PERGAMON

Applied Mathematics Letters 12 (1999) 127–133

Applied Mathematics Letters

www.elsevier.nl/locate/aml

Bridging Gap Between Standard and Differential Polynomial Approximation: The Case of Bin-Packing

M. DEMANGE*, J. MONNOT AND V. TH. PASCHOS†
LAMSADE, Université Paris-Dauphine
Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France
⟨demange⟩⟨monnot⟩⟨paschos⟩@lamsade.dauphine.fr

(Received and accepted January 1998)

Communicated by R. Janssen

Abstract—The purpose of this paper is mainly to prove the following theorem: for every polynomial time algorithm running in time $T(n)$ and guaranteeing standard-approximation ratio ρ for bin-packing, there exists an algorithm running in time $O(nT(n))$ and achieving differential-approximation ratio $2-\rho$ for BP. This theorem has two main impacts. The first one is "operational", deriving a polynomial time differential-approximation schema for bin-packing. The second one is structural, establishing a kind of reduction (to our knowledge not existing until now) between standard approximation and differential one.

1. STANDARD AND DIFFERENTIAL APPROXIMATION

A current and very active research area coping with **NP**-completeness is *polynomial approximation theory*. In this domain, the main objective is either finding a good approximation algorithm for a given **NP**-complete problem, or establishing proofs that such algorithms cannot exist unless an unlikely complexity-theory condition (for example, $\mathbf{P} = \mathbf{NP}$) holds. The "goodness" of an approximation algorithm is commonly measured by its approximation ratio.

Given an instance I of a combinatorial optimization problem Π and an approximation algorithm $\mathtt{A}$ supposed to feasibly solve Π, we will denote by $\omega(I)$, $\lambda_{\mathtt{A}}(I)$, and $\beta(I)$ the values of the worst case solution, the approximated one (provided by $\mathtt{A}$), and the optimal one, respectively.

There exist mainly two thought processes dealing with polynomial approximation. Traditionally [1,2], the quality of an approximation algorithm for an **NP**-complete minimization (respectively, maximization) problem Π is expressed by the ratio (called standard in what follows) $\rho_{\mathtt{A}}(I) = \lambda(I)/\beta(I)$, and the quantity $\rho_{\mathtt{A}} = \inf\{r : \rho_{\mathtt{A}}(I) < r,\ I \text{ instance of } \Pi\}$ (respectively, $\rho_{\mathtt{A}} = \sup\{r : \rho_{\mathtt{A}}(I) > r,\ I \text{ instance of } \Pi\}$) constitutes the approximation ratio of $\mathtt{A}$ for Π. Recent works [3,4] strongly inspired by former ones (see, for example, [5]) bring to the fore another approximation measure, as powerful as the traditional one (concerning the type, the diversity, and the quantity of the produced results), the ratio (called differential in what follows)

*Also, CERMSEM, Université Paris I, Maison des Sciences Economiques, 106-112 boulevard de l'Hôpital, 75647 Paris Cedex 13, France.
†Author to whom all correspondence should be addressed.

0893-9659/99/$ - see front matter. Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII:S0893-9659(99)00112-3

$\delta_{\mathtt{A}}(I) = [\omega(I) - \lambda(I)]/[\omega(I) - \beta(I)]$. The quantity $\delta_{\mathtt{A}} = \sup\{r : \delta_{\mathtt{A}}(I) > r,\ I$ instance of $\Pi\}$ is now the approximation ratio of A for Π.

A special case of a polynomial time approximation algorithm, inducing the strongest possible positive approximation result, is the one of *polynomial time approximation schema.* A polynomial time standard-approximation schema for a problem Π is a sequence $\mathtt{A}_\epsilon$ of polynomial time approximation algorithms (receiving ϵ among their inputs) and guaranteeing standard-approximation ratio $1+\epsilon$, for every fixed $\epsilon > 0$, if Π is a minimization problem and $1-\epsilon$, for every fixed $\epsilon > 0$, if Π is a maximization one. A polynomial time differential-approximation schema for Π is a sequence $\mathtt{A}_\epsilon$ of polynomial time approximation algorithms (receiving ϵ among their inputs) and guaranteeing differential-approximation ratio $1-\epsilon$, for every fixed $\epsilon > 0$.

