On the Efficiency of Worst Improvement for Climbing NK-Landscapes

Matthieu Basseur
LERIA, Université d'Angers, France
matthieu.basseur@univ-angers.fr

Adrien Goëffon
LERIA, Université d'Angers, France
adrien.goeffon@univ-angers.fr

ABSTRACT

Climbers are often used in metaheuristics in order to intensify the search and identify local optima with respect to a neighborhood structure. Even if they constitute a central component of modern heuristics, their design principally consists in choosing the pivoting rule, which is often reduced to two alternative strategies: first improvement or best improvement. The conception effort of most metaheuristics belongs in proposing techniques to escape from local optima, and not necessarily on how to climb toward better local optima. In this paper, we are interested in attaining good local optima with basic hill-climbing techniques. The NK model will be used to evaluate a set of climbers proposed in this paper. By focusing on the pivoting rule definition, we show that choosing the worst improving neighbor often leads to attain better local optima. Moreover, by slightly modifying the worst improvement strategy, one can design efficient climbers which outperform first and best improvement in terms of tradeoff between quality and computational effort.

Categories and Subject Descriptors

I.2.8 [**Computing Methodologies**]: Artificial Intelligence—*Problem Solving, Control Methods, and Search*

Keywords

Local search, hill-climbing, fitness landscapes, NK-landscapes, first improvement, best improvement, worst improvement.

1. INTRODUCTION

Most metaheuristics are based on a neighborhood search technique, which consists in transforming an initial solution by application of local moves chosen from a neighborhood structure. During the last decades, a large panel of neighborhood searches have been designed, like hill-climbing, tabu search, simulated annealing, or iterated local search (see [3]). These metaheuristics mainly differ in the move selection heuristic employed. Hill-climbings are simple neighborhood search techniques and are widely used as basic elements of more sophisticated local search algorithms or advanced metaheuristics (including evolutionary and hybrid algorithms).

While designing a neighborhood search algorithm, three main conceptual issues have to be considered. First, the neighborhood structure design, which directly affects the fitness landscape to explore. This aspect is greatly problem-dependant and will not be discussed in this paper. Second, a method to choose the solution initiating or restarting the search has to be designed. This aspect is often debated, especially in iterated local search studies. Last, the *pivoting rule* defines the acceptance criterion of neighbors during the search. The paper focuses on this major aspect, which is rarely investigated, except for comparing first and best improvements on specific problems [2, 10].

In a previous study [1], we showed that first improvement tends to outperform best improvement on many landscapes derived from several combinatorial problems. An analysis of landscapes properties also indicated that first improvement is particularly efficient for climbing non-smooth landscapes. Intuitively, since best improvement systematically chooses the highest neighbor, it should conduce the search towards the nearest peak (local optima) of the landscape. On the contrary, first improvement often performs reduced improvements, which tends to drive progressively the search toward higher areas, where the potential local optima are also higher. To extrapolate, we believe that choosing the worst improving neighbor at each step of the search will allow to increase the number of steps and then the possibility to reach higher areas of landscapes by avoiding the climbing of small steepest peaks. In this paper, we aim at evaluating worst improvement as well as some other intermediate pivoting rules.

The paper is organized as follow. The next section focuses on fitness landscapes and NK model, and also recall hill-climbing principles. In section 3, we present alternative pivoting rules, including the underestimated worst improvement strategy. Results of empirical comparisons between 12 pivoting rules will be provided and analyzed in this section. In section 4, we propose refined pivoting rules, which allow to approximate the worst improvement strategy, with a reduced number of solution evaluations. Experiments and result analysis are also provided in order to evaluate the effectiveness of such approaches. In the last section we point out some conclusion and discuss on perspectives of this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GECCO'14, July 12–16, 2014, Vancouver, BC, Canada.

Copyright 2014 ACM 978-1-4503-2662-9/14/07 ...$15.00.

http://dx.doi.org/10.1145/2576768.2598268.

2. CLIMBING FITNESS LANDSCAPES

2.1 Combinatorial fitness landscapes

Fitness landscapes constitute a key concept in theoretical / evolutionary biology since the beginning of the 1930s. Originally, a fitness landscape is a genotype-phenotype mapping [11] which emphasizes the effect of mutations (genotype modifications) on fitness (lifetime reproductive success). Nowadays, this concept is frequently used to characterize combinatorial optimization problems within an evolutionary computation context [7, 6, 9].

Formally, a *combinatorial fitness landscape* is a triplet $(\mathcal{X}, \mathcal{N}, \mathfrak{f})$, where $\mathcal{X}$ is a set of *configurations* (or *candidate solutions*) called *search space*, $\mathcal{N} : \mathcal{X} \to 2^{\mathcal{X}}$ is a *neighborhood relation*, and $\mathfrak{f}$ is a *fitness function*. $\mathfrak{f}(x)$ is the *fitness* (or *height*) of a configuration $x \in \mathcal{X}$. $x' \in \mathcal{N}(x)$ is a *neighbor* of x. $\mathcal{N}(x)$ is called the *neighborhood* of x. In a maximization context, a *local optimum* is a configuration $x \in \mathcal{X}$ such that $\forall x' \in \mathcal{N}(x), \mathfrak{f}(x') \leqslant \mathfrak{f}(x)$, while a *global optimum* is a configuration $x^\star \in \operatorname{argmax}_{x\in\mathcal{X}} \mathfrak{f}(x)$.

