Worst Case Analysis of Max-Regret, Greedy and Other Heuristics for Multidimensional Assignment and Traveling Salesman Problems

Gregory Gutin*
Department of Computer Science, Royal Holloway University of London
Egham, Surrey TW20 0EX, UK, gutin@cs.rhul.ac.uk, tel: (+44) 1784414229 and
Department of Computer Science, University of Haifa, Israel

Boris Goldengorin
Department of Econometrics and Operations Research, University of Groningen
P. O. Box 800, 9700 AV Groningen, The Netherlands, B.Goldengorin@rug.nl and
Department of Applied Mathematics, Khmelnitsky National University, Ukraine

Jing Huang
Department of Mathematics and Statistics, P.O. Box 3045, University of Victoria
Canada V8W 3P4, jing@math.uvic.ca and
School of Mathematics and Computer Science, Nanjing Normal University, China

Abstract

Optimization heuristics are often compared with each other to determine which one performs best by means of worst-case performance ratio reflecting the quality of returned solution in the worst case. The domination number is a complement parameter indicating the quality of the heuristic in hand by determining how many feasible solutions are dominated by the heuristic solution. We prove that the Max-Regret heuristic introduced by Balas and Saltzman (1991) finds the unique worst possible solution for some instances of the s-dimensional ($s \geq 3$) assignment and asymmetric traveling salesman problems of each possible size. We show that the Triple Interchange heuristic (for $s = 3$) also introduced by Balas and Saltzman and two new heuristics (Part and Recursive Opt Matching) have factorial domination numbers for the s-dimensional ($s \geq 3$) assignment problem.

Keywords: Traveling Salesman Problem, Multidimensional Assignment Problem, Greedy Heuristics, Domination Analysis

*Corresponding author

1 Introduction

The Multidimensional Assignment Problem (abbreviated s-AP in the case of s dimensions) have been introduced by Pierskalla (1968) as a natural extension of 2-AP. General s-APs have recently been considered to model data association problems in connection with multitarget tracking and multisensor surveillance, see Poore (1994) as well as solving centralized multisensor multitarget tracking, see Robertson (2001). Greedy randomized adaptive search (GRASP) heuristics for multidimensional assignment problems arising in multitarget tracking and data association have been proposed by Murphey et al. (1998). Pusztaszeri et al. (1995) describe another interesting s-AP which arises in the context of tracking elementary particles. By solving a 5-AP, they reconstruct tracks of charged elementary particles generated by the Large Electron-Positron Collider at CERN in Geneva. In fact, several applications described in Burkard and Cela (1999), and Robertson (2001) naturally require the use of s-AP for values of s larger than 3.

The Asymmetric Traveling Salesman Problem (ATSP) has a large variety of applications; see, e.g. Punnen (2002). For recent applications of asymmetric and highly nonmetric instances in industry see Gupta et al. (2005) and in bioinformatics see Xu et al. (2005). Most of ATSP research was concentrated on its symmetric special case (see, e.g., Gutin and Punnen (2002)) and more research of the general case heuristics is required (see, e.g., Johnson et al. (2002)).

Both well-known Greedy algorithm and so-far-less-investigated Max-Regret algorithm (see Balas and Salztman (1991), Ghosh et al. (2006), and Robertson (2001)) are fast construction heuristics that build a solution element by element without an attempt to improve it. We perform worst case analysis of Max-Regret for s-AP and ATSP by means of domination analysis.

While computational experiments in Balas and Salztman (1991) show that Max-Regret significantly outperforms Greedy for s-AP ($s \geq 3$), more extensive experiments in Robertson (2001) indicate that neither of the two heuristics dominates the other. This conclusion is confirmed in our paper. Moreover, we prove that Greedy and Max-Regret find the unique worst assignments for some instances of s-AP ($s \geq 3$) of every possible size. We introduce and discuss heuristics that perform much better in the worst case than Greedy and Max-Regret. Such heuristics can be more reliable alternatives to both Greedy and Max-Regret especially when we deal with previously uninvestigated families of s-AP instances.

Experimental results in Ghosh et al. (2006) indicate that a version of Max-Regret, Max-Regret-FC (called R-R-Greedy in Ghosh et al. (2006)), clearly outperforms Greedy for ATSP. Nevertheless, we prove that, like Greedy, both Max-Regret and Max-Regret-FC find the unique worst tour for some instances of ATSP of each possible size. This, in particular, settles the problem of finding good bounds for the domination number of Max-Regret-FC stated in Ghosh et al. (2006).

The paper is organized as follows. We provide basic notions on domination analysis and Greedy in Section 2. In Section 3, we describe Max-Regret for s-AP and prove that, for each $n \geq 1$ and $s \geq 3$, there is an s-AP instance of size n^s for which Max-Regret constructs the unique worst assignment. For 2-AP we only prove that there are instances for which Max-Regret finds an assignment which is worse than at least $n! - 2^{n-1}$ assignments. We conjecture that, in fact, the domination number of Max-Regret for 2-AP is exactly 2^{n-1}. Section 4 is devoted to three s-AP heuristics which *always* find assignments that are not worse that $((n-1)!)^{s-1}$ assignments. Two of the heuristics Part and Recursive Opt Matching are new and might well be of interest in practice. In Section 5 we describe Max-Regret and its version Max-Regret-FC for ATSP and prove that, for each $n \geq 2$, there is an ATSP instance on n vertices for which both heuristics find the unique worst tour. Conclusions and further research appear in Section 6.

