A Study on the Short-Term Prohibition Mechanisms in Tabu Search

Luca Di Gaspero[1] and **Marco Chiarandini**[2] and **Andrea Schaerf**[1]

Abstract. Tabu Search (TS) is a well known local search method which has been widely used for solving AI problems. Different versions of TS have been proposed in the literature, and many features of TS have been considered and tested experimentally. The feature that is present in almost all TS variants is the so called (short-term) tabu list which is recognised as the crucial issue of TS. However, the definition of the parameters associated with the tabu list remains in most TS applications still a handcrafted activity.

In this work we undertake a systematic study of the relative influence of few relevant tabu list features on the performances of TS solvers. In particular, we apply statistical methods for the design and analysis of experiments. The study focuses on a fundamental theoretical problem (GRAPH COLOURING) and on one of its practical specialisation (EXAMINATION TIMETABLING), which involves specific constraints and objectives. The goal is to determine which TS features are more critical for the good performance of TS in a general context of applicability.

The general result is that, when the quantitative parameters are well tuned, the differences with respect to qualitative parameters become less evident.

1 INTRODUCTION

Local search is a search paradigm which has evidenced to be very effective for a large number of AI problems [18]. Tabu Search (TS) [15] is one of the most successful local search methods, which has been widely used for solving AI problems. Different versions of TS have been proposed in the literature, and many features of TS have been considered and tested experimentally. They range from long-term tabu, to dynamic cost functions, to strategic oscillation, to elite candidate lists, to complex aspiration criteria, just to mention some (see [15] for an overview).

The feature that is included in virtually all TS variants is the so called *(short-term) tabu list*. The tabu list is indeed recognised as the basic ingredient for the effectiveness of a TS-based solution, and its behaviour is a crucial issue of TS.

Despite the importance of a correct empirical analysis, which has been emphasised in the general context of heuristic methods [2, 19] and even in the specific case of TS [28], the definition of the parameters associated with the tabu list remains in most research work still a handcrafted activity. Often, the experimental work behind the parameter setting remains hidden or is condensed in a few lines of text reporting only the final best configuration. Even the recently introduced *racing* methodology for the tuning of algorithms [4] only allows to determine the best possible configuration. These procedures are certainly justified from a practical point of view, but a more detailed description of the behaviour and sensibility of the algorithm with respect to its different factors and parameters is surely of great interest in the research field.

In this work, we aim at determining which factors of basic TS are important and responsible for the good behaviour of the algorithm and its robustness (i.e., the capability of exhibiting a stable behaviour across several set of instances). In contrast to the one-factor-at-a-time approach used in [28], we use different experimental design techniques [23], with parametric and non-parametric analysis of variance and racing algorithms. The suitability and comparison of these techniques in the field of search algorithms is still an open issue, hence we use more than one method to provide complementary support to their conclusions. We focus the analysis on both a fundamental theoretical problem, namely the GRAPH COLOURING problem, and on one of its practical specialisation, the EXAMINATION TIMETABLING problem, for which there is a consistent literature and many benchmark instances.

Although the statistical methods applied indicate that an algorithmic configuration of general validity does not arise, they point out that for these problems TS is robust enough w.r.t. the qualitative features, and thus it is reasonable to focus mainly on tuning the quantitative parameters. In addition, we strengthen our analysis with a comparison with previous work thus providing absolute validity to the results here presented and avoiding the pitfall of a self-standing work. We show indeed that the so-configured solution algorithm produces results that are comparable with the state-of-art.

2 PROBLEM DEFINITIONS

The first problem we consider is the classical GRAPH (VERTEX) COLOURING problem in its chromatic number formulation [14]. Given an undirected graph $G = (V, E)$, the GRAPH COLOURING problem consists in finding the minimal number of colours such that there exists an assignment of exactly one colour to each vertex and vertices connected by an edge $e \in E$ receive different colours.