A fitness landscape $(\mathcal{X}, \mathcal{N}, \mathfrak{f})$ is usually linked with a combinatorial optimization problem instance, where $\mathcal{X}$ represents a discrete set of feasible solutions, and $\mathfrak{f}$ corresponds to a scalar objective function or its estimation.

2.2 NK-landscapes

The NK family of landscapes has been proposed in [4] in order to generate artificial combinatorial landscapes with tunable shape properties: size and ruggedness. NK-landscapes use a basic search space, with binary strings as configurations and bit-flip (1-flip) as neighborhood. Characteristics of an NK-landscape are determined by means of two parameters N and K. N refers to the length of binary string configurations. $K \in \{0, \ldots, N-1\}$ specifies the level of variables interdependency, which directly affects the ruggedness of the landscape. By increasing the value of K from 0 to $N-1$, NK-landscapes can be tuned from smooth to rugged. In particular, if $K = 0$, the landscape contains only one local (global) optimum; on the contrary, setting K to $N-1$ leads to a random fitness assignment.

In NK-landscapes, the fitness function $F_{\mathrm{NK}} : \{0,1\}^N \to [0,1)$ to be maximized is defined as follows:

$$F_{\mathrm{NK}}(x) = \frac{1}{N}\sum_{i=1}^{N} C_i(x_i, \Pi_i(x)) \qquad (1)$$

x_i is the i-th bit of configuration x $(1 \leqslant i \leqslant N)$. Subfunction Π_i defines the dependencies of bit i, with $\Pi_i(x) = \{x_{\pi_1(i)}, \ldots, x_{\pi_K(i)}\}$ such that $\pi_j(i) \in \{1, \ldots, N\} \setminus \{i\}$ and $|\bigcup_{j=1}^{K} \pi_j(i)| = K$. Subfunction $C_i : \{0,1\}^{K+1} \to [0,1)$ defines the contribution value of x_i with respect to its set of dependencies $\Pi_i(x)$.

NK-landscape instances are both determined by the $(K+1)$-uples $(x_i, x_{\pi_1(i)}, \ldots, x_{\pi_K(i)})$ and a fitness contribution matrix C describing the $2^N \times (K+1)$ possible contribution values. In standard NK-landscape instances considered here, dependencies between variables and contribution values are randomly generated from a uniform distribution.

2.3 Hill-climbing and Move Strategies

A *hill-climbing* algorithm (or *climber*) is a basic local search strategy which navigates through the search space thanks to non-deteriorating moves only. Given an initial configuration called *starting point*, a traditional climber iteratively moves to better neighbors, until a local optimum is reached. Such a search mechanism, also known as *iterative improvement*, allows to distinguish several variants which are discussed hereafter. To summarize, climbing a fitness landscape $(\mathcal{X}, \mathcal{N}, \mathfrak{f})$ consists actually in defining a move strategy which reaches a configuration as highest as possible.

Two commonly-used climbing move strategies, *first* and *best* improvement, constitute the widely used *pivoting rules*. These rules define how to select a better neighbor from a not locally optimal configuration [12]. More precisely, the *best improvement* strategy (or *greedy hill-climbing*) consists in selecting, at each iteration, a neighbor which achieves the best fitness. This implies to generate and evaluate the whole neighborhood at each step of the search. The *first improvement* strategy accepts the first evaluated neighbor which satisfies the moving condition and avoids the systematic generation of the entire neighborhood. *Worst improvement*, among other pivoting rules, can easily be considered but, to the best of our knowledge, is not really envisaged for the design of local search algorithms. This paper will focus on determining the actual efficiency of these underestimated move strategies, which are detailed in the next section.

A neutral move policy has to be established for climbing landscapes containing neutrality. We focus here on the evaluation of pivoting rules, then experimental comparisons will be realized on non-neutral NK-landscapes in order to prevent a possible impact of neutral move policies. Thus, climbers considered afterwards will respect the generic basic iterative improvement definition, allowing only stricly improving moves (see algorithm 1).

Algorithm 1 Generic basic iterative improvement algorithm

```
input: a starting point x_s ∈ X
x ← x_s
while x is not a local optimum do
    x ← x', x' choosen in {x+ ∈ N(x), f(x+) > f(x)} w.r.t.
    a pivoting rule
return x
```

3. FROM BEST TO WORST IMPROVEMENT

In a precedent study [1], we empirically show that first improvement allows to find better local optima than best improvement on most combinatorial landscapes, especially on difficult ones (fairly large and/or rugged landscapes). This result was observed on NK-landscapes and fitness landscapes derived from academic combinatorial problems: MAXSAT, QAP, and Flow-shop. In other words, this means that on difficult landscapes, selecting systematically the highest neighbor statistically leads to be trapped in local optimal of lower quality. It should be interesting to determine if strategies which favor the selection of improving neighbors of lower fitnesses conduct to higher local optima. In particular, it makes sense to consider the worst improvement pivoting rule, which selects the worst improving neighbor at each step of the search.

