mathcal{I}_\Pi$, $\forall \varepsilon \in]0,1[\cap\mathbb{Q}$, $f(x,\varepsilon) \in \mathcal{I}_{\Pi'}$; f is possibly multi-valued;
- $\forall x \in \mathcal{I}_\Pi$, $\forall \varepsilon \in]0,1[\cap\mathbb{Q}$, $\forall y \in \mathrm{sol}_{\Pi'}(f(x,\varepsilon))$, $g(x,y,\varepsilon) \in \mathrm{sol}_\Pi(x)$;
- $c :]0,1[\cap\mathbb{Q} \rightarrow]0,1[\cap\mathbb{Q}$;
- $\forall x \in \mathcal{I}_\Pi$, $\forall \varepsilon \in]0,1[\cap\mathbb{Q}$, $\forall y \in \mathrm{sol}_{\Pi'}(f(x,\varepsilon))$, $\delta_{\Pi'}(f(x,\varepsilon),y) \geqslant 1 - c(\varepsilon) \Rightarrow \delta_\Pi(x, g(x,y,\varepsilon)) \geqslant 1-\varepsilon$; if f is multi-valued, i.e., $f = (f_1,\ldots,f_i)$, for some i polynomial in $|x|$, then, the former implication becomes: $\forall x \in \mathcal{I}_\Pi$, $\forall \varepsilon \in]0,1[\cap\mathbb{Q}$, $\forall y \in \mathrm{sol}_{\Pi'}((f_1,\ldots,f_i)(x,\varepsilon))$, $\exists j \leqslant i$ such that $\delta_{\Pi'}(f_j(x,\varepsilon),y) \geqslant 1 - c(\varepsilon) \Rightarrow \delta_\Pi(x, g(x,y,\varepsilon)) \geqslant 1-\varepsilon$. ∎

One of the basic features of di erential approximation ratio is that it is stable uder a ne transformations of the objective functions of the problems dealt. In this sense, problems for which the objective functions of the ones are a ne transformations of the objective functions of the others, are approximate equivalent for the di erential approximation paradigm (this is absolutely not the case for standard paradigm). The most notorious case of such problems is the pair and . A ne transformation is nothing else than a kind of reduction, denoted by AF, in what follows. Two problems Π and Π' are a ne equivalent if $\Pi \leq_{\mathsf{AF}} \Pi'$ and $\Pi' \leq_{\mathsf{AF}} \Pi$. Obviously a ne transformation is a DPTAS-reduction.

Finally, the F-reduction has been introduced in [5] and, as the E-reduction, preserves membership in **FPTAS**. Consider two **NPO** problems Π and Π', Π F-reduces to Π' il and only if there exist three polynomially computable functions f, g and c such that:

- $\forall x \in \mathcal{I}_\Pi$, $f(x) \in \mathcal{I}_{\Pi'}$;
- $\forall I \in \mathcal{I}_\Pi$, $\forall y \in \mathrm{Sol}_{\Pi'}(f(x))$, $g(x,y) \in \mathrm{Sol}_\Pi(x)$;
- $c : \mathcal{I}_\Pi \times (]0,1[\cap\mathbb{Q}) \rightarrow]0,1[\cap\mathbb{Q}$; for all $\varepsilon > 0$, there exists a polynomial p such that $c(x,\varepsilon) = 1/p(|x|,1/\varepsilon)$; moreover, $\forall x \in \mathcal{I}_\Pi$, $\forall \varepsilon \in]0,1[\cap\mathbb{Q}$, $\forall y \in \mathrm{Sol}_{\Pi'}(f(x))$, $\epsilon(f(x),y) \leqslant c(x,\varepsilon) \Rightarrow \epsilon(x, g(x,y)) \leqslant \varepsilon$.

We now introduce two new reductions, denoted by FT and DFT, preserving membership in **FPTAS** and **DFPTAS**, respectively.

Let Π and Π' be two **NP** maximization problems. Let $\square_\alpha^{\Pi'}$ be an oracle for Π' producing, for any $\alpha \in]0,1]$ and for any instance x' of Π', a feasible solution $\square_\alpha^{\Pi'}(x')$ of x' that is an $(1-\alpha)$-approximation for the standard ratio.

Definition 3. Π FT-reduces to Π' (denoted by $\Pi \leq_{\mathsf{FT}} \Pi'$) i , for any $\varepsilon > 0$, there exists an algorithm $\mathtt{A}_\varepsilon(x, \square_\alpha^{\Pi'})$ such that:

- for any instance x of Π, $\mathtt{A}_\varepsilon$ returns a feasible solution which is a $(1-\varepsilon)$-standard approximation;
- if $\square_\alpha^{\Pi'}(x')$ works in time polynomial in both $|x'|$ and $1/\alpha$, then $\mathtt{A}_\varepsilon$ is polynomial in both $|x|$ and $1/\varepsilon$. ∎

For the case where at least one among Π and Π' is a minimization problem it su ces to replace $1-\varepsilon$ or/and $1-\alpha$ by $1+\varepsilon$ or/and $1+\alpha$, respectively. Reduction DFT, dealing with di erential approximation, can be defined analogously.

Clearly, FT- (resp., DFT) reduction transforms a fully polynomial time approximation schema for Π' into a fully polynomial time approximation schema for Π, i.e., it preserves membership in **FPTAS** (resp., **DFPTAS**). Observe also that AF-reduction, mentioned above is also a DFT-reduction.

The F-reduction is a special case of FT-reduction since this latter one explicitely allows multiple calls to oracle $\square$ (this fact is not explicit in F-reduction; in other words, it is not clearly mentioned if f and g are allowed to be multivalued). Also, FT-reduction seems allowing more freedom in the way Π is transformed to Π'; for instance, in F-reduction, function g transforms an optimal solution for Π' into an optimal solution for Π, i.e., F-reduction preserves optimality; this is not the case for FT-reduction. This freedom will allow us in reducing non polynomially bounded **NPO** problems to **NPO-PB** ones. It seems so that the latter reduction is larger than the former one but this fact remains to be confirmed and such proof does not seem to be trivial and is not considered in this paper.

In what follows, given a class $\mathbf{C} \subseteq \mathbf{NPO}$ and a reduction R, we denote by $\overline{\mathbf{C}}^{\mathsf{R}}$ the closure of $\mathbf{C}$ under R, i.e., the set of problems in **NPO** that R-reduce to some problem in $\mathbf{C}$.

3 Poly-APX-completeness

The nature of the E-reduction does not allow transformation of a non-polynomially bounded problem into a polynomially bounded one. In order to extend completeness in the whole **Poly-APX** we have to use a larger (less restrictive) reduction than E. In what follows in this section we show that PTAS-reduction allows us to prove the existence of complete problems for **Poly-APX**. The basic result of this section is the following theorem.