The EXAMINATION TIMETABLING problem consists in scheduling the exams of a set of university courses in rooms and timeslots. In doing this, overlaps of exams with students in common must be avoided, exams must be fairly spread for each students, and room capacity constraints must be satisfied [26]. Among the different EXAMINATION TIMETABLING versions, we focus on the formulation proposed by Burke *et al.* [5]. According to this formulation, the examination session consists of a given fixed number of periods, and each period belongs to a specific day. Exams with common students

[1] DIEGM, University of Udine, via delle Scienze 208, I-33100, Udine, Italy, email: `l.digaspero@uniud.it||schaerf@uniud.it`

[2] Department of Mathematics & Computer Science, University of Southern Denmark, Campusvej 55, DK-5230 Odense M – Denmark, email: `marco@imada.sdu.dk`

cannot be scheduled in the same period (first order conflicts), but they should also be not scheduled in adjacent periods of the same day (second order conflicts). It is easy to see, that this formulation corresponds to the GRAPH COLOURING problem in which the exams represent the vertices, the periods the colours, and two nodes are connected if they have students in common. The differences stem from the fact that for the EXAMINATION TIMETABLING the number of periods is fixed, whereas the objective function consists in the second order conflicts (weighted by the number of students).

3 TABU SEARCH BASICS

Local search is based on the idea of navigating a space of solutions (called *search space*) by iteratively moving from one state to one of its "neighbours", which are obtained by applying a simple local change to the current solution. Several criteria for the selection of the move can be defined upon this idea. In particular, TS explores the full current neighbourhood and selects, as the new current state, the neighbour that gives the minimal value of a cost function, independently of whether its cost is less or greater than the current one. This selection allows the search to *escape* from local minima, but creates the risk of cycling among a set of states. In order to prevent cycling, TS uses a prohibition mechanism based on memory of past iterations: a list (called tabu list) stores the most recently accepted moves, so that the *inverses* of the moves in the tabu list are forbidden (i.e., the moves that are leading again toward the just visited local minimum).

The precise instantiation of the above general scheme can be done in different ways. In details, the three main features related to the tabu list are the following:

Neighbourhood exploration: Often exploring the full neighbourhood and determining the best move can be computationally too expensive for a tabu search algorithm to perform well. The exploration of the neighbourhood is therefore reduced to a part of it. The part of the neighbourhood explored can be determined probabilistically, heuristically or through considerations on structural properties of the problem that guarantee no improvement for the neighbours omitted.

Prohibition power: The prohibition power determines which moves are prohibited by the fact that a move is in the tabu list. A move is normally composed of several attributes; depending on the prohibition power, the tabu status can be assigned only to the move with the same values for all attributes or to the set of moves that have one or more attribute equal.

List dynamics: The list dynamics determines for how many iterations a move remains in the tabu list. This can be a fixed value, or a value selected randomly in an interval, or a value selected adaptively on the basis of the current state of the search.

In the next section, we propose different alternatives for the definition of the above features in the two case studies.

4 INSTANTIATION OF TABU SEARCH FEATURES

The search space considered is the same for both problems and is composed by one variable for each vertex (resp. exam) whose domain corresponds to the set of colours (resp. periods). The neighbourhood of a state is composed by all states reachable by changing exactly one colour at a vertex. In this setting, a local search move m can be identified by three attributes: a vertex v, its old colour o and its new colour n and it can be identified by the triple $m = \langle v, o, n \rangle$. Note that the two attributes v and n suffice to identify the move uniquely.

The exploration of the neighbourhood can be restricted to only those vertices involved in a conflict (i.e., adjacent vertices that are assigned the same colour). In this way, it is guaranteed that no improving move is missed and, although the worst case complexity remains $O(kn)$, in practise we obtain a large reduction of search effort especially when the search explores states that are close to good solutions. This reduction is very helpful, and we do it in all cases. A further possible reduction of the neighbourhood exploration could be achieved by means of the MinConflict heuristic [22]. Consequently, we decide to implement the following three neighbourhood exploration reductions:

No reduction (NoRed): for each vertex involved in a conflict all the colours are tried and the overall best move is taken.

MinConflict Vertex (MinConf$_v$): for a vertex randomly chosen among those involved in a conflict the colour that yields the best move is chosen.

MinConflict Colour (MinConf$_c$): for all vertices involved in a conflict a colour is selected at random and the best overall move is taken.