3.1 Considering Rank-based Pivoting Rules

Let $I_x = (x_1, x_2, \ldots, x_w)$ a tuple constitued by all strictly (distincts) improving neighbors of $x \in \mathcal{X}$ sorted by decreasing fitnesses, ie. $\forall i \in \{1, \ldots, w\}, x_i \in \mathcal{N}(x)$, $\mathfrak{f}(x_i) > \mathfrak{f}(x)$ and $\mathfrak{f}(x_i) \geqslant \mathfrak{f}(x_{i+1})$ $(i < w)$.

Given a configuration x and an associated tuple $I_x = (x_1, x_2, \ldots, x_w)$, let us describe three generic ranking-based pivoting rules:

- $S_i^{=}$ selects at each step of the search the i-th best improving neighbor (x_i).

- $S_i^{\geqslant}$ selects randomly one of the i-th best improving neighbors (belonging to $\{x_1, x_2, \ldots, x_i\}$).

- $S_i^{\leqslant}$ selects randomly one of the $(w - i + 1)$-th worst improving neighbors (belonging to $\{x_i, x_{i+1}, \ldots, x_w\}$).

Let $q_1 = \lceil w/4 \rceil$, $m = \lceil w/2 \rceil$ and $q_3 = \lceil 3w/4 \rceil$ be respectively the indices of the first, second and third quartiles of I_x. $b = 1$ and w denote the indices corresponding to the best and the worst configurations of I_x. In the following, we will consider $S_i^{=}$, $S_i^{\geqslant}$ and $S_i^{\leqslant}$ combined with 5 ranking values: $i \in \{b, q_1, m, q_3, w\}$. Note that, as depicted in figure 1, some strategies are equivalent ($S_b^{=}$ and $S_b^{\geqslant}$, $S_b^{\leqslant}$ and $S_w^{\geqslant}$, $S_w^{=}$ and $S_w^{\leqslant}$). Obviously, $S_b^{=}$ corresponds to best improvement, $S_w^{=}$ to worst improvement, and $S_w^{\geqslant}$ to first improvement — despite the fact that the definition assumes that the whole neighborhood is systematically generated.

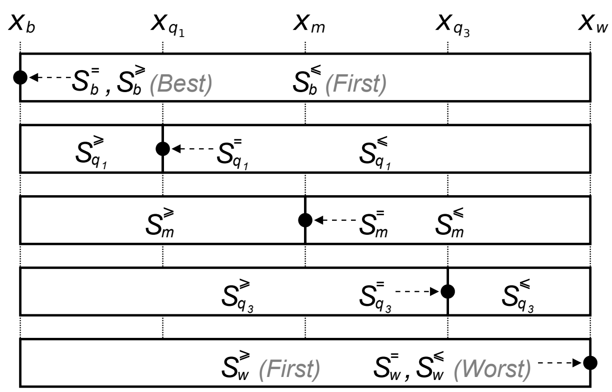

Figure 1: Illustration of different ranking-based pivoting rules. Improving neighbors are sorted from the best (x_b) to the worst (x_w). This ranking determines the candidate set of acceptable neighbors according to the pivoting rule. For instance, $S_{q_3}^{\geqslant}$ consists in randomly selecting a configuration among the 75% most improving neighbors.

Consequently, we extracted 12 distinct ranking-based pivoting rules, which includes first, best and worst improvement move strategies. Their relative efficiency will be discussed below.

3.2 Experimentations

This section aims at competing climbers which use different ranking-based pivoting rules by measuring their respective ability to reach high local optima. Let us notice that this study mainly focus on the worst improvement performance. Other variants will help us to evaluate the impact of the move strategy in local searches.

Experimental process consists in the comparison of 12 hill-climbing versions, each one corresponding to a specific rank-based pivoting rule S (see section 3.1): $S_b^{=}$, $S_{q_1}^{=}$, $S_m^{=}$, $S_{q_3}^{=}$, $S_w^{=}$, $S_{q_1}^{\geqslant}$, $S_m^{\geqslant}$, $S_{q_3}^{\geqslant}$, $S_w^{\geqslant}$, $S_{q_1}^{\leqslant}$, $S_m^{\leqslant}$, $S_{q_3}^{\leqslant}$. All climbers are tested on various NK-landscapes. Recall that N and K parameters allow to build NK instances of different sizes and ruggednesses. In our experiments, we consider 28 (N, K) parametrizations, corresponding to all combinations of $N \in \{128, 256, 512, 1024\}$ and $K \in \{1, 2, 4, 6, 8, 10, 12\}$. For each couple (N, K), we generated 10 random landscapes. Consequently, 280 different NK-landscapes are considered here.

100 random configurations were generated for each NK-landscape, which will be used as starting points for hill-climbings. For each NK-landscape, 100 executions of the 12 hill-climbing versions were performed, using the same set of starting points in order to cancel the initialization bias. Searches are stopped when a local optimum is reached.

In this experiment analysis, we mainly focus on the quality of the local optima reached by all hill-climbings variants. For each triplet (N, K, S), we first compute the average fitness of the 1,000 local optima resulting from the corresponding searches. To increase readability, we only report on table 1 results for $S_i^{=}$ climbers as well as $S_w^{\geqslant}$ (first improvement) for comparison. Other intermediate variants are not outputted here, but will be discussed later. For each NK parametrization, we report the best average value in bold. The background is colored in grey if a method is not statistically outperformed by any other variant with respect to a binomial test (p-value 0.95) comparing, for a given pair of methods, both local optima obtained for each couple (instance,starting point). More precisely, if $\mathcal{S}$ denotes the number of successes of method A over method B, then A statistically outperforms B when $\frac{1}{2^{1000}} \sum_{i=0}^{\mathcal{S}} \binom{1000}{\mathcal{S}} \geqslant 0.95$, ie. $\mathcal{S} \geqslant 526$.

