

CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
(Université Paris-Dauphine)
Unité de Recherche Associée au CNRS ESA 7024



THE MAXIMUM HAMILTONIAN PATH PROBLEM WITH SPECIFIED ENDPOINT(S)

CAHIER N° 176
mars 2001

Jérôme MONNOT ¹

reçu : décembre 2000.

¹ LAMSADE, Université Paris-Dauphine, Place du Maréchal De Latre de Tassigny, 75775 Paris Cedex 16, France (monnot@lamsade.dauphine.fr).

CONTENTS

	<u>Pages</u>
Résumé	i
Abstract	i
1. Introduction	1
2. The Hamiltonian path problem	3
3. Elementary properties	4
4. Approximate algorithms for these problems	6
4.1 The algorithm for two specified endpoints version	6
4.2 The algorithm for one specified endpoint version	8
5. Towards threshold of differential non-approximation	10
References	11

Le problème de la chaîne Hamiltonienne de poids maximum avec extrémité(s) prédéterminée(s)

Résumé

Cet article traite des différents problèmes de la chaîne Hamiltonienne, nommés $HPP_{s,t}$ lorsque les deux extrémités sont connues à l'avance et HPP_s lorsqu'une seule est prédéterminée. Nous montrons que $HPP_{s,t}$ est $1/2$ -différentielle approximable et n'est pas approximable à mieux de $649/650$. Ensuite, nous prouvons que la version HPP_s est mieux approximable avec un ratio de $2/3$. Basés sur ces résultats, nous obtenons des bornes pour la mesure standard: une $1/2$ -approximation standard pour $HPP_{s,t}$ et une $2/3$ -approximation standard pour HPP_s . Nous améliorons ce ratio à $2/3$ pour $Max HPP_{s,t}[a,2a]$ (lorsque les arêtes ont un coût appartenant à l'intervalle $[a,2a]$), à $5/6$ pour $Max HPP_s[a,2a]$ et finalement à $2/3$ pour $Min HPP_{s,t}[a,2a]$, à $3/4$ pour $Min HPP_s[a,2a]$.

Mots-clés: Algorithmes approchés; Mesure différentielle; Mesure standard; Réductions

The maximum Hamiltonian Path Problem with specified endpoint(s)

Abstract

This paper deals with the problem of constructing Hamiltonian paths of optimal weight, called $HPP_{s,t}$ if the two endpoints are specified, HPP_s if only one endpoint is specified. We first show that $HPP_{s,t}$ is $1/2$ - differential approximable and can not be differential approximable greater than $649/650$. We next demonstrate that, when dealing with one specified endpoint, we improve to $2/3$ the ratio. Based upon these results, we obtain new bounds for *standard ratio*: a $1/2$ -standard approximation for $Max HPP_{s,t}$ and a $2/3$ for $Max HPP_s$ which can be improved to $2/3$ for $Max HPP_{s,t}[a,2a]$ (all the edge weights are within an interval $[a,2a]$), to $5/6$ for $Max HPP_s[a,2a]$ and to $2/3$ for $Min HPP_{s,t}[a,2a]$, to $3/4$ for $Min HPP_s[a,2a]$.

Keywords: Approximate algorithms; Differential ratio; Performance ratio; Analysis of Algorithms;

1 Introduction

Routing design problems are of a major importance in combinatorial optimization, and the most important ideas of algorithmic have been applied to them during the last twenty years, see Christofides [4], Fisher et al [12], Hassin and Rubinstein [15] and Engebreetsen [11]. We will be concerned with some problems closely related to the Maximum Traveling Salesman problem, namely, the problem of finding a *Hamiltonian path* of maximum weight. We will study two variants depending on the number of specified endpoints (one or two) of the path. $Max\ HPP_s$ and $Max\ HPP_{s,t}$ respectively denote the Hamiltonian path problem with one fixed endpoint $s \in V$ and two fixed endpoints $s, t \in V$. To our knowledge, these two latter problems have not been studied before, whereas their minimization versions have been studied by Hoogeveen [17] and Guttmann-beck et al. [14] (in particular, it is well known that the minimization problems are *NP-hard*). We also deal with a variant called $HPP_{s,t}[a; b]$, where the edge-weights are in the set $\{a, a+1, \dots, b-1, b\}$. Both *Min-* and *Max* $HPP_{s,t}$ are *NP-hard*, even in their restricted versions with $b > a$, since they are polynomial-time Karp-reducible [19] to each other.

We focus on the design of approximation algorithms with guaranteed performance ratios, that run within polynomial time and produce sub-optimal solutions. Usually, the scientific community works on the ratio (called standard ratio) of the cost of the solution generated by the algorithm to the optimal cost, in the worst-case. However, we mainly refer in this article to another ratio called *differential ratio* which measures the worst ratio of, on the one hand, the difference between the cost of the solution generated by the algorithm and the worst cost, and on the other hand, the difference between the optimal cost and the worst cost. This measure, studied by Ausiello et al. [3], Fisher et al. [12] and more recently by Demange and Paschos [10], leads to new algorithms taking into account the extreme solutions of the instance, and provides the opportunity to better understand these problems. There are great differences between standard and differential approximation for the Hamiltonian path problems. For instance, we can easily prove that the *Best Neighbor Heuristic* [12] is $1/3$ -standard approximable for $Max\ HPP_s$ or that we have a trivial standard approximation scheme for $Max\ HPP_{s,t}[n; n+5]$; however, for the differential ratio, the *Best Neighbor Heuristic* is not approximable with any constant ratio for $Max\ HPP_s$ and $Max\ HPP_{s,t}[n; n+5]$ is not approximable with ratio greater than $649/650$. We present along this paper many examples to illustrate the difference between these two measures.

We now give some standard definitions in the field of optimization and approximation theory. For a more detailed statement of this theory, we refer the reader to Ausiello et al. [2], [1], Hochbaum [16].