As is shown in [3,4], many problems behave in completely different ways regarding traditional or differential approximation. This is, for example, the case of minimum graph-coloring, or even of minimum vertex-covering.[1] The former is approximated within differential ratio 3/4 [6], while no polynomial time algorithm can guarantee standard-approximation ratio n^ϵ, for any constant $\epsilon < 1$, for graph-coloring unless **NP** $\subseteq$ **coRP** [7], where n is the order of the input-graph. On the contrary, for vertex-covering, no polynomial time algorithm can guarantee differential-approximation ratio $n^{(1/2)-\epsilon}$, for any $\epsilon > 0$ (n being the order of the input-graph). This result seems to be strengthened in [8] (cited in [7]) since hardness factor $(1/2) - \epsilon$ is improved to $1-\epsilon$ for any $\epsilon > 0$) unless **NP** = **coRP** [9], while vertex-covering is approximable within standard-approximation ratio $2-(\log\log n/\log n)$ [10]. An easy consequence of the above remarks is that no general approximation-preserving reduction allows transfer of positive, negative, or conditional results from standard approximation to differential one and vice-versa. Moreover, even for particular problems, such reductions have not been devised until now.

2. AN APPROXIMATION-PRESERVING REDUCTION FOR BIN-PACKING

In the Bin-Packing problem (BP), we are given a finite set $L = \{x_1, \dots, x_n\}$ 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. BP is **NP**-complete and, consequently, no polynomial time algorithm can exactly solve it, unless **P** = **NP**.

The purpose of this section is to prove the following theorem.

THEOREM 1. *Let $\rho_{\mathtt{A}} \geq 1$ be a fixed constant and let* A *be an algorithm approximately solving BP within standard-approximation ratio $\rho_{\mathtt{A}}$ (i.e., $\lambda_{\mathtt{A}}(L)/\beta(L) \leq \rho_{\mathtt{A}}$, for every BP-instance L of size n) and running in time $T_{\mathtt{A}}(n)$. Then there exists an algorithm* D(A)*, running in time $T_{\mathtt{D(A)}}(n) = nT_{\mathtt{A}}(n)$ guaranteeing differential-approximation ratio $\delta_{\mathtt{D(A)}} \geq 2 - \rho_{\mathtt{A}}$ (i.e., $[\omega(L) - \lambda_{\mathtt{D(A)}}(L)]/[\omega(L) - \beta(L)] \geq 2 - \rho_{\mathtt{A}}$, for every instance L of BP).*

Let us fix a list L of size n and denote by $B_{\mathtt{A}}$ the solution computed by A and by B^*, the optimal one. These solutions are in fact sets of bins. A bin i will be denoted either by b_i, or by the set of its elements; a BP-solution will be alternatively denoted by the union of its bins. Moreover, consider the following algorithm D, parametrized by a BP-algorithm A.

```
BEGIN /D(A)/
      order L in decreasing order;
      let L = {x_1,...,x_n} be the ordered list obtained;
      FOR k <- 0 TO n-1 DO
            L_k <- {x_{k+1},...,x_n};
            B_k <- {x_1} ∪ {x_2} ∪ ··· ∪ {x_k} ∪ A(L_k);
      OD
```

[1]Note that the differential ratio for the minimum vertex-covering and the maximum independent set are identical.

```
    BD ← argmin_{k=0,...,n-1}{|B_k|};
    OUTPUT BD;
END /D(A)/
```

The following proposition provides an easy, but useful description of an optimal BP-solution.