Numerical results given in table 1 allows to extract several clear conclusions. First, we observe that best improvement systematically achieves best average results on smooth instances ($K = \{1, 2\}$), while worst improvement outperforms other variants on all other landscapes. This observation is clearly confirmed by the statistical analysis. Previous work [1] already shown the superiority of best improvement against first improvement on smooth NK-landscapes, and the superiority of first improvement against best improvement on rugged ones. Here, experiments on rank-based pivoting rules show that worst improvement is clearly more efficient that first improvement each time first improvement outperforms best improvement. For reaching better local optima, it seems to be appropriate to choose only between best and worst improvement strategies, depending of the ruggedness of the landscape. This tendency is confirmed in figure 2, which shows the evolution of average fitness values according to the rank-based pivoting rule used. It indicates that the average quality of local optima is strongly correlated with the rank (quality) of selected neighbors, positively on smooth NK-landscapes and negatively on rugged ones.

Table 2 focus on the best local optima obtained by each strategy on each landscape. For each couple (strategy, instance), we first compute the best local optimum reached among the 100 executions. Given a (N,K) parametrization, values provided in the table corresponds to the average fitness of the best local optima obtained by each strategy on the 10 instances. Globally, observations reported for table 1 remain valid. We can then expect that the superiority of

N_K	$S_b^=$	$S_{q1}^=$	$S_m^=$	$S_{q3}^=$	$S_w^=$	$S_w^{\geqslant}$
128_1	**.6973**	.6916	.6877	.6845	.6809	.6896
256_1	**.7033**	.6986	.6966	.6944	.6918	.6980
512_1	**.6937**	.6886	.6856	.6828	.6803	.6872
1024_1	**.6997**	.6954	.6920	.6895	.6867	.6936
128_2	**.7169**	.7142	.7121	.7107	.7081	.7133
256_2	**.7130**	.7104	.7080	.7055	.7035	.7080
512_2	**.7120**	.7097	.7077	.7063	.7041	.7076
1024_2	**.7132**	.7109	.7092	.7073	.7052	.7088
128_4	.7216	.7214	.7231	.7230	**.7250**	.7207
256_4	.7211	.7224	.7237	.7247	**.7274**	.7218
512_4	.7228	.7238	.7255	.7268	**.7285**	.7232
1024_4	.7232	.7244	.7262	.7275	**.7298**	.7238
128_6	.7170	.7178	.7194	.7217	**.7251**	.7187
256_6	.7207	.7217	.7257	.7269	**.7313**	.7234
512_6	.7214	.7232	.7261	.7291	**.7338**	.7239
1024_6	.7223	.7243	.7276	.7302	**.7353**	.7250
128_8	.7124	.7135	.7150	.7166	**.7199**	.7136
256_8	.7147	.7163	.7201	.7214	**.7267**	.7179
512_8	.7163	.7190	.7220	.7256	**.7306**	.7200
1024_8	.7176	.7206	.7239	.7274	**.7330**	.7215
128_10	.7049	.7056	.7083	.7106	**.7115**	.7078
256_10	.7082	.7095	.7132	.7158	**.7197**	.7126
512_10	.7106	.7128	.7163	.7195	**.7244**	.7143
1024_10	.7121	.7150	.7185	.7223	**.7270**	.7165
128_12	.6970	.6980	.6996	.7022	**.7033**	.7009
256_12	.7015	.7026	.7062	.7087	**.7129**	.7053
512_12	.7045	.7066	.7104	.7133	**.7179**	.7082
1024_12	.7064	.7089	.7125	.7163	**.7209**	.7107

Table 1: Comparison of $S_i^=$ pivoting rules (average fitness of local optima reached from 1000 hill-climbings distributed on 10 instances). Recall that $S_b^=$ is best improvement, $S_w^=$ is worst improvement, and $S_w^{\geqslant}$, which is reported for comparison, is first improvement.