Definition 1.1 An *NP-optimization problem* $\pi \in NPO$ is a five-tuple $(D, sol, m, Triv, goal)$ such that:

- (i) D is the set of instances and is recognizable in polynomial-time.
- (ii) Given an instance $I \in D$, $sol[I]$ is the set of feasible solutions of I ; moreover, there exists a polynomial P such that, for any $x \in sol[I]$, $|x| \leq P(|I|)$; furthermore, it is decidable in polynomial time whether $x \in sol[I]$ for any I and for any x such that $|x| \leq P(|I|)$. Finally, there is a feasible solution $Triv(I)$ ¹ computable in polynomial-time for any I .
- (iii) Given an instance I and a solution x of I , $m[I, x]$ denotes the positive integer value of x . The function m is computable in polynomial time and is also called the objective function.
- (iv) $goal \in \{Max, Min\}$.

◇

¹The common definition of class *NPO* does not require the existence of a trivial solution.

We call $\overline{goal}^2$ the complementary notion of goal and $\bar{\pi}$ the *NPO*-problem $(D, sol, m, Triv, \overline{goal})$. The goal of an *NPO*-optimization problem with respect to an instance I is to find an *optimum solution* x^* such that $OPT(I) = m[I, x^*] = goal\{m[I, x] : x \in sol[I]\}$. Another important solution of π is a *worst solution* x_* defined by: $WOR(I) = m[I, x_*] = \overline{goal}\{m[I, x] : x \in sol[I]\}$. A worst solution for π is an optimal solution for $\bar{\pi}$ and vice versa. In Ausiello et al. [3], the term *trivial solution* referred to as *worst solution* and all the exposed examples have the property that a worst solution can be trivially computed in polynomial-time. For example, this is the case of the maximum Cut problem where, given a graph, the worst solution is the empty edge-set, or the Bin-Packing problem where we can trivially put the items using a distinct bin per item. On the contrary, since a worst solution of the maximum weight Hamiltonian path from s to t is an optimal solution of its corresponding minimization version, the computation of such a solution is far from being trivial! The same property occurs for the f -depth Spanning Tree problem [22] or all the problems exposed in [20].

Approximate algorithms and reductions. In order to study algorithms performances, there are two known measures: *standard ratio* [13], [2], [6] and *differential ratio* [10], [3] and [5].

Definition 1.2 Let π be an *NPO* problem and x be a feasible solution of an instance I . We define the performance ratios of x with respect to the instance I as

- (standard ratio) $\rho[\pi](I, x) = \text{Min}\left\{\frac{m[I, x]}{OPT(I)}, \frac{OPT(I)}{m[I, x]}\right\}$

- (differential ratio) $\delta[\pi](I, x) = \frac{WOR(I) - m[I, x]}{WOR(I) - OPT(I)}$

◇

The performance ratio is a number less than or equal to 1, and is equal to 1 if and only if $m[I, x] = OPT(I)$. Remark that, compared to some definitions, we have inverted the standard performance ratio in the case of minimization problems so that the ratio value is always between 0 and 1. Let π be an *NPO* problem. For any instance I of π , a polynomial time algorithm A returns a feasible solution x^A . The performance of A with respect to $R \in \{\delta, \rho\}$ on the instance I is the quantity $R_A[\pi](I) = R[\pi](I, x^A)$. We say that A is an r -approximation algorithm with respect to R if for any instance I , we have $R_A(I) \geq r$.

Definition 1.3 For any performance ratio $R \in \{\delta, \rho\}$,

- an *NPO* problem belongs to the class $APX(R)$ if there exists an r -approximation with respect to R for some constant $r \in]0; 1]$.
- an *NPO* problem belongs to the class $PTAS(R)$ if there exists an r -approximation A_r for any constant $r \in]0; 1]$. The family $\{A_r\}_{0 < r < 1}$ is said to be an approximation scheme.

◇

Clearly, the following inclusion holds for any measure $R \in \{\delta, \rho\}$: $PTAS(R) \subseteq APX(R)$. As it is usually done, we will denote by APX and $PTAS$, respectively, the classes $APX(\rho)$ and $PTAS(\rho)$. We could argue whether the differential ratio is really pertinent: the authors of [10] and [3] answered positively to that question and concluded that this measure is complementary with the standard ratio. As shown in [8], many problems can have different behavior patterns depending on whether the differential or standard ratio is chosen: consider for instance Vertex Covering or Dominating Set problems. On the other hand, there are problems that establish some connections between the differential and the standard ratios, like Bin Packing [9] or maximum weight bounded-depth spanning tree [22]. Besides, we show that there are

²If $goal = Max$, then $\overline{goal} = Min$ and $\overline{\overline{goal}} = Max$.

tight links between both measures for the problems dealt with in the case where the edge-weights have lower and upper bounds.

Now, consider the following approximation preserving reductions between pairs (π, R) .

Definition 1.4 For $\pi_i \in NPO$ and $R_i \in \{\delta, \rho\}$, $i = 1, 2$,

◦ an *A-reduction* from (π_1, R_1) to (π_2, R_2) , denoted by $(\pi_1, R_1) \leq^A (\pi_2, R_2)$, is a triplet (α, f, c) such that:

- (i) $\alpha: D_{\pi_1} \mapsto D_{\pi_2}$, transforms an instance of π_1 into an instance of π_2 in polynomial-time.
 - (ii) $f: sol_{\pi_2}[\alpha(I)] \mapsto sol_{\pi_1}[I]$, transforms solutions for π_2 into solutions for π_1 in polynomial-time.
 - (iii) $c: [0; 1] \mapsto [0; 1]$ (called *expansion of the A-reduction*) is a function verifying $c^{-1}(0) \subseteq \{0\}$ and $\forall \varepsilon \in [0; 1], \forall I \in D_{\pi_1}, \forall x \in sol_{\pi_2}[\alpha(I)]: R_2[\pi_2](\alpha(I), x) \geq \varepsilon \implies R_1[\pi_1](I, f(x)) \geq c(\varepsilon)$
- an *A * P-reduction* from the pair (π_1, R_1) to the pair (π_2, R_2) , denoted by $(\pi_1, R_1) \leq^{A*P} (\pi_2, R_2)$, is an A-reduction from (π_1, R_1) to (π_2, R_2) such that the restriction of function c to some interval $[a; 1]$ is bijective and $c(1) = 1$ (note that $c(\varepsilon)$ does not necessarily verify $c^{-1}(0) \subseteq \{0\}$). If $(\pi_i, R_i) \leq^{A*P} (\pi_{3-i}, R_{3-i})$ with $c(\varepsilon) = \varepsilon$ for $i = 1$ and $i = 2$, we say that (π_1, R_1) is equivalent to (π_2, R_2) . $\diamond$

An A-reduction preserves constant approximation while A * P-reduction preserves approximation schemes. They are a natural generalization of those described by Orponen and Mannila [23] and Crescenzi and Panconesi [7].