Theorem 1. *If $\Pi \in$ **NPO** is additive and canonically hard for **Poly-APX**, then any problem in **Poly-APX** PTAS-reduces to Π.*

Proof. Let Π' be a maximization problem of **Poly-APX** and let A be an approximation algorithm for Π achieving approximation ratio $1/c(\cdot)$, where $c \in$ **Poly** (the case of minimization will be dealt later in Remark 1). Let Π an additive problem that is canonically hard for **Poly-APX**, let F be a function hard for **Poly** and let k and c' be such that (for $n \geqslant n_0$, for a certain value n_0) $nc(n) \leqslant k(F(n^{c'}) - 1)$. Let, finally, $x \in \mathcal{I}_{\Pi'}$, $\varepsilon \in]0,1[$ and $n = |x|$.

Construction of $f(x, \varepsilon)$

Let $m \geqslant \mathrm{opt}_{\Pi'}(x)/c(n)$. If we try to reproduce identically the analogous proof of [11] we would be faced to the problem that quantity $mc(n)$ is not always polynomially bounded; in other words, transformation f might be not-polynomial. In order to remedy to this, we will uniformly partition the interval $[0, mc(n)]$ of possible values for $\mathrm{opt}_{\Pi'}(x)$ into $q(n) = 2c(n)/\varepsilon$ sub-intervals (remark that q is a polynomial). Consider, for $i \in \{1, \ldots, q(n)\}$, the set of instances $\mathcal{I}_i = \{x : \mathrm{opt}_{\Pi'}(x) \geqslant imc(n)/q(n)\}$.

Set $N = n^{c'}$. We construct, for any i, an instance χ_i of Π of size N such that:

- if $x \in \mathcal{I}_i$, then $\mathrm{opt}_{\Pi}(\chi_i) = N$;
- otherwise, $\mathrm{opt}_{\Pi}(\chi_i) = N/F(n)$.

Define $f(x, \varepsilon) = \chi = \oplus_{1 \leqslant i \leqslant q(n)} \chi_i$ and observe that $c(n)/q(n) = \varepsilon/2$. Then,

$$\mathrm{opt}_{\Pi}(\chi) = N \times \left| \left\{ i : \mathrm{opt}_{\Pi'}(x) \geqslant \frac{im\varepsilon}{2} \right\} \right| + \frac{N}{F(n)} \left(q(n) - \left| \left\{ i : \mathrm{opt}_{\Pi'}(x) \geqslant \frac{im\varepsilon}{2} \right\} \right| \right) \qquad (1)$$

Construction of $g(x, y, \varepsilon)$

Let y be a solution of χ and let j be the largest i for which $m(\chi_i, y) > N/F(n)$. Then, one can compute a solution ψ' of x such that:

$$m\left(x, \psi'\right) \geqslant jm\frac{\varepsilon}{2} \qquad (2)$$

Furthermore, we have obviously using (1):

$$m(\chi, y) \leqslant Nj + (q(n) - j)\,\frac{N}{F(N)} \tag{3}$$

We define $\psi = g(x, y, \varepsilon) = \operatorname{argmax}\{m(x, \psi'), m(x, \mathtt{A}(x))\}$. Note that $m(x, \psi) \geqslant \max\{m, jm\varepsilon/2\}$.

Transfert of approximation ratios

Using (1) and (3), we get:

$$\begin{aligned} r(\chi, y) &\leqslant \frac{j\left(N - \frac{N}{F(N)}\right) + \frac{Nq(n)}{F(N)}}{\left(\left|\left\{i : \operatorname{opt}_{\Pi'}(x) \geqslant \frac{im\varepsilon}{2}\right\}\right|\right)\left(N - \frac{N}{F(N)}\right) + \frac{Nq(n)}{F(N)}} \\ &= \frac{j + \frac{q(n)}{F(n)-1}}{\left|\left\{i : \operatorname{opt}_{\Pi'}(x) \geqslant \frac{im\varepsilon}{2}\right\}\right| + \frac{q(n)}{F(n)-1}} \end{aligned} \tag{4}$$

Since $q(n) = 2c(n)/\varepsilon \leqslant (2k(F(n) - 1))/(\varepsilon n)$, we obtain from (4):

$$r(\chi, y) \leqslant \frac{j + \frac{2k}{n\varepsilon}}{\left|\left\{i : \operatorname{opt}_{\Pi'}(x) \geqslant \frac{im\varepsilon}{2}\right\}\right| + \frac{2k}{n\varepsilon}} \leqslant \frac{j + \frac{2k}{n\varepsilon}}{\frac{2\operatorname{opt}_{\Pi'}(x)}{m\varepsilon} - 1 + \frac{2k}{n\varepsilon}} \leqslant \frac{j + \frac{2k}{n\varepsilon}}{\frac{2\operatorname{opt}_{\Pi'}(x)}{m\varepsilon} - 1} \tag{5}$$

We now consider two cases, namely, $j \leqslant 2/\varepsilon$ and $j \geqslant 2/\epsilon$.

When $j \leqslant 2/\varepsilon$, taking into account that $r_(x, \psi) = m/\operatorname{opt}_{\Pi'}(x)$, we have from (5):

$$r(\chi, y) \leqslant \frac{1 + \frac{k}{n}}{\frac{1}{r_(x,\psi)} - \frac{\varepsilon}{2}} \leqslant \frac{r_(x, \psi)\left(1 + \frac{k}{n}\right)}{1 - \frac{\varepsilon}{2}} \tag{6}$$

For case $j \geqslant 2/\epsilon$, observing that, from (2), $r(x, \psi) \geqslant jm\varepsilon/(2\operatorname{opt}_{\Pi'}(x))$, we get from (5):

$$r(\chi, y) \leqslant \frac{j + \frac{2k}{\varepsilon n}}{\frac{2\operatorname{opt}_{\Pi'}(x)}{m\varepsilon} - 1} \leqslant \frac{j\left(1 + \frac{k}{n}\right)}{\frac{j}{r(x,\psi)} - 1} \leqslant \frac{r_(x, \psi)\left(1 + \frac{k}{n}\right)}{1 - \frac{\varepsilon}{2}} \tag{7}$$

Assuming $n \geqslant 4k/\varepsilon$ (otherwise, Π' can be solved in polynomial time in x) and combining (6) and (7), we finally get for ε su ciently small:

$$r(x, \psi) \geqslant r(\chi, y)\frac{1 - \frac{\varepsilon}{2}}{1 + \frac{\epsilon}{4}} \geqslant r(\chi, y)\left(1 - \frac{\varepsilon}{2}\right)\left(1 - \frac{\varepsilon}{4}\right) \geqslant r(\chi, y)\left(1 - \frac{3\varepsilon}{4}\right)$$

In other words, reduction just described is a PTAS-reduction with $c(\varepsilon) = \varepsilon/(4 - 3\varepsilon)$. The proof of the theorem is complete. ∎

Remark 1. For the case where the problem Π' in the proof of Theorem 1 is a minimization problem, one can reduce it to a maximization problem (for instance using the E-reduction of [11], p. 12) and then one can use the reduction of Theorem 1. Since the composition of an E- and a PTAS-reduction is a PTAS-reduction, the result of Theorem 1 applies also for minimization problems. ∎

Combination of Theorem 1, Remark 1 and of the fact that *is additive and canonically hard for* ***Poly-APX*** ([11]), produces the following concluding theorem.