PROPOSITION 1. *Let B^* be an optimal BP-solution of L (this list is supposed to be ordered in decreasing order) and let k^* ($k^* \in \{0,\dots,n\}$) be the number of 1-item bins of B^*. Then there exists an optimal BP-solution $\tilde{B}^* = \{x_1\} \cup \cdots \cup \{x_{k^*}\} \cup \tilde{B}_2^*$, i.e., consisting of k^* 1-item bins containing the first k^* items of L, one item per bin, and of a set $\tilde{B}_2^*$ of bins, each bin b_i of this set containing at least two items.*

PROOF. Let us denote by $\{y_1\},\dots,\{y_{k^*}\}$ the k^* 1-item bins of B^*. Then there exists a bijection $\varphi : \{x_1,\dots,x_{k^*}\} \to \{y_1,\dots,y_{k^*}\}$ such that, $\forall i \leq k^*$, $\varphi(x_i) \leq x_i$. Given $B^* = \{y_1\} \cup \cdots \cup \{y_{k^*}\} \cup \bar{B}^*$, solution $\tilde{B}^* = \{x_1\} \cup \cdots \cup \{x_{k^*}\} \cup \tilde{B}_2^*$, where $\tilde{B}_2^*$ is identical to $\bar{B}^*$ up to substitution of x_i by $\varphi(x_i)$ in the corresponding bins of $\bar{B}^*$, is solution claimed. This solution is feasible since $x_i \leq 1$, $x_i \geq \varphi(x_i)$, and $\{\varphi(x_i)\} \in B^*$. Moreover, it is optimal since $|\tilde{B}^*| = |B^*|$. Finally, note that, given B^*, φ can be computed in polynomial time. ∎

In order to continue the proof of the theorem, we point out that the following lemma, called Bellman-like principle, holds for BP.

LEMMA 1. BELLMAN-LIKE PRINCIPLE FOR BP. *Let L be an instance of BP and denote by $B^* = \{b_j^* : j = 1,\dots,\beta(L)\}$ an optimal BP-solution for L. Then, for every set $J \subset \{1,\dots,\beta(L)\}$, solution $B_J^* = \{b_j^* : b_j^* \in B^*,\ j \in J\}$ is an optimal solution for the sublist $\cup_{j\in J} b_j^*$.*

Let us now denote by $\xi(B^*, L)$ the list $L' = \{x_{k^*+1},\dots,x_n\}$ and revisit solution $\tilde{B}^*$. According to Lemma 1, set $\tilde{B}_2^*$ is an optimal BP-solution for $\xi(B^*, L)$. Furthermore, since FOR-loop of algorithm D(A) is executed for L, as well as for every sublist resulting from L by removing the k largest elements of L, $k = 0,\dots,n-1$, algorithm A is also called on $\xi(B^*, L) = L' = \{x_{k^*+1},\dots,x_n\}$. Since the smallest of the solutions obtained is finally retained, $|BD| = \lambda_{\mathtt{D(A)}} \leq |B_{k^*}|$. Finally, note that the worst-case BP-solution for L consists in taking a bin per item,[2] i.e., $\omega(L) = |L| = n$. So we have, for every optimal BP-solution B^* of L,

$$\begin{aligned} \beta(L) &= \beta(\xi(B^*, L)) + k^*, \\ \lambda_{\mathtt{D(A)}}(L) &\leq k^* + \lambda_{\mathtt{A}}(\xi(B^*, L)), \\ \omega(L) &= n, \\ |\xi(B^*, L)| &= n - k^*, \\ |\xi(B^*, L)| &\geq 2\beta(\xi(B^*, L)), \end{aligned} \tag{1}$$

where the last of the above expressions holds because each bin of $\tilde{B}_2^*$ contains at least two items.

Combining the expressions above, we get

$$\delta_{\mathtt{D(A)}}(L) = \frac{\omega(L) - \lambda_{\mathtt{D(A)}}(L)}{\omega(L) - \beta(L)} = \frac{n - \lambda_{\mathtt{D(A)}}(L)}{n - \beta(L)} \geq \frac{|\xi(B^*, L)| - \lambda_{\mathtt{A}}(\xi(B^*, L))}{|\xi(B^*, L)| - \beta(\xi(B^*, L))}. \tag{2}$$