2 Domination Analysis and Greedy

Research on combinatorial optimization (CO) heuristics has produced a large variety of heuristics especially for well-known CO problems and, thus, it is important to develop ways of selecting the best ones among them. In most of the literature, heuristics are compared by means of computational experiments and, while experimental analysis is of definite importance, it cannot cover all possible families of instances of the CO problem at hand and, in particular, it usually does not cover the hardest instances. Worst case analysis is normally performed by approximation analysis (see, e.g., Ausiello et el. (1999)), where upper or lower bounds for the worst case performance ratio are of interest. Introduced in Glover and Punnen (1997), domination analysis provides an alternative and a complement to approximation analysis. In domination analysis, we are interested in the domination number or domination ratio of the heuristic solution. We define these parameters below.

Pros and cons of domination analysis are discussed in Gutin and Yeo (2005) and, in our view, it is advantageous to have bounds for both performance ratio and domination ratio of a heuristic whenever it is possible. Roughly speaking this would enable us to see a 2D picture rather than a 1D picture.

Let $\mathcal{P}$ be a minimization CO problem, let $\mathcal{I}$ be an instance of $\mathcal{P}$, let $S(\mathcal{I})$ denote the set of feasible solutions of $\mathcal{I}$, and let H be a heuristic for $\mathcal{P}$. The size of $\mathcal{I}$ is denoted by $|\mathcal{I}|$ and the solution obtained by H for $\mathcal{I}$ is denoted by $H(\mathcal{I})$. When considering the weight of a solution y we write $w(y)$.

The *domination number* of a heuristic H is

$$\mathrm{domn}(H, n) = \min_{\mathcal{I}\in\mathcal{P}:\ |\mathcal{I}|=n} \mathrm{domn}(H, \mathcal{I}),$$

where $\mathrm{domn}(H, \mathcal{I}) = |\{y \in S(\mathcal{I}) :\ w(H(\mathcal{I})) \leq w(y)\}|$.

In other words, the domination number $\mathrm{domn}(H, n)$ is the maximum integer such that the solution $H(\mathcal{I})$ obtained by H for *any* instance $\mathcal{I}$ of $\mathcal{P}$ of size n is not worse than at least $\mathrm{domn}(H, n)$ feasible solutions of $\mathcal{I}$ (including $H(\mathcal{I})$). The *domination ratio* of H is

$$\mathrm{domr}(H, n) = \min_{\mathcal{I} \in \mathcal{P}:\ |\mathcal{I}|=n} \frac{\mathrm{domn}(H, \mathcal{I})}{|S(\mathcal{I})|}.$$

In many cases, domination analysis is very useful. For example, the greedy algorithm has domination number 1 for many CO problems, (see, e.g., Punnen and Kabadi (2002)). In other words, the greedy algorithm, in the worst case, produces the unique worst possible solution. This is reflected in computational experiments with the greedy algorithm for the asymmetric traveling salesman problem (ATSP), (see, e.g., Johnson et al. (2002)), where it was concluded that the greedy algorithm 'might be said to self-destruct.' The fact that the greedy algorithm is of domination number 1 for s-AP ($s \geq 3$) as well (see Theorem 3.2) implies that the algorithm should be used with great care for s-AP. Bounds for domination numbers/ratios were obtained for many CO heuristics; see, e.g., Berend et al. (2006), Gutin and Yeo (2005), Koller and Noble (2004), and Punnen et al. (2003).

Many CO problems can be formulated as follows. We are given a pair $(E, \mathcal{F})$, where E is a finite set and $\mathcal{F}$ is a family of subsets of E, and a weight function w that assigns a real weight $w(e)$ to every element of E. A maximal (with respect to inclusion) set $B \in \mathcal{F}$ is called a *base*. The weight $w(S)$ of $S \in \mathcal{F}$ is defined as the sum of the weights of the elements of S. The objective is to find a base $B \in \mathcal{F}$ of minimum weight.

The well-known Greedy algorithm proceeds as follows. It starts from the empty set X. In every iteration Greedy adds a minimum weight element e to the current set X provided $e \notin X$ and $X \cup \{e\}$ is a subset of a set in $\mathcal{F}$. The algorithm stops when a base has been constructed.

Unfortunately, both computational experiments and domination analysis point out that Greedy is often a poor choice for heuristic even if it is only used to generate initial solutions that will be improved by more sophisticated heuristics (see the previous section). Thus, other heuristics are of definite interest. A promising and quite universal heuristic appears to be Max-Regret algorithm studied in Balas and Saltzman (1991), and Robertson (2001) for the 3-dimensional assignment problem (3-AP). Variations of Max-Regret were introduced and investigated in Ghosh et al. (2006) for ATSP. Our analysis for both both s-AP ($s \geq 3$) and ATSP indicates that Max-Regret is of similar quality in the worst case as Greedy, namely, the domination number Max-Regret for both problems equals 1. Recently, Bendall and Margot (2006) studied an extension of Greedy, which is of domination number 1 for many CO problems as well.

3 Greedy, s-AP-Max-Regret and s-AP-Max-Regret-FC

For a fixed $s \geq 2$, the s-AP is stated as follows. Let $X_1 = X_2 = \cdots = X_s = \{1, 2, \ldots, n\}$. We will consider only vectors that belong to the Cartesian product $X = X_1 \times X_2 \times \cdots \times X_s$. Each vector e is assigned a weight $w(e)$. For a vector e, e_j denotes its jth coordinate, i.e., $e_j \in X_j$. A *partial assignment* is a collection $e^1, e^2, \ldots, e^t$ of $t \leq n$ vectors such that $e^i_j \neq e^k_j$ for each $i \neq k$ and $j \in \{1, 2, \ldots, s\}$. An *assignment* is a partial assignment with n vectors. The *weight* of a partial assignment $A = \{e^1, e^2, \ldots, e^t\}$ is $w(A) = \sum_{i=1}^{t} w(e^i)$. The objective is to find an assignment of minimum weight.