Theorem 2. *is* ***Poly-APX****-complete under PTAS-reduction.*

4 Poly-APX-completeness under the differential paradigm

We now deal with the existence of **Poly-DAPX**-complete problems. This section consists of two parts. The former is about **Poly-DAPX-PB**-completeness, while the latter one extends this completeness to the whole **Poly-DAPX**.

4.1 Poly-DAPX-PB-completeness

The main result of this section is the following theorem proving a su cient condition that a problem is **Poly-DAPX-PB**-hard.

Theorem 3. *If a problem* $\Pi \in$ ***NPO*** *is additive and canonically hard for* ***Poly-APX****, then any problem in* ***Poly-DAPX-PB*** $DPTAS$*-reduces to* Π.

Proof. Let Π be an additive problem, canonically hard for **Poly-APX**, for some function F hard for **Poly**. Let $\Pi' \in$ **Poly-DAPX-PB** be a maximization problem, let $\mathtt{A}$ be an approximation algorithm for Π' achieving di erential approximation ratio $1/c(\cdot)$, where $c \in$ **Poly**. Let finally x be an instance of Π' of size n.

Consider the set of instances $\mathcal{I}_i = \{x \in \mathcal{I}_{\Pi'} : \mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x) \geqslant i\}$, $i = 1, \ldots, mc(n)$, where $m = m_{\Pi'}(x, \mathtt{A}(x))$. Obviously, $\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x) \leqslant c(n)(m - \omega_{\Pi'}(x)) \leqslant mc(n)$. Since Π' is polynomially bounded, one can determine a polynomial p such that $m \leqslant p(n)$. Let k and c' be such that (for $n \geqslant n_0$, for some n_0) $nmc(n) \leqslant np(n)c(n) \leqslant k(F(n^{c'}) - 1)$.

Construction of $f(x, \varepsilon)$

Set $N = n^{c'}$. One can build, for any i, an instance χ_i of Π such that, if $x \in \mathcal{I}_i$, then $\mathrm{opt}_{\Pi}(\chi_i) = N$, otherwise, $\mathrm{opt}_{\Pi}(\chi_i) = N/F(N)$. We define $f(x, \varepsilon) = \chi = \oplus_{1 \leqslant i \leqslant mc(n)} \chi'_i$. Then,

$$\mathrm{opt}_{\Pi}(\chi) = N\left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right) + \frac{N}{F(N)}\left(mc(n) - \left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right)\right) \tag{8}$$

Construction of $g(x, y, \varepsilon)$

Let y be a solution of χ and j the largest i such that $m(\chi_i, y') \geqslant N/F(N)$. Then, one can determine two solutions y_1 and y_2 of x such that

$$m\left(x, \psi_1\right) - m\left(x, \psi_2\right) \geqslant j \tag{9}$$

Furthermore, the following holds:

$$m\left(\chi, y'\right) \leqslant Nj + (mc(n) - j)\frac{N}{F(N)} \tag{10}$$

Define $\psi = g(x, y, \varepsilon) = \mathrm{argmax}\{m(x, \psi_1), m(x, \mathtt{A}(x))\}$ and note that (from (9) and assuming $\omega_{\Pi'}(x) \neq \mathrm{opt}_{\Pi'}(x)$, otherwise Π' is polynomial on x) $m(x, \psi) - \omega_{\Pi'}(x) \geqslant \max\{j, 1\}$.

Transfert of differential ratios

Taking into account that $\omega_{\Pi}(\chi) \geqslant 0$ and using (8) and (10), we get:

$$\begin{aligned}
\delta\left(\chi, y\right) &\leqslant \frac{Nj + (mc(n) - j)\frac{N}{F(N)}}{N\left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right) + \frac{N}{F(N)}\left(mc(n) - \left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right)\right)} \\
&\leqslant \frac{j + \frac{mc(n)}{F(N)-1}}{\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x) + \frac{mc(n)}{F(N)-1}} \leqslant \frac{m(x, \psi) - \omega_{\Pi'}(x) + \frac{mc(n)}{F(N)-1}}{\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x) + \frac{mc(n)}{F(N)-1}}
\end{aligned} \tag{11}$$

Using (11), we immediately get:

$$m(x,\psi) - \omega_{\Pi'}(x) \geqslant \delta(\chi,y)\left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right) - \left(\left(1-\delta(\chi,y)\right)\frac{mc(n)}{F(N)-1}\right) \tag{12}$$

Taking into account that $nmc(n) \leqslant k(F(N)-1)$ and that $m(x,\psi) - \omega_{\Pi'}(x) \geqslant 1$, we obtain from (12):

$$\begin{aligned} m(x,\psi) - \omega_{\Pi'}(x) &\geqslant \delta(\chi,y)\left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right) - \left(\left(1-\delta(\chi,y)\right)\frac{k}{n}\right) \\ &\geqslant \delta(\chi,y)\left(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)\right) \\ &\quad - \left(\left(1-\delta(\chi,y)\right)\frac{k}{n}\right)\left(m(x,\psi) - \omega_{\Pi'}(x)\right) \end{aligned} \tag{13}$$

Dividing both members of (13) by $\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)$, we get:

$$\delta(x,\psi) \geqslant \delta(\chi,y) - \left(\left(1-\delta(\chi,y)\right)\frac{k\delta(x,\psi)}{n}\right) \geqslant \delta(\chi,y) - \frac{k}{n}\delta(x,\psi) \geqslant \frac{\delta(\chi,y)}{1+\frac{k}{n}} \tag{14}$$

Assume now, without loss of generality, that $n \geqslant k/\varepsilon$. Then, taking $c(\varepsilon) = \varepsilon^2$, we deduce, using (14) that $\delta(\chi,y) \geqslant 1 - c(\varepsilon)$ implies $\delta(x,\psi) \geqslant (1-\varepsilon^2)/(1+\varepsilon) = 1-\varepsilon$. Therefore, the reduction just described is a DPTAS-reduction and the proof of the theorem is complete. ∎

4.2 Poly-DAPX-completeness

We now generalize Theorem 3 to the whole **Poly-DAPX** by proving the following theorem.