It suffices now to remark that function $[|\xi(B^*, L)| - \lambda_{\mathtt{A}}(\xi(B^*, L))]/[|\xi(B^*, L)| - \beta(\xi(B^*, L))]$ is increasing in $|\xi(B^*, L)|$ and to use expression (1) to obtain

$$\delta_{\mathtt{D(A)}}(L) \geq 2 - \frac{\lambda_{\mathtt{A}}(\xi(B^*, L))}{\beta(\xi(B^*, L))} = 2 - \rho_{\mathtt{A}}(\xi(B^*, L)) \geq 2 - \rho_{\mathtt{A}}, \tag{3}$$

[2] One can remark that, adopting differential framework, BP can be picturesquely expressed as the problem of maximizing "unused" bins.

where the last inequality is true thanks to the fact that arguments developed above hold for every BP-instance L. So, the approximation result claimed by the theorem is immediately achieved.

Finally, for $T_{\mathtt{D(A)}}(n)$, it suffices to note that algorithm D(A) mainly consists of at most n calls of algorithm A, and this completes the proof of Theorem 1.

REMARK 1. As expression (3) makes clear, the result really proved is somewhat stronger than the one claimed in Theorem 1. In fact, subexpression $\delta_{\mathtt{D(A)}}(L) \geq 2 - \rho_{\mathtt{A}}(\xi(B^*, L))$ establishes a connection between standard and differential approximation working for all ratios $\rho_{\mathtt{A}}$ and not only for fixed constant ones. ∎

Moreover, expression (2) brings to the fore the following corollary which will be used in what follows.

COROLLARY 1. $\delta_{\mathtt{D(A)}}(L) \geq \delta_{\mathtt{A}}(\xi(B^*, L)) \geq 2 - \rho_{\mathtt{A}}(\xi(B^*, L))$.

3. A POLYNOMIAL TIME DIFFERENTIAL-APPROXIMATION SCHEMA FOR BIN-PACKING

As we have already mentioned, Theorem 1 and Remark 1 establish (for the first time) a reduction between standard and differential approximation for an **NP**-complete problem. An impact of this theorem is that any positive standard-approximation result for BP can be transformed into a positive differential-approximation result, while any negative differential-approximation result is transformed into negative standard-one.

Fortunately, BP is a "nice" problem in the sense that most of the standard-approximation results known about it are positive ones (references [11,12] are a small list of older but always exciting works about positive standard-approximation results for BP). In fact, a "bunch" of algorithms, FFD and BFD being the most well-known ones, guarantee constant standard-approximation ratios for it. The strongest standard-approximation result[3] is the one in [13], where it is proved that *for every fixed positive ϵ, BP can be approximated within standard ratio* $1 + \epsilon + [1/\beta(L)]$, *in time identical to the one needed for linear-programming.* Finally, for standard approximation, one can easily prove that *no polynomial time approximation algorithm can achieve standard-approximation ratio (strictly) less than* $1 + [1/\beta(L)]$ *for BP, unless* $\mathbf{P} = \mathbf{NP}$ (let us note that in [1], the question about the existence of a standard-approximation polynomial time algorithm A satisfying, $\forall L$, $\lambda_{\mathtt{A}}(L)/\beta(L) \leq 1 + [1/\beta(L)]$ is evocated). Plainly, if such an algorithm A exists and guarantees, for every BP-instance L, $\lambda_{\mathtt{A}}(L)/\beta(L) < 1 + [1/\beta(L)]$, then $\lambda_{\mathtt{A}}(L) < \beta(L) + 1$. Since quantities $\lambda(L)$ and $\beta(L)$ are integers, equality $\lambda_{\mathtt{A}}(L) = \beta(L)$ is immediately deduced.

The strongest differential-approximation result was, until now, the one in [14], where it is proved that FFD achieves differential-approximation ratio $\delta_{\mathtt{FFD}} \geq 3/4$, in time $O(n \log n)$. Application of Theorem 1, taking into account that, $\forall L$, $\rho_{\mathtt{FFD}} \leq (11/9) + [4/\beta(L)]$ (see [1]), further strengthens the result of [14], since $\delta_{\mathtt{D(FFD)}} \geq 7/9 - [4/\beta(\xi(B^*, L))]$. For BP-instances L with unbounded $\beta(\xi(B^*, L))$-values, this ratio is arbitrarily close to 7/9 while, as we will see below, for instances with bounded $\beta(\xi(B^*, L))$-values, BP is polynomial.