The differential ratio measures how the value of an approximate solution is located in the interval between $WOR(I)$ and $OPT(I)$ whereas for a maximization problem, the usual ratio measures how the value of an approximate solution is placed in the interval between 0 and $OPT(I)$. Hence, for a maximization problem, we have an A * P-reduction from the usual ratio to the differential ratio since $WOR(I) \geq 0$:

Lemma 1.5 If $\pi = (D, sol, m, Triv, Max) \in NPO$, then $(\pi, \rho) \leq^{A*P} (\pi, \delta)$ with $c(\varepsilon) = \varepsilon$.

Remark that, in general, there is no evident transfer of a positive or negative result from one framework to the other.

2 The Hamiltonian path problem

The Hamiltonian path problem, also called the *Traveling Salesman Path problem*, is formally defined as follows.

Definition 2.1 Consider a complete graph K_n with non-negative costs $d(x, y)$ for each vertices pairs of K_n . We want to find an optimal-cost Hamiltonian path, where the cost of a path is the sum of the weights on its edges. We refer this problem as HPP. When one endpoint s (resp. two endpoints s and t) of Hamiltonian path are specified, we use the notation HPP_s (resp. $HPP_{s,t}$).

If goal = Max, the problem is called Max HPP, else Min HPP. We use the notation HPP , HPP_s or $HPP_{s,t}$ when we consider without distinction the cases goal = Max or goal = Min. $\diamond$

For the Hamiltonian Path problem with non specified endpoints, a lot of standard ratio approximation results can be derived for TSP (with goal = Min and goal = Max) thanks to a structural property of solutions and a trivial reduction to the Traveling Salesman problem: the first negative approximation

result (that we can deduce from [26]) states that it is not possible to approximate *Min HPP* within $1/f(|I|)$ where f is any integer function computable within polynomial time unless $P = NP$. On the other hand, *Min metric*³ *HPP* is approximable within $2/3$ [4] and *Min HPP*[1, 2] is APX-complete (deduce from Papadimitriou and Yannakakis [24]). For *Max HPP*, the results are more optimistic since this problem is in APX. The best-known standard ratio is equal to $3/4$ and can be deduced from [27].

The Hamiltonian Path problems with specified endpoints (resp. *metric HPP_s* or *metric HPP_{s,t}*) are as hard to approximate as *HPP* (resp. *metric HPP*). The question put by Johnson and Papadimitriou [18] on the relative hardness of the specified endpoint version compared to the non-specified is open. These conjectures are strengthened by the positive results given on these problems since the best-known standard ratios are respectively $2/3$ and $3/5$ for *Min HPP_s* [17] and *Min HPP_{s,t}* [17], [14]. Moreover, if we consider the case $a \leq d(e) \leq 2a$ there are no specific results. For example Christofides' modification algorithm [17] remain a $2/3$ -standard ratio for *Min HPP_s*[$a; 2a$]. To our knowledge, no standard approximation result has been found for *Max HPP_s* and *Max HPP_{s,t}*.

We show that *HPP_s* is $2/3$ -approximable and *HPP_{s,t}* is $1/2$ -approximable under the differential framework. We can deduce from lemma (1.5) a $2/3$ -standard approximation for *Max HPP_s* and a $1/2$ -standard approximation for *Max HPP_{s,t}*. Moreover, our technique allows to handle the case where all the edge weights are within an interval $[a, 2a]$ for any positive a , and more generally the case $WOR(I) \geq \frac{1}{2}OPT(I)$. Since we give a $3/4$ (resp. $2/3$)-standard approximation for *Min HPP_s*[$a, 2a$] (resp. *Min HPP_{s,t}*[$a, 2a$]), we improve the best-known bounds of $2/3$ (resp. $3/5$) for minimization versions given by Hoogeveen [17] (resp. Guttmann-beck et al. [14] or [17]).

3 Elementary properties

We present some relations between *HPP*, *HPP_s*, *HPP_{s,t}* and different subcases. We prove that *HPP_{s,t}* is the most general problem. As a second step, we establish for each problem some connected relations between differential and standard ratios. In the following paragraph, without specification, the properties that we present for *HPP_{s,t}* are also true for *HPP_s* and *HPP*.

HPP_{s,t} is as hard as *HPP_s* (which is itself as hard as *HPP*) to approximate for both performance ratios. Moreover, from a differential approximability point of view, these different versions are very closed to the Traveling Salesman problem, even if we consider the restriction $a \leq d(e) \leq b$.

Lemma 3.1 *for any goal $\in \{Min, Max\}$, we have:*

- (i) $(goal \ TSP, \delta) \leq^{A*P} (goal \ HPP_{s,t}, \delta)$ with $c(\varepsilon) = \varepsilon$.
- (ii) $(goal \ HPP, \delta) \leq^{A*P} (goal \ TSP, \delta)$ with $c(\varepsilon) = \varepsilon$.

Proof: We only show the case $goal = Max$. Let $I = (K_n, d)$ be an instance.

• For (i): Choose a vertex s in K_n and define $I_v = (K_n, s, v, d)$ for all $v \in V \setminus \{s\}$. If μ_v is a Hamiltonian path from s to v of I_v , we construct the Hamiltonian cycle $\Gamma_v = \mu_v \cup \{(s, v)\}$. Then, we take $\Gamma = argmax\{m[I, \Gamma_v] : v \in V \setminus \{s\}\}$. Now, consider v^* such that $OPT(I_{v^*}) + d(s, v^*) = OPT(I)$; if $m[I_{v^*}, \mu_{v^*}] \geq \varepsilon OPT(I_{v^*}) + (1 - \varepsilon)WOR(I_{v^*})$, then $m[I, \Gamma] \geq m[I, \mu_{v^*}] + d(s, v^*) \geq \varepsilon OPT(I) + (1 - \varepsilon)WOR(I)$.