Theorem 4. *If a problem* $\Pi \in$ ***NPO*** *is additive and canonically hard for* ***Poly-APX****, then any problem in* ***Poly-DAPX*** *DPTAS-reduces to* Π.

Proof. Let Π be an additive problem, canonically hard for **Poly-APX**, for some function F hard for **Poly**. Let $\Pi' \in$ **Poly-DAPX** be a maximization problem, let $\mathtt{A}$ be an approximation algorithm for Π' achieving di erential approximation ratio $1/c(\cdot)$, where $c \in$ **Poly**. Let finally x be an instance of Π' of size n. As for the previously, we cannot directly use the proof of Theorem 3 because quantity $m(x,\mathtt{A}(x))$ may be non-polynomially bounded.

We will use the central idea of [1] (see also [**?**] for more details). We will define a set $\Pi'_{i,l}$ of problems derived from Π'. For any pair (i,l), $\Pi'_{i,l}$ has the same set of instances and the same solution-set as Π'; for any instance x and any solution y of x,

$$m_{i,l}(x,y) = \max\left\{0, \left\lfloor \frac{m(x,y)}{2^i} \right\rfloor - l\right\}$$

Note that, for some pairs (i,l), $\Pi'_{i,l}$ may be not in **Poly-DAPX**; hence, use of an algorithm for Π' (supposed to be in **Poly-DAPX**) may be not possible for $\Pi'_{i,l}$. Next, considering x as instance of any of the problems $\Pi'_{i,l}$, we will build an instance $\chi_{i,l}$ of Π, obtaining so a multi-valued function f. Our central objective is, informally, to determine a set of pairs (i,l) such that we will be able to build a "good" solution for Π' using "good" solutions of $\chi_{i,l}$.

Let $\varepsilon \in]0,1[$; set $M_\varepsilon = 1 + \lfloor 2/\varepsilon \rfloor$ and let c' and k be such that (for $n \geqslant n_0$ for some n_0) $nc(n)M_\varepsilon 2/\varepsilon \leqslant k(F(n^{c'})-1)$ (both c' and k may depend on ε). Assume finally without loss of generality that $n \geqslant k$ and set $N = n^{c'}$. Then, $M_\varepsilon/(F(N)-1) \leqslant 1/c(n)$ and $M_\varepsilon/(F(N)-1) \leqslant \varepsilon/2$. Set $m = m(x,\mathtt{A}(x))$ and let

$$\begin{aligned} \mathcal{F} &= \left\{(0, m+k) : k \in \left\{-\left\lfloor\frac{2}{\varepsilon}\right\rfloor, \ldots, \left\lfloor\frac{2}{\varepsilon}\right\rfloor\right\}\right\} \\ &\quad \bigcup \left\{(i, m_{i,0}(x,\mathtt{A}(x)) + k) : i \in \{1,\ldots,p(n)\}, k \in \left\{-\left\lfloor\frac{2}{\varepsilon}+1\right\rfloor, \ldots, \left\lfloor\frac{2}{\varepsilon}+1\right\rfloor\right\}\right\} \end{aligned}$$

Remark that the size of the family $\mathcal{F}$ is polynomial with n.

The following two lemmata are proved in [**?**].

Lemma 1. *([?]) There exists a pair $(i_0, l_0) \in \mathcal{F}$ such that:*

- *either $i_0 = 0$, $\varepsilon(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)) \leqslant 2$ and $l_0 = \omega_{\Pi'}(x)$;*
- *or $i_0 \geqslant 1$, $2^{i_0} \leqslant \varepsilon(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)) \leqslant 2^{i_0+1}$ and $l_0 = \lfloor \omega_{\Pi'}(x)/2^{i_0} \rfloor$.*

Let a pair (i_0, l_0) be as in Lemma 1; then, $\omega_{i_0,l_0}(x) = 0$. Furthermore, taking into account that

1. $\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x) > 0$,
2. if $i_0 = 0$, then $\mathrm{opt}_{i_0,l_0}(x) - \omega_{i_0,l_0}(x) = \mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)$ and
3. if $i_0 \geqslant 1$, then
$$\frac{\mathrm{opt}_{\Pi'}(x)}{2^{i_0}} \geqslant \frac{\frac{2^{i_0}}{\varepsilon} + \omega_{\Pi'}(x)}{2^{i_0}} \geqslant 1 + \frac{\omega_{\Pi'}(x)}{2^{i_0}}$$

one can assume in what follow that $\mathrm{opt}_{i_0,l_0} > 0$.

Lemma 2. *([?]) If (i, l) is such that $2^i \leqslant \varepsilon(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)) \leqslant 2^{i+1}$ and $l = \lfloor \omega_{\Pi'}(x)/2^i \rfloor$, then $\delta_{i,l}(x, y) \geqslant 1 - \varepsilon$ implies $\delta(x, y) \geqslant 1 - 3\varepsilon$. This remains true even if $i = 0$, $\varepsilon(\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)) \geqslant 2$ and $l = \omega_{\Pi'}(x)$.*

We are well-prepared now to specify the reduction claimed in the statement of the theorem.

Construction of $f(x, \varepsilon)$

Let q be an integer. Consider, for any pair $(i, l) \in \mathcal{F}$, the set of instances $\mathcal{I}^q_{i,l} = \{x \in \mathcal{I}_{\Pi'_{i,l}} : \mathrm{opt}_{i,l}(x) \geqslant q\}$. More precisely, consider these instance-sets for $q \in \{0, \ldots, M_\varepsilon\}$. For any pair $(i, l) \in \mathcal{F}$, one can build an instance $\chi_{i,l}$ of Π such that:

$$\begin{aligned} \mathrm{opt}_\Pi(\chi_{i,l}) &= N\left(\left|\left\{q \in \{1, \ldots, M_\varepsilon\} : \mathrm{opt}_{i,l}(x) \geqslant q\right\}\right|\right) \\ &\quad + \frac{N}{F(N)}\left(M_\varepsilon - \left|\left\{q \in \{1, \ldots, M_\varepsilon\} : \mathrm{opt}_{i,l}(x) \geqslant q\right\}\right|\right) \end{aligned} \tag{15}$$

In other words, we just have defined the function $f : f(x, \varepsilon) = (\chi_{i,l}, (i, l) \in \mathcal{F})$.

Construction of $g(x, y, \varepsilon)$

Given a solution $y_{i,l}$ of $\chi_{i,l}$ with value $m(\chi_{i,l}, y_{i,l})$, one can determine a solution $\psi_{i,l}$ of x (instance of $\Pi'_{i,l}$) with value j, where j is such that:

$$m(\chi_{i,l}, y_{i,l}) \leqslant jN + \left((M_\varepsilon - j)\frac{N}{F(N)}\right) \tag{16}$$

For x (instance of Π'), we consider solution $\psi = g(x, y, \varepsilon) = \mathrm{argmax}\{\mathtt{A}(x), y_{i,l} : (i, l) \in \mathcal{F}\}$.