The rest of this section is devoted to the proof of the following theorem.

THEOREM 2. *BP can be solved by a polynomial time differential-approximation schema.*

In what follows, we denote by E an exhaustive-search algorithm for BP (running in time $O(2^n)$), by A any polynomial algorithm approximately solving BP within (fixed) constant standard-approximation ratio $\rho_{\mathtt{A}} \geq 1$, and by $\mathtt{S}(\epsilon)$ the algorithm of [13].

Consider now the following algorithm $\mathtt{EX}(\mathtt{E}, \mu)$, where L is supposed to be ordered in decreasing order and $\mu \in \{0, \ldots, n\}$.

---

[3]This result turns out to be an asymptotic standard-approximation ratio (see [1] for a definition of asymptotic (standard) approximation ratio) $1+\epsilon$, for every fixed positive ϵ, for BP-instances L with unbounded values for $\beta(L)$.

```
BEGIN /EX(E,μ)/
(1)   LB ← {L̂_i ⊆ L : L̂_i = {x_i,...,x_n}, n − μ + 1 ≤ i ≤ n};
(2)   FOR i ← 1 TO |LB| DO B̂_i ← {{x} : x ∈ L \ L̂_i} ∪ E(L̂_i) OD
(3)   EB ← argmin_{1≤i≤|LB|} {|B̂_i|};
(4)   OUTPUT EB;
END. /EX(E,μ)/
```

It is easy to see that EX(E,μ) finds a feasible BP-solution for L in polynomial time (whenever μ is a fixed constant). Moreover, this solution is optimal whenever the size $|\xi(B^*,L)|$ of $\xi(B^*,L)$ ($\xi(B^*,L)$ being as in the proof of Proposition 1) is bounded by μ, as the following lemma shows.

LEMMA 2. *For lists L admitting optimal BP-solutions B^* such that $|\xi(B^*,L)| \leq \mu$, algorithm* EX(E,μ) *exactly solves BP in L, in time $O(\mu 2^\mu)$ which is polynomial in n whenever μ is a fixed constant.*

PROOF. Following Proposition 1, an optimal BP-solution for L consists of using, for a $k^* \leq n$, k^* bins containing the k^* largest items of L, one item per bin, and $|\tilde{B}_2^*|$ additional bins for the items of the list $\xi(B^*,L) = \{x_{k^*+1},\dots,x_n\}$. Furthermore, note that set LB, computed by algorithm EX(E,μ), consists of all sublists containing at most the μ last elements of L (recall that elements of L are ordered in decreasing order). So, on the hypothesis that $|\xi(B^*,L)| \leq \mu$, $\xi(B^*,L) \in LB$ and, consequently, optimal solution for $\xi(B^*,L)$ is computed by E during execution of FOR-loop of line (2). Let $\xi(B^*,L) = \hat{L}_{i^*}$. Then, $\hat{B}_{i^*}$ is an optimal BP-solution for L and, obviously, being the smallest one, it will be chosen at line (3). Hence, algorithm EX(E,μ) really computes an optimal BP-solution for L. Since $|LB| = \mu$ and, moreover, exhaustive search performed by E($\hat{L}_i$) takes $O(2^\mu)$ steps, overall complexity of EX(E,μ) is $O(\mu 2^\mu)$, polynomial whenever μ is fixed. ∎

We now continue proof of Theorem 2 by proving that, for any polynomial time approximation BP-algorithm A achieving constant standard-approximation ratio $\rho_{\mathtt{A}}$, and for any fixed $\epsilon > 0$, if $|\xi(B^*,L)| \geq 2(\rho_{\mathtt{A}} - 1 + \epsilon)/\epsilon^2$ and if $\beta(\xi(B^*,L)) \leq \epsilon|\xi(B^*,L)|/(\rho_{\mathtt{A}} - 1 + \epsilon)$, then algorithm D(A) (of Section 2) guarantees differential-approximation ratio at least $1 - \epsilon$.