• For (ii): Transform I into instance $\alpha(I) = (K_{n+1}, d')$ as follow: add a new vertex s to graph K_n and define $d'(s, v) = a, \forall v$, $d'(e) = d(e)$ for other edges. Then, the proof is similar. $\square$

³Verifying for all vertices x, y, z the inequality: $d(x, y) \leq d(x, z) + d(z, y)$.

Note that the proof of item (ii) also holds for the standard ratio with $d'(s, v) = 0$ and we can deduce from [21] and item (ii) of lemma (3.1) that HPP is $2/3$ -differential approximable and $HPP[1, 2]$ is $3/4$ -differential approximable. $HPP_{s,t}$ is NP -hard if and only if so is $\overline{HPP}_{s,t}$, hence, computing a worst solution of $HPP_{s,t}$ is as hard as computing an optimal one. For $goal = Min$, we also have the well-known result stemming from Sahni and Gonzales [26] which affirms that if $Min HPP_{s,t} \in APX$, then the following decision problem is polynomial: given a simple graph $G = (V, E)$, does $E' \subseteq E$ exists such that E' is a Hamiltonian path from s to t of G ? Thus, we deduce that $Min HPP_{s,t}$ is not in APX unless $P = NP$, because the associated decision problem is NP -Complete.

Yet, the asymmetry in the approximability of both versions ($Max HPP_{s,t}$ is in APX as later proved) can be considered as somewhat strange given the structural symmetry existing between them. Since differential approximation is stable under an affine transformation of the objective function, $Max HPP_{s,t}$ and $Min HPP_{s,t}$ are differential-equivalent⁴. Besides, another difference with the standard ratio is that $Min HPP_{s,t}$ is not more difficult than the same problem with triangular inequality (called metric).

Proposition 3.2 *The following assertions hold:*

- (i) $Min HPP_{s,t}$ is differential-equivalent to $Max HPP_{s,t}$.
- (ii) $Min HPP_{s,t}[a, b]$ is differential-equivalent to $Max HPP_{s,t}[a, b]$.
- (iii) $HPP_{s,t}$ is differential-equivalent to metric $HPP_{s,t}$.

Proof : Let $I = (K_n, s, t, d)$ be an instance, set $d_{max} = \max_{e \in E} d(e)$ and $d_{min} = \min_{e \in E} d(e)$.

• For (i) and (ii): $(Min HPP_{s,t}, \delta) \leq^{A*P} (Max HPP_{s,t}, \delta)$ with $c(\varepsilon) = \varepsilon$.

We transform I into instance $\alpha(I) = (K_n, s, t, d')$ defined by: $\forall e \in E, d'(e) = d_{max} + d_{min} - d(e)$.

For any Hamiltonian path μ from s to t , we have: $m[\alpha(I), \mu] = (|V| - 1)(d_{max} + d_{min}) - m[I, \mu]$.

Hence if $m[\alpha(I), \mu] \geq \varepsilon OPT(\alpha(I)) + (1 - \varepsilon)WOR(\alpha(I))$, then $m[I, \mu] \leq \varepsilon OPT(I) + (1 - \varepsilon)WOR(I)$.

Conversely, the proof is similar since $\alpha \circ \alpha = Id$.

• For (iii), the proof is similar, except that function d' is now defined by $d'(e) = d_{max} + d(e)$. $\square$

The following easy theorem holds, thus giving a bridge between differential and standard ratios for $goal = Max$ and $goal = Min$, in the case where edge weights belong to an interval $[a, b]$.

Theorem 3.3 $(goal HPP_{s,t}[a, b], \rho) \leq^{A*P} (goal HPP_{s,t}[a, b], \delta)$ with the expansion verifying:

$$\begin{aligned} \bullet \quad c_1(\varepsilon) &= \frac{(b-a)\varepsilon}{b} + \frac{a}{b} \quad \text{if } goal = Max \\ \bullet \quad c_2(\varepsilon) &= \frac{a}{b - (b-a)\varepsilon} \quad \text{if } goal = Min \end{aligned}$$

Proof : We only prove the $goal = Max$ case. Let I be an instance and μ be a Hamiltonian path from s to t .

If $m[I, \mu] \geq \varepsilon OPT(I) + (1 - \varepsilon)WOR(I)$, then $m[I, \mu] \geq c_1(\varepsilon)OPT(I)$ since $WOR(I) \geq \frac{a}{b}OPT(I)$. $\square$

$HPP_{s,t}[a, b]$ (for a and b not depending on the instance) is "easy" to approximate with standard framework (i.e., $HPP_{s,t}[a, b] \in APX$), since even a worst solution is an a/b -approximation (take $\varepsilon = 0$ in theorem (3.3)), while this is not true with the differential measure. Nevertheless, we can deduce from

⁴see definition (1.4)

this theorem that the hardness thresholds for standard and differential framework are identical since $\text{Min HPP}_{s,t}[a, b]$ is *APX - complete*.

Corollary 3.4 *For all $b > a \geq 0$, $\text{HPP}_{s,t}[a, b] \notin \text{PTAS}(\delta)$ unless $P = NP$.*

For some values of a and b , we can also establish a limit on its differential approximation. Keep in mind the negative result of [11]: for any $\epsilon > 0$, no polynomial time algorithm can guarantee a standard approximation ratio greater than, or equal to, $\frac{130}{131} + \epsilon$ for $\text{TSP}[1, 6]$. So, using item(i) of lemma(3.1) and lemma(1.5), we deduce:

Proposition 3.5 *For all $a \geq 0$, $\text{HPP}_{s,t}[a, a + 5]$ is not approximable with differential ratio greater than $\frac{649}{650}$ unless $P = NP$.*

Note that this proposition does not hold for HPP and HPP_s .

4 Approximate algorithms for these problems

In this section, we propose two types of algorithms which yield constant ratio. For Max HPP_s , the algorithm is obtained by getting several feasible solutions and by choosing the best one among them. Each of these individual solutions has a differential approximation ratio tending towards zero with the size of the instance. For $\text{Max HPP}_{s,t}$, the algorithm is very different and takes into account the extreme solutions. So, on the one hand, the algorithm tries to be the nearest from the best solution value and on the other hand, tries to be the furthest from the worst solution value. In order to do that, it iteratively provides a solution of value greater than $(\text{WOR}(I_j) + \text{OPT}(I_j))/2$, where I_j is the sub-graph built at step j .