Transfert of differential ratios

Consider a pair (i_0, l_0) verifying one of the two items of Lemma 1. Then,

$$\mathrm{opt}_{i_0,l_0}(x) \leqslant \left\lfloor \frac{\mathrm{opt}_{\Pi'}(x) - \omega_{\Pi'}(x)}{2^{i_0}} + 1 \right\rfloor \leqslant M_\varepsilon \tag{17}$$

From (15), we get:

$$\mathrm{opt}\left(\chi i_0, l_0\right) = N\mathrm{opt}_{i_0,l_0}(x) + \frac{N}{F(N)}\left(M_\varepsilon - \mathrm{opt}_{i_0,l_0}(x)\right) \tag{18}$$

Since, as we have seen above, one can build a solution ψ_0 of value j for x (instance of Π'_{i_0,l_0}), one can deduce (using also (16) that:

$$\begin{aligned} \delta\left(\chi_{i_0,l_0}, y\right) &\leqslant \frac{j\left(N - \frac{N}{F(N)}\right) + \frac{NM_\varepsilon}{F(N)}}{\mathrm{opt}_{i_0,l_0}(x)\left(N - \frac{N}{F(N)}\right) + \frac{NM_\varepsilon}{F(N)}} = \frac{j + \frac{M_\varepsilon}{F(N)-1}}{\mathrm{opt}_{i_0,l_0}(x) + \frac{M_\varepsilon}{F(N)-1}} \\ &\leqslant \frac{j + \frac{M_\varepsilon}{F(N)-1}}{\mathrm{opt}_{i_0,l_0}(x)} \end{aligned} \tag{19}$$

We distinguish the following two cases, namely $j \neq 0$ and $j = 0$.

- If $j \neq 0$, taking into account that $\delta(x, \psi_0) \geqslant j/\mathrm{opt}_{i_0,l_0}(x)$, (19) gives:

$$\begin{aligned} \delta\left(\chi_{i_0,l_0}, y\right) &\leqslant \delta_{i_0,l_0}\left(x, \psi_0\right) + \frac{M_\varepsilon}{\mathrm{opt}_{i_0,l_0}(x)(F(N)-1)} \leqslant \delta_{i_0,l_0}\left(x, \psi_0\right) + \frac{\varepsilon}{2\mathrm{opt}_{i_0,l_0}(x)} \\ &\leqslant \delta_{i_0,l_0}\left(x, \psi_0\right) + \frac{\varepsilon}{2} \end{aligned} \tag{20}$$

  From (20), the following can be deduced using also Lemma 2 (for legibility, di erential ratios are sub-indexed by the corresponding problems):

$$\begin{aligned} \delta_\Pi\left(\chi_{i_0,l_0}, y\right) \geqslant 1 - \frac{\varepsilon}{2} &\implies \delta_{i_0,l_0}\left(x, \psi_0\right) \geqslant 1 - \varepsilon \implies \delta_{\Pi'}\left(x, \psi_0\right) \geqslant 1 - 3\varepsilon \\ &\implies \delta_{\Pi'}\left(x, \psi\right) \geqslant 1 - 3\varepsilon \end{aligned} \tag{21}$$

- If $j = 0$, taking into account that $\delta(x, \psi)1/c(n)$, (19) gives:

$$\delta\left(\chi_{i_0,l_0}, y\right) \leqslant \frac{M_\varepsilon}{\mathrm{opt}_{i_0,l_0}(x)(F(N)-1)} \leqslant \frac{M_\varepsilon}{F(N)-1} \leqslant \frac{1}{c(n)} \leqslant \delta(x, \psi) \tag{22}$$

  Using (22), we immediately get:

$$\delta\left(\chi_{i_0,l_0}, y\right) \geqslant 1 - \frac{\varepsilon}{2} \Longrightarrow \delta(x, \psi) \geqslant 1 - \frac{\varepsilon}{2} \geqslant 1 - 3\varepsilon \tag{23}$$

From (21) and (23), one can immediately conclude that, considering $\varepsilon' = \varepsilon/3$ and $c(\varepsilon') = \varepsilon/6$, the reduction just described is a DPTAS-reduction, completing so the proof of the theorem. ∎

Using the fact that is additive and canonically hard for **Poly-APX**, Theorem 4 directly exhibits the existence of a **Poly-DAPX**-complete problem.

Theorem 5. *is **Poly-DAPX**-complete under the DPTAS-reduction.*

5 PTAS-completeness

We now study **PTAS**-completeness under FT-reduction. The following theorem introduces the main result of this section.

Theorem 6. *Let Π' be an **NP**-hard a problem of **NPO**. If $\Pi' \in$ **NPO-PB**, then any **NPO** problem FT-reduces to Π'.*

The proof of Theorem 6 immediately follows from Lemmata 3 and 4. The first one introduces a property of Turing-reduction for **NP**-hard problems. In the second one, we transform (under certain conditions) a Turing-reduction into a FT-reduction. Proofs of the two lemmata are given for maximization problems. The case of minimization is completely analogous.

Lemma 3. *If an **NPO** problem Π' is **NP**-hard, then any **NPO** problem Turing-reduces to Π'.*

Proof. Let Π be a **NPO** problem and q be a polynomial such that $|y| \leqslant q(|x|)$ for any instance x of Π and for any feasible solution y of x. Assume that encoding $n(y)$ of y is binary. Then $0 \leqslant n(y) \leqslant 2^{q(|x|)} - 1$. We consider the following problem $\hat{\Pi}$ (see [3]) which is the same as Π up to its objective function that is defined by $m_{\hat{\Pi}}(x, y) = 2^{q(|x|)+1} m_P(x, y) + n(y)$.

Clearly, if $m_{\hat{\Pi}}(x, y_1) \geqslant m_{\hat{\Pi}}(x, y_2)$, then $m_{\Pi}(x, y_1) \geqslant m_{\Pi}(x, y_2)$. So, if y is an optimal solution for the instance x of $\hat{\Pi}$, then it is also an optimal solution for the instance x of Π.

Remark now that for $\hat{\Pi}$, the evaluation problem $\hat{\Pi}_e$ and the constructive problem $\hat{\Pi}$ are equivalent. Indeed, given the value of an optimal solution y, one can determine $n(y)$ (hence y) by computing the remainder of the division of this value by $2^{q(|x|)+1}$.

Since Π' is **NP**-hard, we can solve the evaluation problem $\hat{\Pi}_e$ if we can solve the (constructive) problem Π'. Indeed,

- we can solve $\hat{\Pi}_e$ using an oracle solving the decision version $\hat{\Pi}_d$ of $\hat{\Pi}$, by dichotomy;
- $\hat{\Pi}_d$ reduces to the decision version Π'_d of Π' by a Karp-reduction (see [2, 9] for a formal definition of this reduction);
- finally, one can solve Π'_d using an oracle for the constructive problem Π'.