LEMMA 3. *Let* A *be any polynomial time approximation algorithm for BP guaranteeing standard-approximation ratio $\rho_{\mathtt{A}}$, and let ϵ be any fixed positive constant. If $|\xi(B^*,L)| \geq 2(\rho_{\mathtt{A}} - 1 + \epsilon)/\epsilon^2$ and if $\beta(\xi(B^*,L)) \leq \epsilon|\xi(B^*,L)|/(\rho_{\mathtt{A}} - 1 + \epsilon)$, then $\delta_{\mathtt{D(A)}}(L) \geq 1 - \epsilon$.*

PROOF. Under the hypotheses of the lemma, and since $[|\xi(B^*,L)| - \rho_{\mathtt{A}}\beta(\xi(B^*,L))]/[|\xi(B^*,L)| - \beta(\xi(B^*,L))]$ is decreasing in $\beta(\xi(B^*,L))$, we have

$$\begin{aligned}\delta_{\mathtt{A}}(\xi(B^*,L)) &= \frac{|\xi(B^*,L)| - \lambda_{\mathtt{A}}(\xi(B^*,L))}{|\xi(B^*,L)| - \beta(\xi(B^*,L))} \geq \frac{|\xi(B^*,L)| - \rho_{\mathtt{A}}\beta(\xi(B^*,L))}{|\xi(B^*,L)| - \beta(\xi(B^*,L))} \\ &\geq \frac{|\xi(B^*,L)| - (\rho_{\mathtt{A}}\epsilon|\xi(B^*,L)|)/(\rho_{\mathtt{A}} - 1 + \epsilon)}{|\xi(B^*,L)| - (\epsilon|\xi(B^*,L)|)/(\rho_{\mathtt{A}} - 1 + \epsilon)} \geq 1 - \epsilon.\end{aligned} \tag{4}$$

Next, it suffices to use Corollary 1 affirming that $\delta_{\mathtt{D(A)}}(L) \geq \delta_{\mathtt{A}}(\xi(B^*,L))$; so, $\delta_{\mathtt{A}}(L) \geq 1 - \epsilon$. ∎

We finally prove that, for every fixed $\epsilon > 0$ and for lists L for which $|\xi(B^*,L)| \geq 2(\rho_{\mathtt{A}} - 1 + \epsilon)/\epsilon^2$ and $\beta(\xi(B^*,L)) \geq \epsilon|\xi(B^*,L)|/(\rho_{\mathtt{A}} - 1 + \epsilon)$, algorithm D, parametrized by S($\epsilon/2$), achieves differential-approximation ratio bounded below by $1 - \epsilon$.

LEMMA 4. *Consider BP-algorithm* S(ϵ) *of [13] and let ϵ be any fixed positive constant. If L is such that $|\xi(B^*,L)| \geq 2(\rho_{\mathtt{A}} - 1 + \epsilon)/\epsilon^2$ and if $\beta(\xi(B^*,L)) \geq \epsilon|\xi(B^*,L)|/(\rho_{\mathtt{A}} - 1 + \epsilon)$, then $\delta_{\mathtt{D(S(\epsilon/2))}}(L) \geq 1 - \epsilon$.*

PROOF. Since $\beta(\xi(B^*,L)) \geq \epsilon|\xi(B^*,L)|/(\rho_{\mathtt{A}} - 1 + \epsilon)$ and since $\rho_{\mathtt{S}(\epsilon)}(\xi(B^*,L)) \leq 1 + \epsilon + [1/\beta(\xi(B^*,L))]$ (see [13]), then applying Theorem 1, we obtain

$$\begin{aligned}\delta_{\mathtt{D}(\mathtt{S}(\epsilon/2))}(L) &\geq 2 - \rho_{\mathtt{S}(\epsilon/2)}(\xi(B^*,L)) \geq 2 - \left(1 + \frac{\epsilon}{2} + \frac{1}{\beta(\xi(B^*,L))}\right) \\ &\geq 1 - \frac{\epsilon}{2} - \frac{\rho_{\mathtt{A}} - 1 + \epsilon}{\epsilon|\xi(B^*,L)|} \geq 1 - \epsilon,\end{aligned} \tag{5}$$

where the last inequality holds thanks to lower bound in the size of $\xi(B^*,L)$. ∎