4.1 The algorithm for two specified endpoints version

$\text{Max HPP}_{s,t}$ can also be regarded as the problem of determining a Hamiltonian cycle that contains edge (s, t) . The algorithm works by finding a maximum weight 2-matching among 2-matchings containing (s, t) and at each step, merge two by two the cycles. The main idea consists in pointing out that we could have lost much more by merging in a different way the two cycles at each iteration. Thus, we will build dynamically a bad solution that will actually depend on the choices made by the algorithm at each iteration. Consider two cycles Γ_i and two edges $(x_1, x_2) \in \Gamma_1$ and $(y_1, y_2) \in \Gamma_2$, we call localchange_i for $i = 1, 2$ the following process:

$$\text{localchange}_i[(\Gamma_1, (x_1, x_2)), (\Gamma_2, (y_1, y_2))] = \{(x_1, y_{3-i}), (x_2, y_i)\} \cup (\Gamma_1 \cup \Gamma_2 \setminus \{(x_1, x_2), (y_1, y_2)\})$$

These two processes merge the cycles Γ_1 and Γ_2 into a unique cycle. Remark that the vertices order is important in the processes; thus, edges (x_1, x_2) or (y_1, y_2) are implicitly given as directed edges and we have: $\text{localchange}_1[(\Gamma_1, (x_2, x_1)), (\Gamma_2, (y_1, y_2))] = \text{localchange}_2[(\Gamma_1, (x_1, x_2)), (\Gamma_2, (y_1, y_2))]$. Moreover, when $\Gamma_1 = \Gamma_2$ and (x_1, x_2) is not adjacent to (y_1, y_2) , these processes amount to simply local edges swaps. We associate to localchange_i a function cost_i that represents the loss in merging two cycles:

$$\text{cost}_i[(x_1, x_2), (y_1, y_2)] = d(x_1, x_2) + d(y_1, y_2) - d(x_1, y_{3-i}) - d(x_2, y_i)$$

[$\text{LocalchangeHPP}_{s,t}$]

input: An instance (K_n, s, t, d) ;

output : A Hamiltonian path SOL from s to t of G ;

Change the cost of (s, t) into $|V|d_{max} + 1$. Call d' this function ;

Compute a maximum weight 2-matching $M = \{\Gamma_i, i = 1, \dots, k\}$ of (K_n, d') ;

Suppose that $(s, t) \in \Gamma_1$

Choose 2 consecutive edges (x_1^1, x_2^1) and (x_2^1, x_3^1) in Γ_1 different from (s, t) ;

$sol_1 = \Gamma_1$, $e_1^1 = (x_1^1, x_2^1)$ and $e_2^1 = (x_2^1, x_3^1)$;

For $i=2$ to k do

Choose 2 consecutive edges (x_1^i, x_2^i) and (x_2^i, x_3^i) in Γ_i ;

If $cost_1[e_1^{i-1}, (x_1^i, x_2^i)] \leq cost_2[e_2^{i-1}, (x_2^i, x_3^i)]$ then

$sol_i = localchange_1[(sol_{i-1}, e_1^{i-1}), (\Gamma_i, (x_1^i, x_2^i))]$;

Suppose $e_1^{i-1} = (x, y)$, and x_0^i is the other neighbor of x_1^i in Γ_i

Set $e_1^i = (y, x_1^i)$ and $e_2^i = (x_1^i, x_0^i)$;

Else

$sol_i = localchange_2[(sol_{i-1}, e_2^{i-1}), (\Gamma_i, (x_2^i, x_3^i))]$;

Suppose $e_2^{i-1} = (x, y)$, and x_4^i is the other neighbor of x_3^i in Γ_i

Set $e_1^i = (y, x_3^i)$ and $e_2^i = (x_3^i, x_4^i)$;

End if ;

End for i ;

$SOL = sol_k \setminus \{(s, t)\}$;

As this algorithm is polynomial, let us then show that SOL is an Hamiltonian path. First, remark that by construction, (s, t) belongs to every maximum weight 2-matching of (K_n, d') . Moreover, e_1^i and e_2^i obviously belong to sol_i for every iteration $i \leq k$ of the algorithm. These two facts lead to the result.

Theorem 4.1 *The algorithm $[LocalchangeHPP_{s,t}]$ is a $\frac{1}{2}$ -differential approximation for $Max HPP_{s,t}$ and this ratio is tight.*

Proof: Given $I = (K_n, s, t, d)$, an instance of $Max HPP_{s,t}$, we denote $(i_2, \dots, i_k)$ with $i_j \in \{1, 2\}$ the sequence of choices produced by the algorithm such that, for $j \in \{2, \dots, k\}$:

$$sol_j = localchange_{i_j}[(sol_{j-1}, e_{i_j}^{j-1}), (\Gamma_j, (x_{i_j}^j, x_{i_j+1}^j))]$$

Thus, $d(sol_j) = d(sol_{j-1}) + d(\Gamma_j) - cost_{i_j}(j)$ with $cost_{i_j}(j) = cost_{i_j}[e_{i_j}^{j-1}, (x_{i_j}^j, x_{i_j+1}^j)]$. Summing up these equalities for $j = 2$ to k , and since $d(sol_1) = d(\Gamma_1)$ and $d(SOL) = d(sol_k) - d(s, t)$, we obtain:

$$d(SOL) = d(M) - d(s, t) - \sum_{j=2}^k cost_{i_j}(j) \quad (4.1)$$

The main idea is to note that edge-subset $\{e_{3-i_j}^{j-1}, (x_{3-i_j}^j, x_{4-i_j}^j) : j = 2, \dots, k\}$ belongs to solution sol_k . Hence, we can "damage" the current solution by local edges-swap from this edge-subset. More formally, consider solutions sol'_j defined by $sol'_1 = sol_k$ and for $j = 2, \dots, k$,

$$sol'_j = localchange_{3-i_j}[(sol'_{j-1}, e_{3-i_j}^{j-1}), (sol'_{j-1}, (x_{3-i_j}^j, x_{4-i_j}^j))]$$