For the prohibition power, assuming that the move $\langle v, n, o \rangle$ is in the tabu list, we consider the following three alternatives (where the underscore means "any value"):

Strong: All moves of the form $\langle v, \_, \_ \rangle$ are prohibited.
Medium: All moves of the form $\langle v, \_, o \rangle$ are prohibited
Weak: Only the single move $\langle v, n, o \rangle$ is prohibited

In other terms, in the first case, it is not allowed to recolour the vertex in the tabu iterations. In the second case, it is not allowed to recolour the vertex with its old colour. In the third case, it is not allowed only to make the reverse of the tabu move.

Finally, in regards to the list dynamics, we also consider two possibilities:

Interval: The size of the list can vary in the range $[t_b, t_b + w]$ of width w. Each accepted move remains in the list for a number t of iterations equal to $t_b + Random(0, w)$, where $Random(a, b)$ is the uniform integer-valued random distribution in $[a, b]$

Adaptive: The value t depends on the current state. We use the formula (proposed in [13]) $t = \lfloor \alpha * c_s \rfloor + Random(0, t_b)$ where t_b and α (real value) are parameters, and c_s is the number of conflicts in the current state s.

A special case of Interval is the case, called Fixed, in which $w = 0$, so that the tabu list is a queue of size t_b. At each iteration, the accepted move gets in and the oldest one gets out. All moves remain in the list for exactly t_b iterations.

As a final remark, it is worth to note that according to the common practise in Tabu Search applied to the GRAPH COLOURING problem (e.g., [16]), the issue of finding the minimum number of colours is solved as a sequence of decision problems. The procedure stops and returns the value k, as soon as no solution with $k - 1$ colours can be found.

5 EXPERIMENTAL METHODOLOGY

It is recognised that studying local search algorithms in an analytical way is a complex task which often yields results with scarce practical implications [17, 24]. The assessment of these algorithms is therefore carried out through empirical tests and the experimental design theory, largely applied in the engineering fields, provides the guidelines to do this in a methodological way [25, 3, 8].

The tabu search features introduced above are *treatment factors* (i.e., aspects that are hypothesised to have an effect on the behaviour of the algorithm) and they can be subdivided in two types: quantitative and qualitative. The three qualitative factors are the neighbourhood exploration, the prohibition power and the list dynamics, and consist each of different levels. The quantitative factors are the numerical parameters of the list dynamics strategy and may assume an unlimited number of values. There are two issues that complicate the factorial design: (i) the qualitative parameters do not cross with all other factors (for example, there is no α parameter with Interval list dynamics); (ii) the importance of each of the two qualitative factors strongly depends on the values assigned to the underlying quantitative parameters.

We follow two approaches in the analysis: a *bottom-up* approach consisting in first studying the best configurations of quantitative parameters for each combination of qualitative factors and then analysing the effects of the latter; and a *top-down* approach consisting in analysing the effects of the qualitative factors in a first step and then refining the analysis on the quantitative parameters for the best configuration. Both these approaches have a drawback: the bottom-up approach may require a huge amount of experiments and analysis; the top-down approach may be biased by the few values chosen for the qualitative parameters.

A further issue is the level of the analysis to be undertaken. Clearly discriminating results on single instances is not meaningful in practise; however, both an aggregate analysis and a separated analysis on single classes of instances are relevant for practical applications. We include, therefore, in the case of GRAPH COLOURING six classes of instances[3] determined by the type of random graphs (geometric and uniform) and edge density (0.1, 0.5, 0.9). We maintain, instead, the size of the graph fixed to 1000 vertices.

Finally, a matter of concern in the statistical analysis of algorithms is the choice between parametric and non-parametric methods. We opt even in this case for maintaining both approaches. Indeed, being based on two different transformations of the results (normalisation and ranking) the two approaches are likely to produce a different statistical inference. From a theoretical perspective, the parametric analysis of variance through the F-ratio (ANOVA) is a procedure which has been recognised as particularly robust against deviation from the assumption of normal distribution of data [23]. The non-parametric analysis of variance through the Friedman test [9] by ranks is instead grounded on less assumptions but is unable to study interactions between factors.

6 ANALYSIS

We split our analysis in two parts, one for each problem, and we report the results on the two problems separately.

6.1 Graph Colouring

The experimental designs and the values considered for the factors of analysis are reported in Table 2a. We consider the random seed to generate different instances as a blocking factor [10]. The distinction between two designs, $\mathcal{D}_A$ and $\mathcal{D}_B$, is necessary because of the different meaning of parameter t_1. Note that the degenerate case $t_1 = w = 0$ in Interval list dynamics corresponds to a fixed tabu length depending only on the parameter $t_2 = t_b$.