We will start from Greedy for s-AP. Using Theorem 2.1 in Gutin and Yeo (2002) one can prove that, for each $s \geq 2$, $n \geq 2$, there exists an instance of s-AP for which Greedy will find the unique worst possible assignment. We will give a short direct proof of this result, which is also of interest later in this section.

A vector h is *backward* if $\min\{h_i : 2 \leq i \leq s\} < h_1$; a vector h is *horizontal* if $h_1 = h_2 = \cdots = h_s$. A vector is *forward* if it is not horizontal or backward.

Lemma 3.1 *Let F be an assignment of s-AP ($s \geq 2$). Either all vectors of F are horizontal or F contains a backward vector.*

Proof: Let $F = \{f^1, f^2, \ldots, f^n\}$, where $f^i_1 = i$ for each $1 \leq i \leq n$. Assume that not every vector of F is horizontal. We show that F has a backward vector. Suppose it is not true. Then F has a forward vector f^i. Thus, there is a subscript j such that $f^i_j > i$. By the pigeonhole principle, there exists a superscript $k > i$ such that $f^k_j \leq i$, i.e., f^k is backward; a contradiction. □

Theorem 3.2 *For each $s \geq 2$, $n \geq 2$, there exists an instance of s-AP for which* Greedy *will find the unique worst possible assignment.*

Proof: Let $M > n$ and let $E = \{e^1, e^2, \ldots, e^n\}$, where $e^i = (i, i, \ldots i)$ for every $1 \leq i \leq n$. We define the required instance $\mathcal{I}$ as follows: $w(e^i) = iM$ for each $1 \leq i \leq n$ and, for each $f \notin E$, $w(f) = \min\{f_i : 1 \leq i \leq s\} \cdot M + 1$.

Observe that Greedy will construct E. Let $F = \{f^1, f^2, \ldots, f^n\}$ be any other assignment, where $f^i_1 = i$ for each $1 \leq i \leq n$. By Lemma 3.1, F has a backward vector f^k. Notice that

$$w(f^k) \leq (k-1)M + 1 \tag{1}$$

By the definition of the weights and (1),

$$w(F) = \sum_{i=1}^{n} w(f^i) = \sum_{i \neq k} w(f^i) + w(f^k)$$

$$
\begin{aligned}
&\leq \sum_{i\neq k}(iM+1)+(k-1)M+1 \\
&= \sum_{i=1}^{n} iM+n-M \\
&< \sum_{i=1}^{n} iM = w(E).
\end{aligned}
$$

□

The first successful application of max-regret for improving the Transportation Simplex Algorithm is appeared in the so called Vogel's Approximation Method (see Reinfeld and Vogel (1958)) and has been used as a base of the Max-Regret heuristic for solving the 3-AP in Balas and Saltzman (1991). The authors of Goldengorin et el. (2006) gave a general approach that extends Max-Regret heuristics.

s-AP-Max-Regret proceeds as follows. Set $W_j = A = \emptyset$ for each $j = 1, 2, \ldots, s$. While $|X_1| \neq |W_1|$ do the following: For each $i \in \{1, 2, \ldots, s\}$ and $a \in X_i \setminus W_i$, find two lightest vectors $e^{i,a}$ and $f^{i,a}$ ($w(e^{i,a}) \leq w(f^{i,a})$) in the set

$$H = \{h \in X : \ h_i = a, h_j \in X_j \setminus W_j, j \in \{1, 2, 3, \ldots, s\} \setminus \{i\}\}$$

and compute the difference (called *regret*) $\Delta_{i,a} = w(f^{i,a}) - w(e^{i,a})$. Compute the *max-regret*

$$\Delta_{i_0,a_0} = \max\{\Delta_{i,a} : \ i \in \{1, 2, \ldots, s\}, \ a \in X_i \setminus W_i\}.$$

Add e^{i_0,a_0} to A and each $e_j^{i_0,a_0}$ to W_j, $j = 1, 2, \ldots, s$.

A modification of s-AP-Max-Regret that computes the regrets only for the first coordinates, i.e., only $\Delta_{1,a}$'s will be denoted s-AP-Max-Regret-FC (FC abbreviates First Coordinate).

Remark 3.3 *In* s-AP-Max-Regret, *when* $|H| = 1$ *we set* $\Delta_{i,a} = 0$. *Since we perform the worst case analysis, when breaking ties, we will follow the choice leading to the worst solution among possible options.*

Theorem 3.4 *The domination number of both* s-AP-Max-Regret *and* s-AP-Max-Regret-FC *equals 1 for each* $s \geq 3$.

Proof: Consider the instance $\mathcal{I}$ described in the proof of Theorem 3.2. Observe that $\Delta_{i,1} = (M+1) - M = 1$ for each i and $\Delta_{i,a} = (M+1) - (M+1) = 0$ for each $a > 1$. Thus, both s-AP-Max-Regret and s-AP-Max-Regret-FC will choose e^1 first. Similarly, we can see that both heuristics will uniquely choose $e^2, \ldots, e^n$ one by one. In Theorem 3.2, we showed that $E = \{e^1, e^2, \ldots, e^n\}$ is unique worst possible for $\mathcal{I}$. □

Notice that the proof of Theorem 3.4 cannot be extended to 2-AP-MAX-REGRET or 2-AP-MAX-REGRET-FC. Moreover, it was proved in Ghosh et al. (2006) that 2-AP-MAX-REGRET-FC is of domination number 2^{n-1}. We believe that 2^{n-1} is also the domination number for 2-AP-MAX-REGRET, but we are unable to prove it. In support of this conjecture we prove the following:

Theorem 3.5 *The domination number of* 2-AP-MAX-REGRET *is at most* 2^{n-1}.