N_K	$S_b^=$	$S_{q1}^=$	$S_m^=$	$S_{q3}^=$	$S_w^=$	$S_w^{\geqslant}$
128_1	**.7103**	.7065	.7036	.7022	.6992	.7061
256_1	**.7127**	.7108	.7083	.7058	.7035	.7092
512_1	**.7011**	.6970	.6943	.6917	.6895	.6962
1024_1	**.7049**	.7013	.6981	.6954	.6930	.6998
128_2	**.7454**	.7419	.7418	.7382	.7386	.7397
256_2	**.7316**	.7288	.7273	.7257	.7233	.7279
512_2	**.7257**	.7240	.7219	.7196	.7182	.7220
1024_2	**.7233**	.7199	.7196	.7175	.7150	.7192
128_4	.7569	.7569	.7595	.7578	**.7604**	.7544
256_4	.7466	.7502	.7510	.7507	**.7544**	.7477
512_4	.7413	.7435	.7429	.7448	**.7470**	.7425
1024_4	.7363	.7366	.7389	.7410	**.7439**	.7372
128_6	.7569	.7575	.7579	.7589	**.7603**	.7554
256_6	.7449	.7512	.7524	.7536	**.7564**	.7507
512_6	.7425	.7420	.7443	.7476	**.7528**	.7435
1024_6	.7368	.7388	.7407	.7446	**.7478**	.7394
128_8	.7532	.7521	**.7562**	.7533	.7560	.7504
256_8	.7433	.7428	.7464	.7483	**.7524**	.7427
512_8	.7369	.7409	.7407	.7442	**.7486**	.7410
1024_8	.7308	.7343	.7371	.7395	**.7457**	.7365
128_10	.7432	.7447	**.7480**	.7436	.7471	.7448
256_10	.7356	.7344	.7380	.7420	**.7453**	.7373
512_10	.7317	.7315	.7348	.7377	**.7425**	.7337
1024_10	.7266	.7307	.7335	.7348	**.7401**	.7312
128_12	.7370	.7350	.7373	.7389	**.7400**	.7392
256_12	.7314	.7280	.7320	.7348	**.7375**	.7300
512_12	.7255	.7261	.7293	.7321	**.7360**	.7271
1024_12	.7228	.7222	.7264	.7307	**.7338**	.7238

Table 2: Average fitness of best local optima reached for each type of instance (10 instances per NK parametrization).

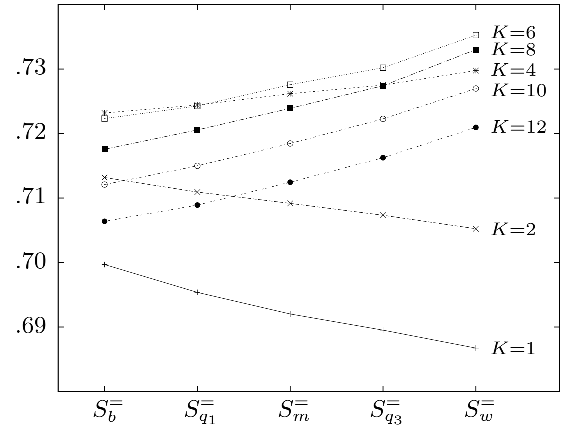

Figure 2: $S_i^=$ pivoting rules results with respect to landscape ruggedness, for a fixed size N=1024. Similar overall tendency is observed for other values of N (see table 1).

worst improvement should be observed also in an iterated local search context (considering a fixed number of local search realized).

In the graphic of figure 3, we report the average fitness values obtained by the 12 studied climbers versions (for $N = 1024$). Methods are ordered according to the average rank of solutions selected by the pivoting rule. For each ruggedness level (K), we observe a monotonic or nearly monotonic evolution of the average final fitnesses. This confirms that the local search efficiency is directly correlated (negatively or positively) with the fitness improvement realized at each step of the search. The table included in figure 3 reports numerical results for $K = 12$. This allows to visualize the monotonic evolution of results obtained by the generic ranking-based pivoting rules ($S^=$, $S^{\geqslant}$, and $S^{\leqslant}$).

The different results obtained and reported in tables 1 and 2, and in figure 3, show a clear interest in performing worst improvement hill-climbings while considering rugged landscapes. However, achieving a worst improvement search naturally requires most solution evaluations. When comparing worst improvement against best improvement, attaining a local optimum requires more moves: each best improvement step significantly increases the fitness value while worst improvement focus on minimizing the fitness benefit at each step of the search. First improvement naturally allows to avoid the systematic generation of the entire neighborhood, which drastically reduces the number of evaluations needed to reach the local optima. In the next section, we focus on the number of evaluations, and propose ways to keep advantages of worst improvement with a reduced number of solution evaluations.

4. APPROXIMATING WORST IMPROVEMENT

First results emphasizes that for climbing efficiently rugged landscapes, the search should focus on selecting the improving neighbors which achieve lower fitness values. However, such a strategy is time-consuming for two reasons. First, in

Strategy	b	q_1	m	q_3	w
$S^{=}$	.7064	.7089	.7125	.7163	**.7209**
$S^{\geqslant}$	.7064	.7072	.7083	.7094	.7107
$S^{\leqslant}$	.7107	.7133	.7158	.7185	.7209

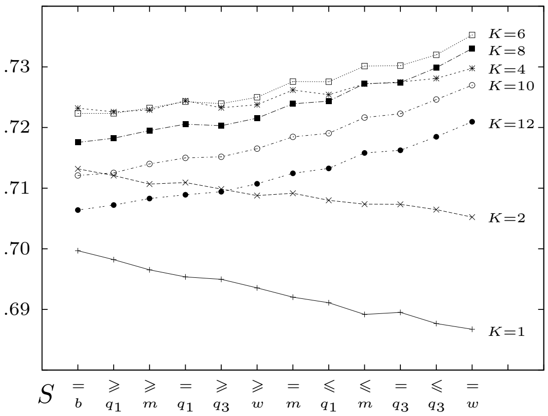

Figure 3: Performance of 12 ranking-based pivoting rules on 7 sets of NK-landscapes grouped by ruggedness (k=1 to 12, N=1024 only). Strategies are sorted according to the average ranking values $\frac{\text{avg\_rank\_of\_selected\_neighbor}-1}{\text{number\_of\_improving\_neighbors}-1}$ (from $S^{=}_{B}$ to $S^{=}_{W}$: 0, 1/8, 1/4, 1/4, 3/8, 1/2, 1/2, 5/8, 3/4, 3/4, 7/8, 1). Table (on top): numerical values for K=12.

minimizing the improvement realized at each step, we tend to greatly increase the number of steps needed to reach a local optimum. Second, as for best improvement, the whole neighborhood has to be generated for ensuring the selection of the worst neighbor — unless an incremental neighborhood evaluation is feasible.