For each algorithmic configuration we collect one run on each instance. The response variable is the minimal number of colours found in a run after 10 million algorithm iterations. In the parametric analysis the number of colours $r(i)$ obtained by each algorithmic configuration on an instance i is normalised by the formula $e(r, i) = \frac{r(i)-r_{best}(i)}{r_{RLF}(i)-r_{best}(i)}$, where r_{best} is the best result achieved on instance i by all algorithms and r_{RLF} is the average solution produced by the RLF heuristic [20], which is used as the starting solution for the tabu search. Therefore, we consider $e(r, i)$ as our response variable. In the non-parametric analysis, instead, results are ranked within instances and no normalisation is needed. A level of significance of 5% is maintained over all the statistical tests.

In the top-down approach we maintain the two designs $\mathcal{D}_A$ and $\mathcal{D}_B$ separated. A parametric analysis of variance (ANOVA) by means of the F-ratio provides a general view of the significant effects. ANOVA assumes a linear model of responses and treatments. We use an automated stepwise model selection based on AIC [1] to distinguish which treatment combinations are significant enough to be included in the model. Accordingly, all main effects and interactions between two factors result significant. The analysis should therefore discriminate too much and this approach is not helpful. It is important to remark however that also the two instance class factors interact with the algorithm factors hence indicating that results might vary between instance classes.

In the bottom-up approach we use a sequential testing methodology for tuning of the parameters t_1 and t_2. In particular we adopt the RACE algorithm proposed by Birattari [4] based on t-test for multiple comparisons and Holm's adjustment of the p-value. This procedure determines the best t_1 and t_2 for each combination of the other factors and allows to remove the quantitative parameters from the analysis. What remains is a unique design $\mathcal{D}$ of 5 qualitative factors that can be analysed by ANOVA. Interaction with instance type and density is significant and indicates that a fine grained analysis should discriminate among the instance classes. We decide to focus, however, primarily on robust results, i.e., settings that are robust across multiple classes of instances. Hence, simplifying the analysis we remain with 3 algorithmic factors and study their main effects and interactions. In Table 3a and Figure 1 we report the results of this analysis. The most important result, emphasised from the visualisation of the analysis through interaction plots in Figure 1, is the bad performance of the MinConf neighbourhood exploration strategy on the colour. Another ineffective choice is the employment of a Strong prohibition power.

The non-parametric analysis through the Friedman analysis of variance followed by its extension in to multiple comparisons [9] confirms these results. In these tests, the factors studied are the combination of neighbourhood exploration, prohibition power and list dynamics (hence, a unique factor with $3 \times 4 \times 2$ levels) and the instance blocking factor. The conclusions above are recognised by the fact that all levels composed by the MinConf$_c$ neighbourhood exploration strategy or Strong prohibition power are statistically inferior to the others.

6.2 Examination Timetabling

For space limits, we restrict the presentation of results of the bottom-up analysis only. Given the results on the GRAPH COLOURING we consider only the NoRed neighbourhood. The values of the param-

[3] Instances are newly generated using the Culberson's generator available at www.cs.alberta.ca/~joe/Coloring

Table 1: Experimental designs employed in the experimentation for the two problems

Factors	Design $\mathcal{D}_A$	Design $\mathcal{D}_B$
Qualitative		
Neighbourhood	{NoRed, MinConf$_v$, MinConf$_c$}	
Prohibition power	{Weak, Medium, Strong}	
List dynamics	Interval	Adaptive
Quantitative		
t_1=$w\|\alpha$	$\{0, 5, 10\}$	$\{0.3, 0.6, 0.9\}$
t_2=t_b	$\{5, 10, 15, 20\}$	$\{5, 10, 20\}$
Blocking		
Type	{Uniform, Geometric}	
Density	$\{0.1, 0.5, 0.9\}$	
Random seed	$[1, \ldots, 10]$	