Proof: Choose n positive numbers $d_1 > d_2 > \cdots > d_n$ arbitrarily and consider the following instance of 2-AP: $w(i,i) = -d_i$ for each $i = 1, 2, \ldots, n$, $w(i,j) = 0$ for each $1 \le i < j \le n$ and $w(i,j) = -\sum_{k=j}^{i} d_k$ for each $1 \le j < i \le n$.

Initially 2-AP-MAX-REGRET computes the regrets as follows: $\Delta_{1,k} = d_1$ and $\Delta_{2,k} = d_n$ for each $k = 1, 2, \ldots, n$. We may assume that 2-AP-MAX-REGRET chooses (1,1) (see Remark 3.3). Similarly, we can see that 2-AP-MAX-REGRET chooses $(2,2), (3,3), \ldots, (n,n)$ one by one. Thus, the weight of the assignment $M = \{(1,1), (2,2), \ldots, (n,n)\}$ built by 2-AP-MAX-REGRET equals $-\sum_{i=1}^{n} d_i$.

For an integer $p \ge 1$, let $\mathrm{Op}(i,p)$ denote an operation that replaces in M the vectors $\{(i,i), (i+1,i+1), \ldots, (i+p,i+p)\}$ by the vectors $\{(i,i+1), (i+1,i+2), \ldots, (i+p-1,i+p), (i+p,i)\}$. The operation $\mathrm{Op}(i,0)$ does nothing. Consider the following procedure. It starts from $i := 1$. It chooses an arbitrary integer p with $0 \le p \le n-i$, performs $\mathrm{Op}(i,p)$, sets $i := i+p+1$ and continues this loop while $i < n$.

Notice that $\mathrm{Op}(i,p)$ preserves the weight of the assignment and, thus, every assignment obtained by the procedure is of weight $w(M)$. Let $f(n)$ be the number of all possible assignments that can be obtained by the procedure. Clearly, $f(1) = 1$ and set $f(0) = 1$. To compute $f(n)$ observe that after using $\mathrm{Op}(1,p)$ we will have $f(n-p-1)$ possible assignments. Thus, for each $n \ge 2$ we have $f(n) = f(n-1) + f(n-2) + \ldots + f(0)$. This implies that $f(n) = 2^{n-1}$ for $n \ge 1$.

To show that any assignment that cannot be constructed by the procedure is of weight smaller than $w(M)$, build a complete digraph DK_n with vertices $\{1, 2, \ldots, n\}$ and with a loop on every vertex. For arbitrary $1 \le i, j \le n$, the arc (i,j) of DK_n corresponds to the vector (i,j) and we set the weight of arc (i,j) equal $w(i,j)$. We call an arc (i,j) with $i < j$ *forward* and with $i \ge j$ *backward*. Notice that the weight of every forward arc is 0.

An assignment corresponds to a *cycle factor* of DK_n, which is a collection of disjoint cycles (some of them may be loops) that cover all vertices of DK_n. In particular, the weight of an assignment equals the weight of the corresponding cycle factor in DK_n. Notice that the weight of every forward arc is 0 and, thus, the weight of a cycle factor equals the sum of the weights of its backward arcs. We call a pair $(i,j), (i',j')$ of backward arcs *intersecting* if the intervals $[j,i]$ and $[j',i']$ of real line intersect (one of these intervals may be just a point). Observe that if a cycle factor does not have intersecting backward arcs, then its weight equals $-\sum_{i=1}^{n} d_i = w(M)$ and every such cycle factor corresponds to an assignment that can be obtained by the procedure above. Thus, there are exactly $f(n) = 2^{n-1}$ cycle factors without intersecting backward arcs.

Now suppose that a cycle factor F has an intersecting pair $(i,j),(i',j')$ of backward arcs. Thus, there is an integer k such that $k \in [j,i] \cap [j',i']$. By the definition of a cycle factor, $k < n$. Observe that the above arguments imply that $w(F) \le -\sum_{i=1}^{n} d_i - d_k < w(M)$.

So, there are only 2^{n-1} assignments of weight not smaller than $w(M)$. $\square$

4 s-AP Heuristics of Large Domination Number

For ATSP, there are several heuristics with domination number at least $(n-2)!$; see, e.g., Punnen and Kabadi (2002). In this section, we will demonstrate that s-AP admits a number of heuristics of domination number at least $((n-1)!)^{s-1}$. We introduce two such new heuristics Part and Recursive Opt Matching, which might well be of interest in practice. The key lemma is the following result similar to the corresponding result in Gutin and Yeo (2002).

The average weight of an assignment (denoted by $\bar{w}$) is the total weight of all assignments divided by the number of assignments. The average weight of a vector in X is $w(X)/n^s$. Thus, by linearity of expectation, the average weight of an assignment equals $\bar{w} = w(X)/n^{s-1}$.