Ideas in proofs of Lemmata 2–4 can be combined into the following algorithm for BP.

```
BEGIN/PTDAS(ε)/
(1)  fix a constant ε > 0;
(2)  μ ← ⌊2(ρ − 1 + ε)/ε²⌋;
(3)  EB ← EX(E,μ)(L)
(4)  DA ← D(A)(L);
(5)  DS ← D(S(ε/2))(L);
(6)  B ← argmin{|EB|,|DA|,|DS|};
(7)  OUTPUT B
END./PTDAS(ε)/
```

Let us fix a BP-instance L. Then, since $\rho_{\mathtt{A}}$ and ϵ do not depend on n, neither does μ, computed at line (2). Consequently, by Lemma 2, computation at line (3) can be performed in polynomial time and, if $|\xi(B^*,L)| \leq \mu$, provides optimal solution for L. On the other hand, if $|\xi(B^*,L)| > \mu = \lfloor 2(\rho - 1 + \epsilon)/\epsilon^2 \rfloor$, then $|\xi(B^*,L)| \geq 2(\rho - 1 + \epsilon)/\epsilon^2$ and Lemmata 3 and 4 guarantee achievement of differential-approximation ratio $1 - \epsilon$ for algorithm PTDAS(ϵ) for every possible value of $\beta(\xi(B^*,L))$. Moreover, since arguments above hold for every L, expressions (4) and (5) always hold and, consequently, algorithm PTDAS is a polynomial time differential-approximation schema for BP and proof of Theorem 2 is completed.

4. LIMITS ON DIFFERENTIAL APPROXIMABILITY OF BIN-PACKING

The result of Theorem 2 affirms that BP is better approximated in a differential framework than in a standard one. A common thought process for proving existence of positive (standard) approximation asymptotic results for simple[4] problems is to partition their instances into two classes following their optimal values; the former class consists of bounded optimal-value instances and the latter of unbounded optimal-value ones. Then, one proves that for the former class, an optimal polynomial time algorithm[5] providing optimal solutions exists, while, for the latter class, one proves the existence of a polynomial time standard-approximation algorithm achieving a certain ratio. This is the only way known for proving standard-approximation asymptotic positive results. Following such a thought process to extend the result of [13] cannot work here since, unfortunately, BP is not simple (in the sense of [15]). In fact, it is easy to see that for $\beta(L) = 2$, partition problem [1] is a restricted case of BP.

What are the limits of differential approximability for BP? Unfortunately, it cannot be approximated by fully polynomial time differential-approximation schemata, as the following proposition shows.

[4]An **NP**-complete problem is called simple [15] if in instances for which optimal values are bounded by fixed constants, the problem can be solved in polynomial time; a lot of problems, even hard to approximate ones (from both standard- or differential-approximation points of view), as maximum independent set or minimum vertex-covering are simple (on the contrary, minimum-graph-coloring is not simple).

[5]Usually, this algorithm is an exhaustive search performed in polynomial time thanks to the fact that β is bounded.

PROPOSITION 2. *Unless* $\mathbf{P}{=}\mathbf{NP}$, *BP cannot be solved by a polynomial time differential-approximation algorithm within ratio bounded below by* $1-(1/n)$. *Consequently, BP does not admit a fully polynomial time differential-approximation schema.*

PROOF. If a polynomial time algorithm A exists, achieving, for every L, ratio $[n-\lambda_{\mathtt{A}}(L)]/[n-\beta(L)] \geq 1-(1/n)$, then, for every L, $\lambda_{\mathtt{A}}(L)-\beta(L) \leq 1-(\beta(L)/n) < 1$. Since quantities $\lambda_{\mathtt{A}}(L)$ and $\beta(L)$ are integers, $\lambda_{\mathtt{A}}(L)=\beta(L)$ holds for every BP-instance. So, A would be an exact polynomial time algorithm for BP, consequently, $\mathbf{P}{=}\mathbf{NP}$. ∎