So, with a polynomial number of queries to an oracle solving Π', one can solve $\hat{\Pi}_e$, $\hat{\Pi}$ and the proof of the lemma is complete. ∎

We now show how, starting from a Turing-reduction (that only preserves optimality) between two **NPO** problems Π and Π' where Π' is polynomially bounded, one can obtain an FT-reduction transforming a fully polynomial time approximation schema for Π' into a fully polynomial time approximation schema for Π.

Lemma 4. *Let $\Pi' \in$ **NPO-PB**. Then, any **NPO** problem that is Turing-reducible to Π' is also FT-reducible to Π'.*

Proof. Let Π be an **NPO** problem and suppose that there exists a Turing-reduction between Π and Π'. Let $\square_\alpha^{\Pi'}$ be an oracle computing, for any instance x' of Π' and for any $\alpha > 0$, a feasible solution y' of x' such that $r(x', y') \geqslant 1 - \alpha$. Moreover, let p be a polynomial such that for any instance x' of Π' and for any feasible solution y' of x', $m(x', y') \leqslant p(|x'|)$.

Let x be an instance of Π. The Turing-reduction claimed gives an algorithm solving Π using an oracle for Π'. Consider now this algorithm where we use, for any query to the oracle with the instance x' of Π', the approximate oracle $\square_\alpha^{\Pi'}(x')$, with $\alpha = 1/(p(|x'|) + 1)$. This algorithm produces an optimal solution, since a solution y' being an $(1 - (1/(p(|x'|) + 1)))$-approximation for x' is an optimal one (recall that we deal with problems having integer-valued objective functions, cf., Definition 1). Really,

$$\begin{aligned} \frac{m_{\Pi'}(x', y')}{\mathrm{opt}_{\Pi'}(x')} \geqslant 1 - \frac{1}{p(|x'|) + 1} \quad &\Longrightarrow \quad m_{\Pi'}(x', y') > \mathrm{opt}_{\Pi'}(x') - 1 \\ &\Longrightarrow \quad m_{\Pi'}(x', y') = \mathrm{opt}(x') \end{aligned}$$

It's easy to see that this algorithm is polynomial when $\square_\alpha^{\Pi'}(x')$ is polynomial in $|x'|$ and in $1/\alpha$.

Obviously, any exact algorithm for Π can be a posteriori seen as a fully polynomial time approximation schema; so, $\Pi \leq_{\mathsf{FT}} \Pi'$ and the proof of the lemma is now complete. ∎

From Theorem 6, one can immediately deduce the two following corollaries.

Corollary 1. $\overline{\textbf{\textit{PTAS}}}^{\mathsf{FT}} = \textbf{\textit{NPO}}$.

Corollary 2. *Any polynomially bounded problem in* ***PTAS*** *is* ***PTAS****-complete under* FT*-reduction.*

For instance, and are in **PTAS** ([4]). What has been discussed in this section concludes then the following result.

Theorem 7. *and are* ***PTAS****-complete under* FT*-reduction.*

Remark that the results of Theorem 7 cannot be trivially obtained using the F-reduction of [5].

6 DPTAS-completeness

We study in this section **DPTAS**-completeness under DFT-reduction. The results we shall obtain are analogous to the case of the **DPTAS**-completeness: we show that any **NPO-DPB NP**-hard problem in **DPTAS** is **DPTAS**-complete.

Theorem 8. *Let* Π' *be an* ***NPO-DPB NP****-hard problem. Then any problem in* ***NPO*** *is* DFT*-reducible to* Π'.

Theorem 8 is an immediate consequence of Lemma 3 and of the following lemma, di erential counterpart of Lemma 4.

Lemma 5. *If* $\Pi' \in$ ***NPO-DPB****, then any* ***NPO*** *problem that is Turing-reducible to* Π' *is also* DFT*-reducible to* Π'.

Proof. Let Π be an **NPO** problem, and suppose that $\Pi \leq_{\mathsf{T}} \Pi'$. Let $\square_\alpha^{\Pi'}$ be an oracle computing, for any instance x' of Π' and for every $\alpha > 0$, a feasible solution y' such that $\delta(x', y') \geqslant (1 - \alpha)$. Let p be a polynomial such that for any instance x' of Π', $|\operatorname{opt}(x') - \omega(x')| \leqslant p(|x'|)$.

In the same way as in Lemma 4, we modify the algorithm of the Turing-reduction between Π and Π' using the approximate oracle $\square_\alpha^{\Pi}$ with $\alpha = 1/(p(|x'|) + 1)$. This algorithm computes, as in Lemma 4, an optimal solution and it is polynomial if the oracle is polynomial in $|x'|$ and in $1/\alpha$. This algorithm is obviously a di erential fully polynomial time approximation schema, and hence, $\Pi \leq_{\mathsf{DFT}} \Pi'$. ∎

Corollary 3. $\overline{\textbf{\textit{DPTAS}}}^{\mathsf{DFT}} = \textbf{\textit{NPO}}$.

Corollary 4. *Any* ***NPO-DPB*** *problem in* ***DPTAS*** *is* ***DPTAS****-complete under* DFT*-reductions.*

The following concluding theorem deals with the existence of **DPTAS**-complete problems.

Theorem 9. *Problems , and are* ***DPTAS****-complete under* DFT*-reduction.*

Proof. For the **DPTAS**-completenes of , just observe that for any instance G, $\omega(G) = 0$. So, standard and di erential approximation ratios coincide for this problem; moreover, it is in both **NPO-PB** and **NPO-DPB**. Then, inclusion $\in$ **PTAS** su ces to conclude $\in$ **DPTAS** and, by Corollary 4, that it is **DPTAS**-complete.

and are a ne equivalent; hence $\leq_{\mathsf{AF}}$. Since AF-reduction is a particular

kind of DFT-reduction, the **DPTAS**-completeness of is immediately concluded.

Finally, the **DPTAS**-completeness of is concluded from the facts: (i) $\in$ **DPTAS** ([7]) and (ii) $\in$ **NPO-DPB** (since, for any instance L of size n, $\omega(L) = n$ and $\mathrm{opt}(L) > 0$. ∎

7 Looking for intermediate problems

FT-reduction is weaker than the F-reduction of [5] and, as we mentioned before, there exist **PTAS**-intermediate problems under this latter reduction. The question of existence of such problems is posed for our reduction too. In this section, we partially answer this question via the following theorem.