Lemma 4.1 *Let H be a heuristic that for each instance of s-AP constructs an assignment of weight at most the average weight of an assignment. Then the domination number of H is at least $((n-1)!)^{s-1}$.*

Proof: Consider an instance $\mathcal{I}$ of s-AP. Let C denote the set of all vectors of $\mathcal{I}$ with the first coordinate equal 1. Consider $\mathcal{P} = \{A_f : f^1 \in C\}$, where $A_f = \{f^1, f^2, \ldots, f^n\}$ is an assignment with $f_j^i = f_j^1 + i - 1$ (modulo n), $j = 1, 2, \ldots, s$. Observe that each vector is in exactly one A_f and, thus, $\mathcal{P}$ is a partition of $X = X_1 \times X_2 \times \cdots \times X_s$ into assignments. Since $\sum_{f \in C} w(A_f) = w(X)$, $|C| = n^{s-1}$ and $\bar{w} = w(X)/n^{s-1}$, the heaviest assignment A_h in $\mathcal{P}$ is of weight at least $\bar{w}$.

Let $S(X_i)$ be the set of all permutations on X_i $(2 \le i \le s)$ and let $\pi_2 \in S(X_2), \pi_3 \in S(X_3), \ldots, \pi_s \in S(X_s)$. To obtain $\mathcal{P}(\pi_2, \pi_3, \ldots, \pi_s)$ from $\mathcal{P}$, replace f_j^i with $\pi_j(f_j^i)$ for each $j \ge 2$ and $i = 1, 2, \ldots, n$. Thus, we obtain a family

$$\mathcal{F} = \{\mathcal{P}(\pi_2, \pi_3, \ldots, \pi_s) : \pi_2 \in S(X_2), \pi_3 \in S(X_3), \ldots, \pi_s \in S(X_s)\}$$

of partitions of X into assignments. The family consists of $(n!)^{s-1}$ partitions. We may choose the heaviest assignment in each partition and, thus, obtain a family $\mathcal{A}$ of assignments of weight at least $\bar{w}$.

However, we can have several occurrences of the same assignment in $\mathcal{A}$. We claim that no assignment $G = \{g^1, g^2, \ldots, g^n\}$ (with $g_1^i = i$ for $i = 1, 2, \ldots, n$) can be in more than n^{s-1} partitions of $\mathcal{F}$. We may assume that $G \in \mathcal{P}$. Let G be also in some $\mathcal{P}(\pi_2, \pi_3, \ldots, \pi_s)$. By definition, there is an assignment $\{d^1, d^2, \ldots, d^n\}$ in $\mathcal{P}$ with $d_1^i = i$ for $i = 1, 2, \ldots, n$ such that $g_j^i = \pi_j(d_j^i)$ for each $j = 2, 3, \ldots, s$ and $i = 1, 2, \ldots, n$. These relations uniquely define the permutations $\pi_2, \pi_3, \ldots, \pi_s$. Thus, $\{g^1, g^2, \ldots, g^n\}$ can be repeated in $\mathcal{F}$ at most $|\mathcal{P}| = n^{s-1}$ times.

So, each assignment in $\mathcal{A}$ is of weight at least $\bar{w}$, no assignment in $\mathcal{A}$ can be repeated more than n^{s-1} times, and $\mathcal{A}$ has $(n!)^{s-1}$ assignments with repetitions. Therefore, we can find (in $\mathcal{A}$) $((n-1)!)^{s-1}$ distinct assignments of weight at least $\bar{w}$. Since $w(H(\mathcal{I})) \leq \bar{w}$ and $\mathcal{I}$ is arbitrary, we conclude that H is of domination number at least $(n!)^{s-1}$. $\square$

Consider a new heuristic Part that finds a partition $\mathcal{P}$ of X into assignments and computes an assignment in $\mathcal{P}$ of minimum weight. The proof above shows that Part is of domination number at least $((n-1)!)^{s-1}$. This heuristic is fast (of time complexity $O(n^s)$) and might be of interest at least for producing initial assignments for local improvement heuristics such as the Triple Interchange introduced in Balas and Saltzman (1991) for 3-AP. Before studying Triple Interchange we consider another new heuristic Recursive Opt Matching for s-AP.

Recursive Opt Matching proceeds as follows. Compute a new weight $\bar{w}(i,j) = w(X_{ij})/n^{s-2}$, where X_{ij} is the set of all vectors with last two coordinates equal i and j, respectively. Solving the 2-AP with the new weights to optimality, find an optimal assignment $\{(i, \pi_s(i)) : i = 1, 2, \ldots, n\}$, where π_s is a permutation on X_s. While $s \neq 1$, introduce $(s-1)$-AP with weights given as follows: $w'(f^i) = w(f^i, \pi_s(i))$ for each vector $f^i \in X'$, where $X' = X_1 \times X_2 \times \cdots \times X_{s-1}$, with last coordinate equal i and apply Recursive Opt Matching recursively. As a result we have obtained permutations $\pi_s, \pi_{s-1}, \ldots, \pi_2$. The output is the assignment $\{(i, \pi_2(i), \pi_3(\pi_2(i)), \ldots, \pi_s(\pi_{s-1}(\ldots(\pi_2(i)))\ldots)) : i = 1, 2, \ldots, n\}$.

Theorem 4.2 *For each $s \geq 2$,* Recursive Opt Matching *is of domination number at least* $((n-1)!)^{s-1}$.