Finally, proceeding as previously, we obtain $d(sol'_k) = d(M) - \sum_{j=2}^k (cost_{i_j}(j) + cost_{3-i_j}(j))$. By construction, $cost_{i_j}(j) + cost_{3-i_j}(j) \geq 2cost_{i_j}(j)$ and $WOR(I) \leq d(sol'_k) - d(s, t)$. Hence:

$$WOR(I) \leq d(M) - d(s, t) - 2 \sum_{j=2}^k cost_{i_j}(j) \quad (4.2)$$

Remark that M is an optimal weight 2-matching among the 2-matching of (G, d) containing the edge (s, t) ; thus

$$OPT(I) \leq d(M) - d(s, t) \quad (4.3)$$

By combining expressions (4.3), (4.2) and (4.1), we obtain:

$$d(SOL) \geq \frac{1}{2}OPT(I) + \frac{1}{2}WOR(I)$$

We now show that this ratio is tight. Let $J_n = (K_n, s, t, d)$ be an instance defined by: $V = (\{x_i^j, 1 \leq i \leq 3, 2 \leq j \leq 2n+1\} \cup \{s, u, t\})$, $d(x_1^j, x_1^{j+1}) = d(x_1^j, x_2^{j+1}) = 1 \forall j = 2, \dots, 2n$, $d(x_1^j, x_2^{j+2}) = 1 \forall j = 2, \dots, 2n-1$, $d_n(s, x_2^2) = d_n(u, x_1^2) = d(u, x_2^2) = d(t, x_3^2) = 1$ and let the cost of all other edges be two. The 2-matching is composed of $\Gamma_1 = \{s, u, t\}$ and $\Gamma_j = \{x_1^j, x_2^j, x_3^j\}$ $j = 2, \dots, 2n+1$. The edges produced by the algorithm are: $e_1^1 = (s, u)$, $e_2^1 = (u, t)$, $e_1^2 = (u, x_1^2)$, $e_2^2 = (x_1^2, x_3^2)$, $e_1^j = (x_1^{j-1}, x_1^j)$, $e_2^j = (x_1^j, x_3^j)$ $j = 3, \dots, 2n+1$ and $cost_1(2) = cost_2(2) = 2$, $cost_1(j) = cost_2(j) = 1$ $j = 3, \dots, 2n+1$.

$$d(SOL) = 10n + 4, WOR(J_{2n+1}) = 8n + 3, OPT(J_{2n+1}) = 12n + 4$$

Thus, we obtain $\delta_{LocalchangeHPP_{s,t}}(J_{2n+1}) \rightarrow \frac{1}{2}$. □

For the standard ratio, we deduce new improved results from lemma (1.5) and from theorem (3.3):

Corollary 4.2 *We have the following results:*

- *Max $HPP_{s,t}$ is $\frac{1}{2}$ -standard approximable and Max $HPP_{s,t}[a, 2a]$ is $\frac{3}{4}$ -standard approximable.*
- *Min $HPP_{s,t}[a, 2a]$ is $\frac{2}{3}$ -standard approximable.*

4.2 The algorithm for one specified endpoint version

We propose an algorithm which differs from the one previously studied since we explicitly compute several solutions. Our algorithm is based upon a simple idea and uses structural properties of solutions. It still works by finding a maximum weight 2-matching containing specified edges and then discarding some edges and arbitrarily connecting the resulting paths to form an Hamiltonian path from s . The principle of our algorithm is to generate not only one but several feasible solutions following this method.

Consider a maximum weight 2-matching M_r among those containing (s, r) , including elementary cycles Γ_i , $i = 1, \dots, k$. In order to do that, we substitute $|V|d_{max} + 1$ for the cost of (s, r) and we compute a maximum 2-matching in this new graph. Finally, for each cycle Γ_i , we consider four consecutive vertices $x_1^i, x_2^i, x_3^i, x_4^i$. Remark that we have numbered vertices such that $x_1^1 = r$ and $x_2^1 = s$. Moreover, if $|\Gamma_i| = 3$ then $x_4^i = x_1^i$. For the last cycle Γ_k , we consider an additional vertex y which is the other neighbor of x_1^k in Γ_k . Thus, if $|\Gamma_k| = 4$ then $y = x_4^k$ while y is a new vertex in the other case.

[Patching 2-matching]

input: An instance (K_n, s, d) ;

output: A Hamiltonian path SOL from s of K_n ;

For any $r \in V \setminus \{s\}$ do

Change the cost of (s, r) into $|V|d_{max} + 1$. Call d' this function;

Compute a maximum weight 2-matching $M_r = \{\Gamma_i, i = 1, \dots, k\}$ of (K_n, d') ;

if $k = 1$ then $SOL_r = M_r \setminus \{(s, r)\}$;

if k is even then

$$S_1 = \cup_{j=1}^{k-1} \{(x_2^j, x_3^j)\} \cup \{(x_1^k, x_2^k), (s, r)\} ;$$

build $sol_1 = (M_r \setminus S_1) \cup \{(x_1^k, x_3^1), (x_2^1, x_2^2)\} \cup_{j=1}^{(k-2)/2} \{(x_3^{2j}, x_3^{2j+1}), (x_2^{2j+1}, x_2^{2j+2})\}$;
 (sol_1 is a Hamiltonian path from s to r)
 $S_2 = \cup_{j=2}^{k-1} \{(x_1^j, x_2^j)\} \cup \{(y, x_1^k), (s, r)\}$;
 build $sol_2 = (M_r \setminus S_2) \cup \{(x_1^1, x_1^2)\} \cup_{j=1}^{(k-2)/2} \{(x_2^{2j}, x_2^{2j+1}), (x_1^{2j+1}, x_1^{2j+2})\}$;
 (sol_2 is a Hamiltonian path from s to y)
 $S_3 = \cup_{j=1}^{k-1} \{(x_3^j, x_4^j)\} \cup \{(x_2^k, x_3^k), (s, r)\}$;
 build $sol_3 = (M_r \setminus S_3) \cup \{(x_2^k, x_4^1), (x_3^1, x_3^2)\} \cup_{j=1}^{(k-2)/2} \{(x_4^{2j}, x_4^{2j+1}), (x_3^{2j+1}, x_3^{2j+2})\}$;
 (sol_3 is an Hamiltonian path from s to r)