Theorem 10. *If there exists an **NPO**-intermediate problem for the Turing-reduction, then there exists a problem **PTAS**-intermediate for FT-reductions.*

Proof. Let Π be an **NPO** problem, intermediate for the Turing-reduction. Suppose that Π is a maximization problem (the minimization case is completely similar). Let p be a polynomial such that, for any instance x and any feasible solution y of x, $m(x, y) \leqslant 2^{q(|x|)}$. Consider the following maximization problem $\widetilde{\Pi}$ where:

- instances are the pairs (x, k) with x an instance of Π and k an integer in $\{0, \ldots 2^{q(|x|)}\}$;
- for an instance (x, k) of $\widetilde{\Pi}$, its feasible solutions are the feasible solutions of the instance x of Π;
- the objective function of $\widetilde{\Pi}$ is:

$$m_{\widetilde{\Pi}}((x,k),y) = \begin{cases} |(x,k)| & \text{if } v(x,y) \geqslant k \\ |(x,k)| - 1 & \text{otherwise} \end{cases}$$

We will now show the three following properties:

1. $\widetilde{\Pi} \in$ **PTAS**;
2. If $\widetilde{\Pi}$ were in **FPTAS**, then Π would be polynomial;
3. if $\widetilde{\Pi}$ were **PTAS**-complete, then Π would be **NPO**-complete under Turing-reductions.

If Properties 1, 2 and 3 hold, then since Π is supposed to be intermediate, one can conclude that $\widetilde{\Pi}$ is **PTAS**-intermediate, under FT.

Proof of Property 1. Remark that $\widetilde{\Pi}$ is clearly in **NPO-PB**. Consider $\varepsilon \in]0, 1]$ and the algorithm $\mathtt{A}_\varepsilon$ which, on the instance (x, k) of $\widetilde{\Pi}$, solves exactly (x, k), if $|(x, k)| \leqslant 1/\varepsilon$; otherwise, it produces some solution. Algorithm $\mathtt{A}_\varepsilon$ is polynomial and guarantees standard approximation ratio, $1 - \varepsilon$. Therefore, $\widetilde{\Pi}$ is in **PTAS**.

Proof of Property 2. Remark that $\Pi \leq_{\mathsf{T}} \widetilde{\Pi}$. Indeed, let x be an instance of Π. We can find an optimal solution of x solving $\log(2^{p(|x|)}) = p(|x|)$ instances (x, k) of $\widetilde{\Pi}$ (by dichotomy). Note that if $\widetilde{\Pi}$ were in **FPTAS**, it would be polynomial since the fully polynomial time approximation schema $\mathtt{A}_\varepsilon$ applied on instance (x, k) with $\varepsilon = 1/(|(x, k)| + 1)$ is an exact and polynomial algorithm. The fact that $\Pi \leq_{\mathsf{T}} \widetilde{\Pi}$ would imply in this case that Π is polynomial.

Proof of Property 3. Assume that $\widetilde{\Pi}$ is **PTAS**-complete (under FT-reductions). Then, - FT-reduces to $\widetilde{\Pi}$. Let $\square$ be an oracle solving Π. Then, we immediately obtain an exact algorithm for $\widetilde{\Pi}$, polynomial if $\square$ is so. Clearly, this algorithm can be considered as a fully polynomial time approximation schema for $\widetilde{\Pi}$. Reduction $\leq_{\mathsf{FT}} \widetilde{\Pi}$ provides a fully polynomial time approximation schema for and, since it is in **NPO-PB**, we get an exact (and polynomial if $\square$ is so) algorithm for it. In other words, if $\widetilde{\Pi}$ is **PTAS**-complete, then

$\leq_{\mathsf{T}} \Pi$. To conclude, is **NPO**-complete under Turing-reduction, since it is **NP**-hard (cf., Lemma 3). Therefore, if $\widetilde{\Pi}$ were **PTAS**-complete, Π would be **NPO**-complete (under Turing-reductions). The proof of Property 3 and of the theorem are now completed. ∎

We now state an analogous result about the existence of **DPTAS**-intermediate problems (under DFT-reductions).

Theorem 11. *If there exists an **NPO**-intermediate problem under Turing-reductions, then there exists a problem **DPTAS**-intermediate, under DFT-reductions.*

Proof. The proof is analogous to one of Theorem 10, up to modification of definition of $\widetilde{\Pi}$. Indeed, if we don't modify it, then $\widetilde{\Pi}$ is not in **DPTAS**, because the value of the worst solution of an instance (x, k) is $|(x, k)| - 1$. We only have to change this definition in order to have $\omega((x, k)) = 0$ for any instance (x, k). For instance, we can define $\widetilde{\Pi}$ as follows:

- instances of $\widetilde{\Pi}$ are, as previously, the pairs (x, k) where x is an instance of Π and k is an integer between 0 and $2^{q(|x|)}$;
- for an instance (x, k) of $\widetilde{\Pi}$, its feasible solutions are the feasible solutions of the instance x of Π, plus a solution y_x^0;
- the objective function of $\widetilde{\Pi}$ is:

$$m_{\widetilde{\Pi}}((x, k), y) = \begin{cases} 0 & \text{if } y = y_x^0 \\ |(x, k)| & \text{if } v(x, y) \geqslant k \\ |(x, k)| - 1 & \text{otherwise} \end{cases}$$

Then, the result claimed is get in exactly the same way as in the proof of Theorem 10. ∎

8 A new DAPX-complete problem

All **DAPX**-complete problems given in [1] are also **APX**-complete under **E**-reduction ([11]), a generalization of the L-reduction of [13]. An interesting question is if there exist **DAPX**-complete problems that are not also **APX**-complete for some standard approximation-preserving reduction. In this section, we positively answer this question by the following theorem.

Theorem 12. *is **DAPX**-complete under DPTAS-reductions.*

Proof. Consider problem and remark that standard ratio for it coincides with di erential ratio of . In fact, these problems are a ne equivalent; so, a posteriori

$$\leq_{\mathsf{AF}} \tag{24}$$

has been proved **MAX-SNP**-hard under L-reduction ([10]). Moreover, as it is shown in [11], $\overline{\textbf{MAX-SNP}}^{\mathsf{E}} =$ **APX-PB** (the set **NPO-PB** $\cap$ **APX**). Since $\text{-}B \in$ **APX-PB**, $\text{-}B \leq_{\mathsf{E}}$. On the other hand, E-reduction being a particular kind of PTAS-reduction, $\text{-}B \leq_{\mathsf{PTAS}}$. Note now that this PTAS-reduction is simultaneously a DPTAS-reduction between $\text{-}B$ and - . In fact, standard and di erential approximation ratios for $\text{-}B$, on the one hand, standard and di erential approximation ratios for and di erential ratio of , on the other hand, coincide. So,

$$\text{-}B \leq_{\mathsf{DPTAS}} \tag{25}$$

Reductions (24) and (25), together with the fact that the composition DPTAS $\circ$ AF is obviously a DPTAS-reduction, establish immediately the **DAPX**-completeness of and the proof of the theorem is now complete. ∎

As we have already mentioned, is, until now, the only problem known to be **DAPX**-complete but not **APX**-complete. In fact, in standard approximation, it belongs to the class **Poly-APX** (of problems for which the best standard ratio known is a polynomial on the size of their instances) and is inapproximable, in a graph of order n, within $n^{1-\varepsilon}$, $\forall \varepsilon > 0$, unless **NP** coincides with the class of problems that could be optimally solved by slightly super-polynomial algorithms ([8]).