Proof: By Lemma 4.1, it suffices to show that the assignment obtained by Recursive Opt Matching is of weight at most $\bar{w} = w(X)/n^{s-1}$, the average weight of an assignment. Our proof is by induction on $s \geq 2$. Clearly the assertion holds for $s = 2$ and consider $s \geq 3$. Observe that

$$\frac{w(X)}{n^{s-1}} = \bar{w} = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \bar{w}(i,j) \geq \sum_{i=1}^{n} \bar{w}(i, \pi_s(i)) = \frac{w'(X')}{n^{s-2}}.$$

Let $A = \{(g^1, \pi_s(1)), \ldots, (g^n, \pi_s(n))\}$ be an assignment obtained by Recursive Opt Matching, where $g^i \in X'$ such that $g^i_{s-1} = i$ for every $i = 1, \ldots, n$. Let $A' = \{g^1, \ldots, g^n\}$. Then by induction hypothesis, $\bar{w}' = w'(X')/n^{s-2} \geq w'(A') = w(A)$ and we are done. □

It is straightforward to see that for any fixed $s \geq 3$, Recursive Opt Matching is of running time merely $O(n^s)$.

Consider 3-AP. Triple Interchange is a local search heuristic that at every step tries to improve an assignment $D = \{d^1, d^2, \ldots, d^n\}$ by looking at a triple of vectors d^i, d^j, d^k. It compares $w(d^i)+w(d^j)+w(d^k)$ with the weight of each of 35 triples $(h^i, h^j, h^k) \neq (d^i, d^j, d^k)$ such that $h^i_1 = d^i_1, h^j_1 = d^j_1, h^k_1 = d^k_1$, $\{h^i_2, h^j_2, h^k_2\} = \{d^i_2, d^j_2, d^k_2\}$ and $\{h^i_3, h^j_3, h^k_3\} = \{d^i_3, d^j_3, d^k_3\}$. If Triple Interchange finds a triple h^i, h^j, h^k lighter than d^i, d^j, d^k, it replaces d^i, d^j, d^k with h^i, h^j, h^k in D. The heuristic stops when no triple in the current assignment D can be replaced by a lighter one.

The following theorem does not depend on the initial assignment in Triple Interchange.

Theorem 4.3 *The domination number of* Triple Interchange *is at least* $((n-1)!)^2$.

Proof: Assume that $E = \{e^1, e^2, \ldots, e^n\}$, where $e^i = (i, i, i)$, is an assignment that cannot be improved using Triple Interchange. The set of all vectors $X = Y \cup Z \cup E$, where Y is the set of vectors with exactly two equal coordinates and Z is the set of vectors with all coordinates being different. Clearly, $w(X) = w(Y) + w(Z) + w(E)$. We will prove that $w(Y) \geq 3(n-1)w(E)$ and $w(Z) \geq (n-1)(n-2)w(E)$, which imply that $w(E) \leq \bar{w} = w(X)/n^2$ and the result of the theorem follows from Lemma 4.1.

Observe that $|Y| = 3n(n-1)$ (there are 3 ways to choose which coordinate is different from the other two, n ways to choose value from $\{1, 2, \ldots, n\}$ for this coordinate and $n-1$ ways to choose value for the two coordinates). The set Y can be partitioned into $|Y|/2$ pairs of the form f^i, f^j such that f^i has one coordinate equal i and two coordinates equal j and f^j has one coordinate equal j and two coordinates equal i. For each such pair f^i, f^j, we have $w(f^i)+w(f^j) \geq w(e^i)+w(e^j)$ as otherwise we could improve e^i, e^j, e^k by f^i, f^j, e^k $(k \neq i, j)$. Summing up all the inequalities we obtain $w(Y) \geq 3(n-1)w(E)$.

Note that $|Z| = n(n-1)(n-2)$. For a vector $f = (i, j, k)$, let $f^+ = (k, i, j)$ and $f^- = (j, k, i)$. Let $F = \{(i, j, k) \in X : \ i < j < k\}$ and $G = \{(i, j, k) \in X : \ j < i < k\}$. Then $Z = \{\{f, f^+, f^-\} : \ f \in F \cup G\}$ is a partition of Z into $|Z|/3$ triples. Observe that for a triple $h = (i, j, k), h^+, h^-$, we have $w(h) + w(h^+) + w(h^-) \geq w(e^i) + w(e^j) + w(e^k)$. This implies $w(Z) \geq (n-1)(n-2)w(E)$. □

The Pair Interchange heuristic also described in Balas and Saltzman (1991) is similar to Triple Interchange, but tries to improve pairs of vectors in the current assignment. Pair Interchange does not always produce an assignment whose weight is at most the average weight of an assignment. To see that consider an instance of 3-AP with the following weights: $w(i,i,i) = 0$ for each $i = 1, 2, \ldots, n$, $w(i,j,k) = 1$ for each triple i, j, k in which exactly two members equal, and $w(i,j,k) = -n^3$ for each triple i, j, k in which all members of different. The assignment $E = \{(1,1,1), (2,2,2), \ldots, (n,n,n)\}$ cannot be improved by Pair Interchange, but $w(E) = 0$ and the average weight of an assignment is negative for each $n \geq 3$.

5 ATSP–Max-Regret and ATSP-Max-Regret-FC

A variation of Max-Regret for ATSP, ATSP-Max-Regret-FC (FC abbreviates First Coordinate), was first introduced in (Ghosh et al. 2006) under a different name, R-R-Greedy. The authors of (Ghosh et al. 2006) found an exponential upper bound on the domination number of ATSP-Max-Regret-FC and stated a problem to obtain a nontrivial lower bound for the domination number. Extensive computational experiments in (Ghosh et al. 2006) demonstrated a clear superiority of ATSP-Max-Regret-FC over Greedy and several other construction heuristics in Glover et al. (2001). Therefore, the result of Theorem 5.2 is somewhat unexpected.