End if ;

if k is odd then

$S_1 = \cup_{j=1}^k \{(x_2^j, x_3^j)\} \cup \{(s, r)\}$;
 build $sol_1 = (M_r \setminus S_1) \cup \{(x_2^k, x_3^1)\} \cup_{j=1}^{(k-1)/2} \{(x_2^{2j-1}, x_2^{2j})(x_3^{2j}, x_3^{2j+1})\}$;
 (sol_1 is an Hamiltonian path from s to r)
 $S_2 = \{(s, r)\} \cup_{j=2}^k \{(x_1^j, x_2^j)\}$;
 build $sol_2 = (M_r \setminus S_2) \cup_{j=1}^{(k-1)/2} \{(x_1^{2j-1}, x_1^{2j}), (x_2^{2j}, x_2^{2j+1})\}$;
 (sol_2 is a Hamiltonian path from s to x_1^k)
 $S_3 = \cup_{j=1}^k \{(x_3^j, x_4^j)\} \cup \{(s, r)\}$;
 build $sol_3 = (M_r \setminus S_3) \cup \{(x_3^k, x_4^1)\} \cup_{j=1}^{(k-1)/2} \{(x_3^{2j-1}, x_3^{2j}), (x_4^{2j}, x_4^{2j+1})\}$;
 (sol_3 is a Hamiltonian path from s to r)

End if ;

$SOL_r = \operatorname{argmax}\{d(sol_1), d(sol_2), d(sol_3)\}$;

End for r ;

$SOL = \operatorname{argmax}\{d(SOL_r) : r \in V \setminus \{s\}\}$;

Observe that for any r , the solutions sol_1 , sol_2 and sol_3 are Hamiltonian paths (from s to different endpoints) since the additional edges are adjacent to the ones substituted. The time-complexity of this algorithm is greater than the time-complexity of the previous algorithm but remains polynomial since the computation of the 2-matching problem is polynomial.

Theorem 4.3 *The algorithm [Patching 2-matching] is a $\frac{2}{3}$ -differential approximation for Max HPP_s and this ratio is tight.*

Proof: Let $I = (K_n, s, d)$ be an instance and let SOL^* be an optimal Hamiltonian path from s to r^* . We denote $loss_i$, $i = 1, 2, 3$, the quantity $d(sol_i) - d(M_{r^*}) + d(s, r^*)$. Obviously, $loss_i \leq 0$ and we have

$$d(SOL) \geq d(SOL_r^*) \geq d(M_{r^*}) - d(s, r^*) + \frac{1}{3}(loss_1 + loss_2 + loss_3) \quad (4.4)$$

Moreover, the following structural property holds:

$SOL_* = \cup_{j=1,2,3} (sol_j \setminus M_{r^*}) \cup M_{r^*} \setminus (S_1 \cup S_2 \cup S_3)$ is a Hamiltonian path starting from s

$d(SOL_*) = d(M_{r^*}) - d(s, r^*) + loss_1 + loss_2 + loss_3$ since $d(sol_j \setminus M_{r^*}) = loss_j + d(S_j) - d(s, r^*)$ and $d(M_{r^*} \setminus (S_1 \cup S_2 \cup S_3)) = d(M_{r^*}) - d(S_1) - d(S_2) - d(S_3) + 2d(s, r^*)$. Hence, we deduce

$$WOR(I) \leq d(M_{r^*}) - d(s, r^*) + loss_1 + loss_2 + loss_3 \quad (4.5)$$

Since $SOL^* \cup (s, r^*)$ is a particular 2-matching containing (s, r^*) , we have:

$$OPT(I) \leq d(M_{r^*}) - d(s, r^*) \quad (4.6)$$

Finally, combining (4.4), (4.5) and (4.6) we obtain:

$$d(SOL) \geq \frac{1}{3}WOR(I) + \frac{2}{3}OPT(I)$$

To show that the bound is approachable, consider the following instances. Let $I_n = (K_n, s, d)$ be an instance defined by: $V = \{x_i^j : 1 \leq i \leq 3, 1 \leq j \leq 2n+1\}$ with $x_2^1 = s$, $d(x_1^j, x_2^j) = d(x_3^j, x_2^j) = d(x_1^j, x_3^j) = 2$, $\forall j = 1, \dots, 2n+1$, $d(x_1^j, x_3^{j+1}) = 2$, $\forall j = 1, \dots, 2n$ and $d(x_1^1, x_3^j) = d(x_1^1, x_2^j) = 2$, $\forall j = 2, \dots, 2n+1$. Let the cost of all other edges be one. We have:

$$d(SOL) \leq 10n + 4, OPT(I_n) = 12n + 4, WOR(I_n) = 6n + 2$$

leading to the conclusion $\lim \delta_{Patching \ 2-matching}(I_n) \leq \frac{2}{3}$. $\square$

For the standard ratio, we deduce new improved results from theorem (3.3):

Corollary 4.4 *We have the following results:*

- *Max HPP_s is $\frac{2}{3}$ -standard approximable and Max HPP_s[a, 2a] is $\frac{5}{6}$ -standard approximable.*
- *Min HPP_s[a, 2a] is $\frac{3}{4}$ -standard approximable.*

5 Towards thresholds of differential non-approximation

A useful technic to establish some standard non-approximation thresholds of *NP*-hard problems consists in establishing the fact they are not *simple*. Recall that an *NPO* problem is called *simple* Paz and Moran [25] if its restriction to instances verifying for any fixed integer k , $OPT(I) \leq k$ can be resolved within polynomial time. So, we can also prove a standard non-approximation threshold equal to $2/3$ for the Bin-Packing problem because its restriction to instances verifying $OPT(I) \leq 2$ is still a *NP*-hard problem. Similarly, we will say that π is δ -*simple* if its restriction π_k to instances verifying for any integer fixed k , $|WOR(I) - OPT(I)| \leq k$ can be solved within polynomial time. Thus, we also have the following proposition:

Proposition 5.1 *If $\pi \in PTAS(\delta)$, then π is δ -simple.*

Proof: Start from a problem $\pi \in NPO$ and build the problem $\phi(\pi)$ defined by:

if $\pi = (D, sol, m, Triv, goal)$ then $\phi(\pi) = (D, sol, m', Triv, Max)$ where $m'[I, x] = |WOR(I) - m[I, x]|$.