(a) Experimental designs for GRAPH COLOURING

Factors	Design $\mathcal{D}_A$	Design $\mathcal{D}_B$
Qualitative		
Neighbourhood	{NoRed}	
Prohibition power	{Weak, Medium, Strong}	
List dynamics	Interval	Adaptive
Quantitative		
t_1=$w\|\alpha$	$\{0, 5, 10\}$	$\{0.3, 0.5, 0.8\}$
t_2=t_b	$\{5, 10, 15, 20, 25\}$	$\{5, 10, 20, 30\}$
Blocking		
Instances	Carter's data-set [7]	

(b) Experimental designs for EXAMINATION TIMETABLING

Table 2: The p-values corresponding to the ANOVA F-ratio in the bottom-up analysis performed on the two problems: quantitative parameters have been set by the RACE procedure; blocking factors are not taken into account.

Main Effects	Pr($>F$)	Interaction Effects	Pr($>F$)
Prohibition power	$< 10^{-15}$	Prohibition|Neighb.	$< 10^{-15}$
Neighbourhood	$< 10^{-15}$	Prohibition|List dyn.	0.6450
List Dynamics	0.5971	Neighb.|List dyn.	0.1909

(a) Numerical results of ANOVA performed on GRAPH COLOURING

Main Effects	Pr($>F$)	Interaction Effects	Pr($>F$)
Prohibition power	0.0669	Prohibition p.|List dyn.	0.3738
List Dynamics	0.6896	—	

(b) Numerical results of ANOVA on EXAMINATION TIMETABLING

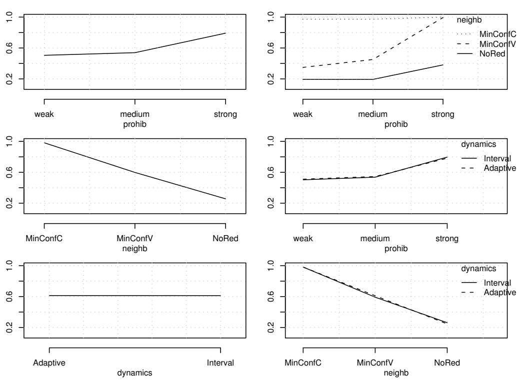

Figure 1: Interaction plots for the factors of the bottom-up analysis on the GRAPH COLOURING problem.

eters tested are shown in Table 2b.

The RACE algorithm is used for the selection of the quantitative parameters. Experiments are carried out on 7 instances taken from Carter's data-set [7]. Each run of a configuration was granted 120 seconds of CPU time on an AMD Athlon 1.5GHz computer running Linux. The outcomes of the RACE procedure are reported in Table 3 that shows the configurations for which no significant difference was found after 50 stages.

Table 3: RACE results for EXAMINATION TIMETABLING

Prohibition power / List dynamics	t_1-t_2	avg $z(r,i)$
Weak/Adaptive	0.3-5 0.5-10 0.5-20 0.3-30	-0.085
Medium/Adaptive	0.3-5 0.8-5 0.3-20 0.5-20 0.8-20	-0.087
Strong/Adaptive	0.3-10 0.5-10 0.5-30	0.010
Weak/Interval	5-5	-0.128
Medium/Interval	5-5 20-25 20-30	-0.084
Strong/Interval	25-25	0.068

Once good values for the parameters have been determined we study main effects and interactions of the two remaining qualitative factors, list dynamics and prohibition power. To this aim, we run a full factorial design collecting 25 replicates per instance. Each experimental unit exploits a different combination of list dynamics, prohibition power and test instance. The result $r(i)$ is the penalty for constraint violation, which has to be minimised. In order to have comparable values across all the tested instances we consider the z-score $z(r,i) = \frac{r(i)-\bar{r}(i)}{s[r(i)]}$ as the response variable in our analysis.

In the analysis of variance instances are treated as blocks and hence their influence on the performance of the algorithms, though recognised, is not taken into account (this entails that the results of the analysis are robust with respect to the set of instances). The results of the F-ratio and the Friedman test indicate as non significant both the main effects and the interactions (The F-ratios are shown in Table 3b, whereas the Friedman test yields a p-value of 0.54).

The visual investigation of results by means of box-plots in Figure 2 reveals indeed that the numerical differences in results are apparently negligible. In conclusion, if the quantitative parameters are well chosen by means of a statistically sound procedure, all configurations of qualitative parameters are equally good.