Let K_n^* be a complete digraph with vertices $V = \{1, 2, \ldots, n\}$. The weight of an arc (i,j) is denoted by w_{ij}. The *ATSP* is the problem of finding a tour (i.e., a Hamilton cycle) of K_n^* of total minimum weight. Let Q be a collection of disjoint paths in K_n^*. An arc $a = (i,j)$ is a *feasible addition* to Q if $Q + a$ is either a collection of disjoint paths or a tour in K_n^*. Consider ATSP-Max-Regret-FC and ATSP-Max-Regret.

ATSP-Max-Regret-FC proceeds as follows. Set $W = T = \emptyset$. While $V \neq W$ do the following: For each $i \in V \setminus W$, compute two lightest arcs (i,j) and (i,k) that are feasible additions to T, and compute the difference $\Delta_i = |w_{ij} - w_{ik}|$. For $i \in V - W$ with maximum Δ_i choose the lightest arc (i,j), which is a feasible addition to T and add (i,j) to M and i to W.

ATSP-Max-Regret proceeds as follows. Set $W^+ = W^- = T = \emptyset$. While $V \neq W^+$ do the following: For each $i \in V \setminus W^+$, compute two lightest arcs (i,j) and (i,k) that are feasible additions to T, and compute the difference $\Delta_i^+ = |w_{ij} - w_{ik}|$; for each $i \in V \setminus W^-$, compute two lightest arcs (j,i) and (k,i) that are feasible additions to T, and compute the difference $\Delta_i^- = |w_{ji} - w_{ki}|$. Compute $i' \in V \setminus W^+$ with maximum $\Delta_{i'}^+$ and $i'' \in V \setminus W^-$ with maximum $\Delta_{i''}^-$. If $\Delta_{i'}^+ \geq \Delta_{i''}^-$ choose the lightest arc (i', j'), which is a feasible addition to T and add (i', j') to M, i' to W^+ and j' to W^-. Otherwise, choose the lightest arc (j'', i''), which is a feasible addition to T and add (j'', i'') to M, i'' to W^- and j'' to W^+.

Remark 5.1 *In* ATSP-Max-Regret-FC*, if* $|V \setminus W| = 1$ *we set* $\Delta_i = 0$*. A similar remark applies to* ATSP-Max-Regret*.*

Theorem 5.2 *The domination number of both* ATSP-Max-Regret-FC *and* ATSP-Max-Regret *equals* 1 *for each* $n \geq 2$.

Proof: Since the proofs for both heuristics use the same family of instances and are similar, we restrict ourselves only to ATSP-Max-Regret-FC.

Consider an instance of ATSP on the complete digraph with vertex set $\{1, 2, \ldots, n\}$, $n \geq 2$. Let the weights be as follows: $w_{ik} = \min\{0, i - k\}$ for each $1 \leq i \neq k \leq n$, $i \neq n$, and $w_{nk} = -k$ for each $1 \leq k \leq n - 1$. We will slightly modify the weights: $w'_{ij} = w_{ij}$ unless $j = i+1$ modulo n. We set $w'_{i,i+1} = -1 - \frac{1}{n+1}$ for $1 \leq i \leq n-1$ and $w'_{n,1} = -1 - \frac{1}{n+1}$. ATSP-Max-Regret-FC will use the weight function w'.

ATSP-Max-Regret-FC constructs the tour $T_{MR} = (1, 2, 3, \ldots, n, 1)$ by first choosing the arc $(n-1, n)$, then the arc $(n-2, n-1)$, etc. The last two arcs are $(1, 2)$ and $(n, 1)$ (they must be included in the tour). Indeed, initially $\Delta_{n-1} = \frac{n+2}{n+1} > \Delta_i$ for each $i \neq n-1$. Once $(n-1, n)$ is added to T_{MR}, $\Delta_{n-2} = \frac{n+2}{n+1}$ becomes maximal, etc.

Let T', T'' be a pair of tours. Since $\sum_{(i,j)\in K_n^*} |w_{ij} - w'_{ij}| < 1$, $w(T') < w(T'')$ implies $w'(T') < w'(T'')$. Thus, to prove that $w'(T) < w'(T_{MR})$ for each tour $T \neq T_{MR}$, it suffices to show that $w(T) < w(T_{MR})$.

Observe that $w(T_{MR}) = -n$. Let $T = (i_1, i_2, \ldots, i_n, i_1)$ be an arbitrary tour, where $i_1 = 1$. Suppose that $i_s = n$. Observe that the weight of the path $P = (i_1, i_2, \ldots, i_s)$ equals $\sum_{k=1}^{s-1} \min\{0, i_k - i_{k+1}\}$. Thus, $w(P) \leq 1 - n$ and $w(P) = 1 - n$ if and only if $i_1 < i_2 < \cdots < i_s$. Since $i_s = n$, the weight of the arc (i_s, i_{s+1}) equals $-i_{s+1}$. Thus, $w(T) \leq 1 - n - i_{s+1}$ and $w(T) \geq w(T_{MR})$ if and only if $i_{s+1} = 1$ and $i_1 < i_2 < \cdots < i_s$. We conclude that $w(T) \geq w(T_{MR})$ if and only if $T = T_{MR}$. $\square$

6 Conclusions and Further Research

We have carried out worst-case analysis of Max-Regret for the Multidimensional Assignment Problem (s-AP, $s \geq 3$) and Asymmetric Traveling Salesman Problem (ATSP). We proved that Max-Regret for both problems may find unique worst possible solution. Thus, like Greedy, Max-Regret should be used with great care and, possibly, avoided all together when instances of previously unstudied families are to be solved. In such a case heuristics of factorial domination number that have a proven excellent computational record (such as Helsgaun's version of Lin-Kernighan heuristic for the Symmetric TSP (see Helsgaun (2000), and Punnen et al. (2003)) appear to be a much better choice.