⁵ Finally, we use the results of [25] for the standard ratio since we have that $\rho[\phi(\pi)](I, x) = \delta[\pi](I, x)$. $\square$

In other words, problems which are not δ -*simple* do not admit differential approximation scheme and we are even able to establish a differential non-approximation threshold; thus, if the sub-problem verifying $|WOR(I) - OPT(I)| \leq k_0$ is *NP*-hard then for any $\epsilon > 0$, no polynomial time algorithm can guarantee a differential approximation ratio greater than, or equal to $\frac{k_0}{k_0+1} + \epsilon$ for π . The proposition above allows to achieve better hardness thresholds given by k_0 . We conjecture that $HPP_{s,t}$ is not δ -*simple* for small values of k_0 . If this was true, the threshold of proposition (3.5) could be meaningfully improved. Anyway, that will allow us to better identify the space of feasible solutions and to understand how we can (or not) polynomially navigate in it.

⁵Observe that the problem $\phi(\pi)$ is not in *NPO* anymore when $WOR(I)$ is not computable in polynomial time, which is the case in this paper.

References

- [1] 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 Verlag, 1999.
- [2] G. AUSIELLO, P. CRESCENZI, and M. PROTASI, *Approximate solutions of NP-optimization problems*, Theo. Comp. Sci. **150** (1995), 1–55.
- [3] G. AUSIELLO, A. D’ATRI, and M. PROTASI, *Structure preserving reductions among convex optimization problems*, J. Comp. Sys. Sci. **21** (1980), 136–153.
- [4] N. CHRISTOFIDES, *Worst-case analysis of a new heuristic for the traveling salesman problem*, Technical report 338, Grad. School of Industrial Administration, CMU, 1976.
- [5] G. CORNUEJOLS, M. L. FISHER, and G. L. NEMHAUSER, *Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms*, Manag. Sci. **23** (1977), 789–810.
- [6] P. CRESCENZI and V. KANN, *A compendium of NP-optimization problems*, Available on www address: <http://www.nada.kth.se/~viggo/problemlist/compendium.html>, 2000.
- [7] P. CRESCENZI and A. PANCONESI, *Completeness in approximation classes*, Inf. and Comp. **93** (1991), 241–262.
- [8] M. DEMANGE, P. GRISONI, and V. Th. PASCHOS, *Differential approximation algorithms for some combinatorial optimization problems*, Theo. Comp. Sci. **209** (1998), 107–122.
- [9] M. DEMANGE, J. MONNOT, and V. Th. PASCHOS, *Bridging gap between standard and differential polynomial approximation: the case of bin-packing*, Appl. Math. Let. **12** (1999), 127–133.
- [10] M. DEMANGE and V. Th. PASCHOS, *On an approximation measure founded on the links between optimization and polynomial approximation theory*, Theo. Comp. Sci. **156** (1996), 117–141.
- [11] L. ENGBRETSSEN and M. KARPINSKI, *Approximation hardness of tsp with bounded metrics*, E.C.C.C. Report TR00 – 089, (2000).
- [12] M. L. FISHER, G. L. NEMHAUSER, and L. A. WOLSEY, *An analysis of approximations for finding a maximum weight hamiltonian circuit*, Op. Res. **27** (1979), no. 4, 799–809.
- [13] M. R. GAREY and D. S. JOHNSON, *Computers and intractability. a guide to the theory of NP-completeness*, CA, Freeman, 1979.
- [14] N. GUTTMANN-BECK, R. HASSIN, S. KHULLER, and B. RAGHAVACHARI, *Approximation algorithms with bounded performance guarantees for the clustered traveling salesman problem*, Algorithmica (to appear), 2000.
- [15] R. HASSIN and S. RUBINSTEIN, *An approximation algorithm for maximum traveling salesman problem*, Inf. Proc. Let. **75** (2000), 181–186.
- [16] D. HOCHBAUM, EDITOR, *Approximation algorithms for NP-hard problems*, P.W.S, 1997.
- [17] J. A. HOOGEVEEN, *Analysis of Christofides’ heuristic: Some paths are more difficult than cycles*, Oper. Res. Let. **10** (1991), 291–295.

- [18] D. S. JOHNSON and C. H. PAPADIMITRIOU, *Performance guarantees for heuristics*, E. L. LAWLER, J. K. LENSTRA, A. H. G. RINNOOY KAN AND D. B. SHMOYS (EDS) *The Traveling Salesman Problem: a guided tour of Combinatorial Optimization* WILEY CHICHESTER 145 – 180, 1985.
- [19] R. M. KARP, *Reductibility among combinatorial problems*, R.E MILLER and J.W. THATCHER, Complexity of Computer Computations, 85-103, Plenum Press, NY, 1972.
- [20] J. MONNOT, *Approximation of NP-hard problems with constant size feasible solutions*, submitted article (2000).
- [21] ———, THE MAXIMUM TRAVELING SALESMAN PROBLEM, submitted article (2000).
- [22] ———, THE MAXIMUM f -DEPTH SPANNING TREE PROBLEM, Inf. Proc. Let. (to appear) (2001).
- [23] P. ORPONEN and H. MANNILA, *On approximation preserving reductions: complete problems and robust measures*, Tech. report, Department of Computer Science, University of Helsinki, 1987.
- [24] C. H. PAPADIMITRIOU and M. YANNAKAKIS, *The traveling salesman problem with distance one and two*, Math. Oper. Res. 18 (1993), 1–11.
- [25] A. PAZ and S. MORAN, *Non deterministic polynomial optimisation problems and their approximation*, Theo. Comp. Sci. 95 (1981), 251–277.
- [26] S. SAHNI and T. GONZALES, *P-complete approximation problems*, J. A.C.M. 23 (1976), 555–565.
- [27] A. I. SERDYUKOV, *An algorithm with an estimate for the traveling salesman problem of the maximum*, Upravlyaemye Sistemy (in Russian) 25 (1984), 80–86.