Here, we propose to approximate the complete worst improvement strategy by generating the neighborhood partially. Such a mechanism is commonly used for approximating best improvement on large neighborhoods [8]. We propose here to select the worst improving neighbor of a solution within a subset of its neigborhood. More precisely, we introduce the *partial worst improvement* (denoted as $\mathcal{W}_\kappa$), which consists in stopping the neighborhood generation when κ improving neighbors are found (unless the whole neighborhood has been evaluated). At each step of the search, $\mathcal{W}_\kappa$ selects the configuration with the lowest fitness among the (at most) κ improving neighbors. As a consequence, the parameter κ directly affects the number of solution evaluations needed to perform a hill-climbing step. Note that $\mathcal{W}_1$ is first improvement, while $\mathcal{W}_N$ corresponds to the (complete) worst improvement.

In the experiments, several partial worst improvement versions have been tested: $\mathcal{W}_2$, $\mathcal{W}_4$, $\mathcal{W}_8$, $\mathcal{W}_{16}$, and also $\mathcal{W}_1$ and $\mathcal{W}_N$ for comparison. Experimental process consists then in the comparison of 6 hill-climbing versions which are competing on the set of 280 NK-landscapes used in the previous section. The experimental protocol remains also unchanged in terms of starting points, number of executions, and stopping criterion. Let us precise that neighbors are always generated in a random order. Table 3 reports results on rugged instances only ($K \geqslant 4$), where worst improvement dominates

N_K	Best	First	Partial Worst				Worst
	-	$\mathcal{W}_1$	$\mathcal{W}_2$	$\mathcal{W}_4$	$\mathcal{W}_8$	$\mathcal{W}_{16}$	$\mathcal{W}_N$
128_4	.7216 5k	.7207 1k	.7224 2k	.7237 4k	.7234 8k	.7243 16k	**.7250** 31k
256_4	.7211 19k	.7218 2k	.7233 5k	.7254 10k	.7257 21k	.7262 41k	**.7274** 136k
512_4	.7228 76k	.7232 5k	.7248 11k	.7262 23k	.7278 49k	.7281 97k	**.7285** 568k
1024_4	.7232 302k	.7238 12k	.7253 25k	.7270 54k	.7286 112k	.7291 223k	**.7298** 2336k
128_6	.7170 4k	.7187 1k	.7214 2k	.7228 5k	.7242 10k	.7240 21k	**.7251** 46k
256_6	.7207 15k	.7234 2k	.7264 5k	.7285 12k	.7301 26k	.7309 56k	**.7313** 221k
512_6	.7214 63k	.7239 5k	.7272 12k	.7298 28k	.7315 63k	.7328 135k	**.7338** 979k
1024_6	.7223 251k	.7250 13k	.7280 29k	.7310 65k	.7330 145k	.7343 311k	**.7353** 4151k
128_8	.7124 3k	.7136 1k	.7168 2k	.7194 5k	.7197 11k	.7192 24k	**.7199** 56k
256_8	.7147 13k	.7179 2k	.7218 5k	.7243 13k	.7259 30k	**.7267** 66k	**.7267** 284k
512_8	.7163 53k	.7200 5k	.7242 13k	.7266 31k	.7285 73k	.7297 163k	**.7306** 1324k
1024_8	.7176 214k	.7215 13k	.7251 31k	.7285 74k	.7306 172k	.7316 382k	**.7330** 5837k
128_10	.7049 3k	.7078 1k	.7103 2k	.7122 5k	.7116 11k	**.7127** 24k	.7115 59k
256_10	.7082 11k	.7126 2k	.7159 5k	.7176 13k	.7196 32k	**.7201** 71k	.7198 325k
512_10	.7106 46k	.7143 5k	.7190 13k	.7212 33k	.7232 81k	.7237 183k	**.7243** 1593k
1024_10	.7121 187k	.7165 14k	.7204 33k	.7235 80k	.7255 189k	.7265 439k	**.7270** 7325k
128_12	.6970 2k	.7009 1k	.7021 2k	.7035 5k	**.7037** 11k	.7028 24k	.7033 59k
256_12	.7015 10k	.7053 2k	.7089 5k	.7105 13k	.7124 33k	.7122 74k	**.7129** 346k
512_12	.7045 41k	.7082 5k	.7127 13k	.7151 34k	.7168 85k	**.7179** 199k	**.7179** 1816k
1024_12	.7064 166k	.7107 14k	.7150 34k	.7178 84k	.7197 206k	.7206 487k	**.7210** 8544k

Table 3: Comparison of first improvement, worst improvement, and partial worst improvement pivoting rules (average fitness of local optima reached from 1,000 hill-climbings distributed on 10 instances).

other strategies presented in the previous section. Average results and statistical significance are provided as described in section 3.2. We also report, for each instance, the average number of solution evaluations needed to reach a local optima (in thousands).