For s-AP we considered three heuristics of factorial domination number. Two of the heuristics are new and, we believe, that they might well be of practical interest. Gerold Jäger has already performed preliminary computational experiments comparing Recursive Opt Matching and its modifications with other fast heuristics including Greedy and Max-Regret for s-AP, $s \geq 3$. The experiments demonstrated that on average Recursive Opt Matching and its variants outperform the other tested heuristics especially for $s > 3$. We plan to report on these and other experimental results in a future paper.

Acknowledgements We are thankful to the anonymous referees for useful comments and suggestions. Most of this paper was written when the first author was visiting Department of Mathematics and Statistics, University of Victoria, Canada. He would like to thank the department for its hospitality. His research was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. The second author acknowledges support by the DFG project SI 657/5.

References

Ausiello, G., P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi. (1999). Complexity and Approximation. Berlin: Springer-Verlag.

Balas, E., M.J. Saltzman. (1991). "An algorithm for the three-index assignment problem," Oper. Res. 39, 150–161.

Bendall, G., F. Margot. (2006). "Greedy Type Resistance of Combinatorial Problems," Discrete Optimization 3, 288–298.

Berend, D., S. Skiena, Y. Twitto. (2006). "Combinatorial Dominance Guarantees for Problems with Infeasible Solutions." Submitted.

Burkard R., E. Cela. (1999). "Linear assignment problems and extensions." In Z. Du and P. Pardalos (eds.), Handbook of Combinatorial Optimization. Dordrecht: Kluwer Acad. Publishers, 75–149.

Ghosh, D., B. Goldengorin, G. Gutin and G. Jäger. (2007). "Tolerance based greedy algorithms for the traveling salesman problem." To appear in *Communic. in DQM.*

Glover, F., G. Gutin, A. Yeo, A. Zverovich. (2001). "Construction heuristics for the asymmetric TSP," Europ. J. Oper. Res. 129, 555–568.

Glover, F., A. Punnen. (1997). "The traveling salesman problem: New solvable cases and linkages with the development of approximation algorithms," J. Oper. Res. Soc. 48, 502–510.

Goldengorin, B., G. Jäger and P. Molitor. (2006). "Tolerances Applied in Combinatorial Optimization." Journal of Computer Science 2(9), 716–734.

Gupta P., A. B. Kahng, S. Mantik. (2005). "Routing-Aware Scan Chain Ordering," ACM Transactions on Design Automation of Electronic Systems 10(3), 546–560.

Gutin, G., A. Punnen (eds.) (2002). The Traveling Salesman Problem and its Variations, Dordrecht: Kluwer Acad. Publishers.

Gutin, G., A. Yeo. (2005). "Domination Analysis of Combinatorial Optimization Algorithms and Problems." In M. Golumbic, and I. Hartman, (eds.) Graph Theory, Combinatorics and Algorithms: Interdisciplinary Applications, Berlin: Springer-Verlag.

Gutin, G., A. Yeo. (2002). "Polynomial approximation algorithms for the TSP and the QAP with a factorial domination number," Discrete Appl. Math. 119, 107–116.

Helsgaun, K. (2000). "An effective implementation of the Lin-Kernighan traveling salesman heuristic," Europ. J. Oper. Res. 126, 106–130.

Johnson, D.S., G. Gutin, L. McGeoch, A. Yeo, X. Zhang, A. Zverovitch. (2002). "Experimental Analysis of Heuristics for ATSP." In G. Gutin, and A. Punnen (eds.), The Traveling Salesman Problem and its Variations, Dordrecht: Kluwer Acad. Publishers.

Koller, A.E., S.D. Noble. (2004). "Domination analysis of greedy heuristics for the frequency assignment problem," Discrete Math. 275, 331–338.

Murphey, R., P.M. Pardalos, L.S. Pitsoulis. (1998). "A Parallel GRASP for the Data Association Multidimensional Assignment Problem." In Parallel Processing of Discrete Problems, The IMA Volumes in Mathematics and its Applications, 106, 159–180.

Pierskalla, W.P. (1968). "The multidimensional assignment problem." Oper.Res. 16, 422–431.

Poore, A.B.(1994). "Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking." Comput. Optim.Appl. 3, 27–54.

Punnen, A.P. (2002). "The Traveling Salesman Problem: Applications, Formulations and Variations." In G. Gutin, and A. Punnen (eds.), The Traveling Salesman Problem and its Variations, Dordrecht: Kluwer Acad. Publishers.

Punnen, A.P., F. Margot, S.N. Kabadi. (2003). "TSP heuristics: domination analysis and complexity," Algorithmica 35, 111–127.

Punnen, A.P., S. Kabadi. (2002). "Domination analysis of some heuristics for the traveling salesman problem," Discrete Appl. Math. 119, 117–128.

Pusztaszeri, J., P.E. Rensing, T.M. Liebling. (1995). "Tracking elementary particles near their primary vertex: a combinatorial approach." J. Global Optim. 16, 422–431.

Reinfeld, N.V. and W.R. Vogel, 1958. Mathematical Programming. Prentice-Hall, Englewood Cliffs, N.J.

Robertson, A.J. (2001). "A set of greedy randomized adaptive local search procedure implementations for the multidimentional assignment problem," Computational Optimization and Applications 19, 145–164.

Xu, L., A.C. Tan, D.Q. Naiman, D. Geman and R.L. Winslow. (2005). "Robust prostate cancer marker genes emerge from direct integration of inter-study microarray data," Bioinformatics 21, 3905–3911.