9 Conclusion

We have defined suitable reductions and obtained natural complete problems for classes **PTAS** and **DPTAS**.of problems admitting a polynomial time approximation schema, in standard and di erential approximation. This work extends the one in [1]; both aim in studying a structure of di erential approximation classes.

However, the fact that problems proved complete here do not admit a standard or di erential fully polynomial time approximation schema is already known. It would be interesting to use reductions proposed in order to get new inapproximability results. This is in fact a major computational impact of structuring approximation classes. Another interesting open question concerns relationships between F and FT-reductions; for example, is the latter strictly weaker than the former? Finally, the existence of natural **PTAS**-, or **DPTAS**-intermediate problems (as for **APX** under AP-reduction) for F-, FT and DFT-reductions remains open.

References

[1] G. Ausiello, C. Bazgan, M. Demange, and V. Th. Paschos. Completeness in di erential approximation classes. In *Mathematical Foundations of Computer Science, MFCS'03*, number 2747 in Lecture Notes in Computer Science, pages 179–188. Springer-Verlag, 2003.

[2] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and approximation. Combinatorial optimization problems and their approximability properties*. Springer, Berlin, 1999.

[3] G. Ausiello, P. Crescenzi, and M. Protasi. Approximate solutions of NP optimization problems. *Theoret. Comput. Sci.*, 150:1–55, 1995.

[4] B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. Assoc. Comput. Mach.*, 41(1):153–180, 1994.

[5] P. Crescenzi and A. Panconesi. Completeness in approximation classes. *Inform. and Comput.*, 93(2):241–262, 1991.

[6] P. Crescenzi and L. Trevisan. On approximation scheme preserving reducibility and its applications. *Theory of Computing Systems*, 33(1):1–16, 2000.

[7] M. Demange, J. Monnot, and V. Th. Paschos. Bridging gap between standard and di erential polynomial approximation: the case of bin-packing. *Appl. Math. Lett.*, 12:127–133, 1999.

[8] U. Feige and J. Kilian. Zero knowledge and the chromatic number. In *Proc. Conference on Computational Complexity*, pages 278–287, 1996.

[9] M. R. Garey and D. S. Johnson. *Computers and intractability. A guide to the theory of NP-completeness*. W. H. Freeman, San Francisco, 1979.

[10] M. M. Halldórsson. Approximating discrete collections via local improvements. In *Proc. Symposium on Discrete Algorithms,SODA*, pages 160–169, 1995.

[11] S. Khanna, R. Motwani, M. Sudan, and U. Vazirani. On syntactic versus computational views of approximability. *SIAM J. Comput.*, 28:164–191, 1998.

[12] R. E. Ladner. On the structure of polynomial time reducibility. *J. Assoc. Comput. Mach.*, 22:155–171, 1975.

[13] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation and complexity classes. *J. Comput. System Sci.*, 43:425–440, 1991.

[14] C. H. Papadimitriou and M. Yannakakis. The traveling salesman problem with distances one and two. *Math. Oper. Res.*, 18:1–11, 1993.

A A list of NPO problems

We present the list of **NPO** problems mentioned and/or discussed in the paper, together with a characterization of their worst-value solutions. For most of these problems, comments about their approximability in standard approximation can be found in [2].

Maximum variable-weighted satisfiability.
Given a boolean formula φ with non-negative integer weights $w(x)$ on any variable x appearing in φ, maximum variable-weighted satisfiability consists of computing a truth assignment to the variables of φ that both satisfies φ and maximizes the sum of the weights of the variables set to 1. We consider that the assignment setting all the variables to 0, even if it does not satisfy φ, is feasible and represents the worst-value solution for the problem. Maximum linear variable-weighted satisfiability-B denotes the version of Maximum linear variable-weighted satisfiability where the variable-weights are polynomially bounded and their sum lies in the interval $[B, (n/(n-1))B]$. For this problem, it is assumed that the assignment setting all variables to 0 is feasible and that its value is B. Obviously, this assignment represents the worst feasible value.

Maximum independent set (MAX INDEPENDENT SET).
Given a graph $G(V, E)$, an *independent set* is a subset $V' \subseteq V$ such that whenever $\{v_i, v_j\} \subseteq V'$, $v_i v_j \notin E$, and consists in finding an independent set of maximum size. By -B, we denote in bounded-degree graphs. Finally, by , we denote in planar graphs. Worst-value solution: the empty set.

Minimum coloring (MIN COLORING) and maximum color saving (MAX UNUSED COLORS).
Given a graph $G(V, E)$, we wish to color V with as few colors as possible so that no two adjacent vertices receive the same color. Worst-value solution: V. is the problem consisting, given a a graph $G(V, E)$ and a set of $|V|$ colors, of coloring G using colors from the set given, in such a way that the number of unused colors is maximized. Clearly, since the initial set is feasible, worst solution for this problem is exactly this set. It can be immediately seen that the complement of a legal coloring with respect to the vertex-set V is a feasible solution for ; on the other hand, the complement of a feasible solution with respect to the set of initial colors, is a feasible solution for . In other words, and are a ne equivalent.

Minimum vertex-covering (MIN VERTEX COVER).
Given a graph $G(V, E)$, a *vertex cover* is a subset $V' \subseteq V$ such that, $\forall uv \in E$, either $u \in V'$,

or $v \in V'$, and consists of determining a minimum-size vertex cover. By , we denote in planar graphs. Worst-value solution: V.

Bin packing (BIN PACKING).
Given a finite set $L = \{x_1, \ldots, x_n\}$ of n rational numbers and an unbounded number of bins, each bin having a capacity equal to 1; we wish to arrange all these numbers in the least possible bins in such a way that the sum of the numbers in each bin does not violate its capacity. Worst solution: L.

Minimum traveling salesman problem (MIN TSP).
Given a complete graph on n vertices, denoted by K_n, with positive distances on its edges, consists of minimizing the cost of a Hamiltonian cycle (an ordering $\langle v_1, v_2, \ldots, v_n \rangle$ of V such that $v_n v_1 \in E$ and, for $1 \leqslant i < n$, $v_i v_{i+1} \in E$), the cost of such a cycle being the sum of the distances of its edges. We denote by the version of where edge distances satisfy triangle inequalities. Worst-value solution: the total distance of the longest Hamiltonian cycle.