Let us recall that $\mathcal{W}_1$ and $\mathcal{W}_N$ were respectively denoted as $S^{=}_{w}$ and $S^{\geqslant}_{w}$ in table 1, then similar results can be retrieved for these methods in both tables, ie. worst improvement outperforms first improvement when $K \geqslant 4$. Globally, according to figure 4, which graphically summarizes results of table 3, $\mathcal{W}_N \succ \mathcal{W}_{16} \succ \mathcal{W}_8 \succ \mathcal{W}_4 \succ \mathcal{W}_2 \succ \mathcal{W}_1$, considering that $A \succ B$ denotes that hill-climbing algorithm A outperforms algorithm B in average on a particular set of instances. This confirms the overall tendency observed in the previous section, which states that when first improvement outperforms best improvement, the climbing efficiency is inversely proportional to the quality of the selected improving neighbors. Nevertheless, on most instances, a relatively low value of κ (regarding to N) is sufficient to obtain a partial worst improvement whose results are not statistically outperformed by the complete worst improvement.

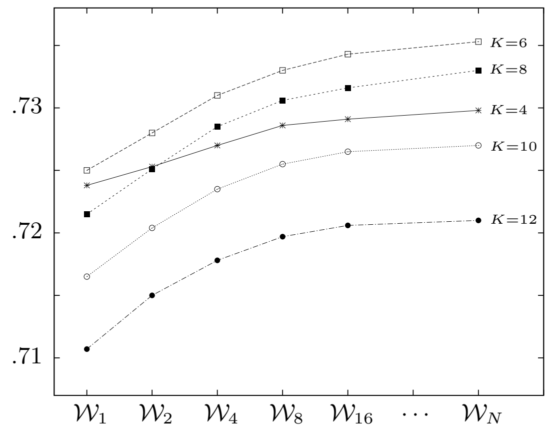

Figure 4: Average results of partial worst improvement strategies ($\mathcal{W}_\kappa$, with $\kappa \in \{1, 2, 4, 8, 16, 1024\}$) on rugged $N = 1024$ instances ($K \in \{4, 6, 8, 10, 12\}$).

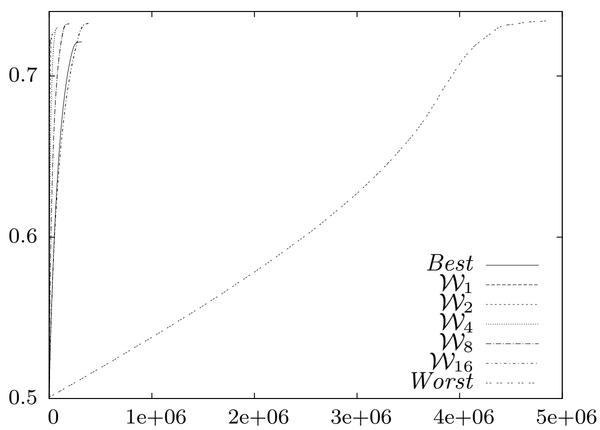

Figure 5: Average convergence of best, worst, and 4 partial worst improvements on a 1024_6 NK-landscape (number of evaluations in horizontal axis, average fitness in vertical axis).

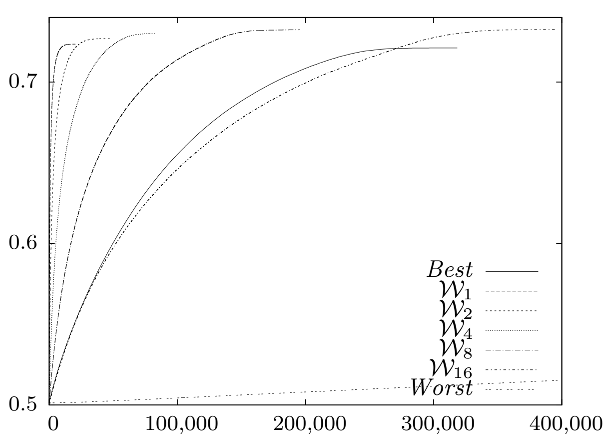

Figure 6: Average convergence of best, worst, and 4 partial worst improvements (focus on the first 400,000 evaluations).

The use of such technique is particularly interesting since it clearly requires less solution evaluations than worst improvement. In most cases, with an accurate κ value, a partial worst improvement can yield statistically comparable local optima while requiring significantly less solution evaluations than a complete worst improvement. Setting κ consists in determining the best compromise between hill-climbing efficiency and computational costs. Note that despite best and (complete) worst improvement both need to evaluate the whole neighborhood at each step of the search, the number of solution evaluations required by worst improvement is significantly greater than by best improvement since the number of (small) steps necessarily increases. The computational difference is especially observed during the first part of the search, when improving neighbors are numerous.

In figure 5, we report the average fitness evolution with respect to the number of solutions evaluated on a 1024_6 instance. This figure emphasizes that the computation cost of worst improvement is really high when compared to the other strategies. It is interesting to notice that although the convergence speed of worst improvement is low, it tends to increase during the search[1]. A similar evolution was observed for all instances.