Finally, let us conclude this paper with a rather optimistic remark. Revisit Theorem 2 and Proposition 2. It is true that differential ratio for BP can be greater than $1-\epsilon$, for every $\epsilon > 0$, but it cannot be greater than $1-(1/|L|)$, for every L. However, between a fixed constant and $|L|$, there exists a continuum of ϵs, even depending on $|L|$, for which strong positive differential-approximation results are obtained via Theorem 2.

For example, consider in algorithm PTDAS, $\epsilon = 1/(\log n)^{1/2}$. Since complexity[6] of PTDAS is of $O(\max\{T_{\mathtt{A}}(n), n^4/\log n, (2/\epsilon^2)4^{(1/\epsilon)^2}\})$, then applying Theorem 2, the following corollary holds.[7]

COROLLARY 2. *BP can be approximated by an* $O(n^4 \log n)$ *approximation algorithm within differential ratio* $1-[1/(\log n)^{1/2}]$.

REFERENCES

1. M.R. Garey and D.S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, W.H. Freeman, San Francisco, CA, (1979).
2. C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, NJ, (1981).
3. M. Demange and V.T. Paschos, On an approximation measure founded on the links between optimization and polynomial approximation theory, *Theoret. Comput. Sci.* **158**, 117–141 (1996).
4. M. Demange, P. Grisoni and V.T. Paschos, Differential approximation algorithms for some combinatorial optimization problems, *Theoret. Comput. Sci.* (to appear).
5. G. Ausiello, A. D'Atri and M. Protasi, Structure preserving reductions among convex optimization problems, *J. Comput. System Sci.* **21**, 136–153 (1980).
6. M.M. Halldórsson, Approximating k-set cover and complementary graph coloring, In *Proc. International Integer Programming and Combinatorial Optimization Conference*, Number 1084 in Lecture Notes in Computer Science, pp. 118–131, Springer-Verlag, (1996).
7. U. Feige and J. Kilian, Zero knowledge and the chromatic number, In *Proc. Conference on Computational Complexity*, pp. 278–287, (1996).
8. J. Håstad, Clique is hard to approximate within $n^{1-\epsilon}$, In *Proc. FOCS '96*, pp. 627–636, (1996).
9. J. Håstad, Testing on the long code and hardness for clique, In *Proc. STOC '96*, pp. 11–19, (1996).
10. R. Bar-Yehuda and S. Even, A local-ratio theorem for approximating the weighted vertex cover problem, *Ann. Discrete Appl. Math.* **25**, 27–46 (1985).
11. D.S. Johnson, Fast algorithms for bin packing, *J. Comput. System Sci.* **8**, 272–314 (1974).
12. D.S. Johnson, A. Demers, J.D. Ullman, M.R. Garey and R.L. Graham, Worst-case performance bounds for simple one-dimensional packing algorithms, *SIAM J. Comput.* **3** (4), 298–325 (1974).
13. W. Fernandez de la Vega and G.S. Lueker, Bin packing can be solved within $1+\epsilon$ in linear time, *Combinatorica* **1** (4), 349–355 (1981).
14. J. Monnot, Familles d'instances critiques et approximation polynomiale, Ph.D. Thesis, LAMSADE, Université Paris-Dauphine (in preparation).
15. A. Paz and S. Moran, Nondeterministic polynomial optimization problems and their approximations, *Theoret. Comput. Sci.* **15**, 251–277 (1981).
16. K.M. Anstreicher, Linear programming in $O((n^3/\log n)L)$ operations, Discussion Paper 9746, CORE, Université Catholique de Louvain, (June 1997).

[6]The best complexity known for linear programming is, to our knowledge, $O(n^3/\log n)$ (see [16]).

[7]Most of the classical standard-approximation algorithms for BP work in $O(n \log n)$.