Figure 6 reports the results outputted in figure 5 with a focus on the first 400,000 evaluations which suffice to terminate best and partial worst improvements. This figure allows us to visualize, on non-smooth NK-landscapes, the clear superiority of $\mathcal{W}_1$ (first improvement), $\mathcal{W}_2$, $\mathcal{W}_4$, $\mathcal{W}_8$, and $\mathcal{W}_{16}$ against best improvement. Moreover, the tradeoff between convergence speed and quality of obtained local optima is easy to observe. Increasing κ allows to reach better local optima but with higher computation costs. In particular, setting κ to 2 suffices to improve results obtained by first improvement ($\kappa = 1$) without increasing drastically the number of solution evaluations. When $\kappa = 16$, the observed convergence speed is similar to best improvement, but searches are trapped later in local optima.

5. CONCLUSION

Climbers are often considered as basic components of advanced search methods. However, influence of their conception choices are rarely discussed through advanced studies. In particular, most studies only focus on best and/or first improvement strategies. In this paper we focused on the design of hill-climbing pivoting rules, the aim being to reach high local optima in various landscapes. In particular, we first evaluate the ability of the worst improvement strategy to reach better configurations. Experiments show that worst improvement is more efficient than both first and best improvement strategies while exploring rugged NK-landscapes. Results also show that the average quality of the attained local optima is negatively correlated with the quality of selected neighbors: worst are the ranks of selected improving neighbors, better are the average local optima attained by the hill-climbing strategy. Results on smooth NK-landscapes lead to the opposite conclusion.

[1]For each worst improvement execution, note that the convergence speed increases during the whole search, and especially during the last steps. However, on figure 5, the worst improvement convergence seems slowing down after 4.10^6 average solution evaluations. It can be explained by the fact that the 100 executions reach a local optimum between 4.10^6 and 5.10^6 evaluations, but not simultaneously.

The second part of this study focused on reducing the computational cost induced by worst improvement. To this aim, we proposed intermediate pivoting rules between first and worst improvement. It mainly consists in combining the low computational cost of first improvement with the ability of worst improvement to reach good local optima. Experiments show that choosing the worst improving neighbor among a fixed number κ of potential improving moves leads to good tradeoffs between quickness (first improvement) and efficiency (worst improvement). In particular, choosing an adequate κ allows to define a partial worst improvement climber, whose results are statistically comparable to worst improvement ones while requiring a reduced computational effort.

Perspectives of this work include the application of worst-based improvement principle to classical combinatorial problems. In [1], we showed that first improvement outperforms best improvement in many cases when applied to flow-shop, QAP and MAXSAT problems. We can expect that worst improvement should outperforms both classical pivoting rules also on these problems. Naturally, some adaptation of this study have to be proposed in order to tackle combinatorial landscapes containing neutrality, which is a particularity of numerous landscapes derived from combinatorial problems.

Other perspective includes the extension of this analysis to Iterative Local Search methods [5]. Indeed, it should be interesting to determine if improving climbers can help to improve local search-based metaheuristics.

Acknowledgements

This work is partially supported by the Pays de la Loire Region (France) within the LigeRO project.

6. REFERENCES

[1] Matthieu Basseur and Adrien Goëffon. Hill-climbing strategies on various landscapes: an empirical comparison. In Christian Blum and Enrique Alba, editors, *GECCO*, pages 479–486. ACM, 2013.

[2] Pierre Hansen and Nenad Mladenović. First vs. best improvement: An empirical study. *Discrete Applied Mathematics*, 154(5):802–817, 2006.

[3] Holger Hoos and Thomas Stützle. *Stochastic Local Search: Foundations & Applications*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[4] Stuart A. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, USA, 06 1993.

[5] Helena R. Lourenço, Olivier Martin, and Thomas Stützle. Iterated local search. In Fred Glover and Gary Kochenberger, editors, *Handbook of Metaheuristics*, volume 57 of *International Series in Operations Research & Management Science*, pages 321–353. Kluwer Academic Publishers, Norwell, Massachusetts, USA, 2002.

[6] Peter Merz and Bernd Freisleben. Fitness landscapes and memetic algorithm design. *New ideas in optimization*, pages 245–260, 1999.

[7] Melanie Mitchell, Stephanie Forrest, and John H Holland. The royal road for genetic algorithms: Fitness landscapes and GA performance. In *Proceedings of the first european conference on artificial life*, pages 245–254. Cambridge: The MIT Press, 1992.

[8] Nenad Mladenović and Pierre Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100, November 1997.

[9] Alain Ratle. Kriging as a surrogate fitness landscape in evolutionary optimization. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 15(01):37–49, 2001.

[10] Darrell Whitley, Adele E. Howe, and Doug Hains. Greedy or not? best improving versus first improving stochastic local search for MAXSAT. In *Proceedings of Twenty-Seventh Conference on Artificial Intelligence*. AAAI, 2013.

[11] S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution. *Proceedings of the VI International Congress of Genetics*, 1:356–366, 1932.

[12] Mihalis Yannakakis. The analysis of local search problems and their heuristics. In *Proceedings of the seventh annual symposium on Theoretical aspects of computer science*, STACS 90, pages 298–311, New York, NY, USA, 1990. Springer-Verlag New York, Inc.