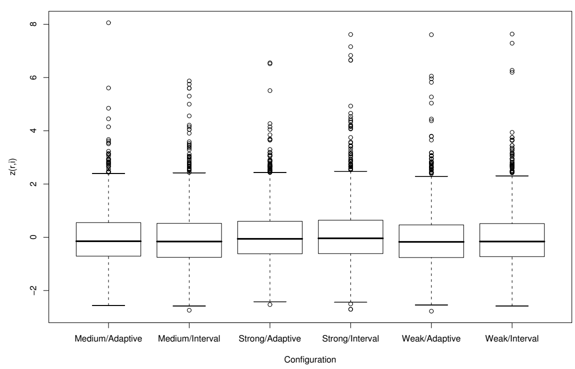

Figure 2: Results of the six configurations for the qualitative features

Finally we compare, in Table 4, our overall best results (in bold) with the currently published ones (see references on the column labels). (The use of best results to infer performance is known in Statistics to be a biased procedure, here however it is justified by the usual practise in the literature.) From Table 4 it is evident that we improved significantly w.r.t. both our previous best results [11] and other results obtained by using similar local search techniques [6]. In addition, although our results are still far from the best results attained by Merlot *et al.* [21], they are in most cases the second best ones.

Table 4: Best results found for each configuration and comparison with the best known results.

Instance	p	W/A	W/D	W/F	M/A	M/D	M/F	S/A	S/D	S/F	[11]	[5]	[6]	[21]
car-f-92	40	271	292	265	263	278	275	297	**224**	242	424	331	268	158
car-s-91	51	46	38	38	53	54	**32**	37	40	33	88	81	74	31
kfu-s-93	20	1148	1027	1103	1047	**914**	984	1172	1104	932	512	974	912	247
nott	23	112	89	109	103	106	**69**	109	112	73	123	269	—	7
nott	26	17	17	17	**7**	15	15	16	13	20	11	53	—	—
tre-s-92	35	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	4	3	2	0
uta-s-92	38	534	544	526	584	572	**507**	565	537	567	554	772	680	334

7 RELATED WORK & CONCLUSIONS

This study has been inspired by the works [2] and [19], and by the more recent work [3], which in different ways trace the guidelines for a correct empirical analysis of algorithms. The application of these guidelines to analyse TS features has been proposed in [28]. With respect to these works, we take into consideration also interaction effects between factors and hence remove the bias of a one-feature-at-a-time procedure.

The work in [27] proposes a similar analysis, although it considers specifically the *lesser* used features of TS, whereas we focus on the most used ones. In addition, also that work uses the one-feature-at-a-time approach, thus missing the interactions among them.

We adopted different statistical procedures, such as ANOVA, Friedman test and racing procedures to corroborate the results and guarantee fairness to all TS features. On the GRAPH COLOURING we have been able to discriminate which components of neighbourhood exploration and prohibition power are important to be selected correctly. Less relevant is instead the choice of the tabu length.

On the EXAMINATION TIMETABLING the analysis provides a similar indication to the results found on the GRAPH COLOURING. No significant influence of both main and interaction effects on the qualitative factors considered have been detected. Nevertheless, the analysis permits to improve the tuning of the numerical parameters and improve over previously published configurations. The results of the final algorithm are comparable with the state-of-the-art solvers.

A common conclusion arising in both studies is that, once a good setting has been found for the quantitative parameters, the differences between some of the qualitative parameters are less pronounced.

More features could be added to TS implementations. One example is the case in which the neighbourhood explored is the union of a set of basic neighbourhood relations (see [12]). In this case, typically, there are different tabu dynamics for moves of different sort. Another example, is the use of long term memory or reactive mechanisms. Extending the scope of this analysis to these and similar cases will be the subject of future work.

REFERENCES

[1] H. Akaike, 'A new look at statistical model identification', *IEEE Trans. on Automatic Control*, **19**, (1974).

[2] R. Barr, B. Golden, J. Kelly, M. Resende, and W. Stewart, 'Designing and reporting on computational experiments with heuristic methods', *J. of Heur.*, **1**, 9–32, (1995).

[3] T. Bartz-Beielstein, 'Experimental analysis of evolution strategies – overview and comprehensive introduction', Technical Report Reihe CI 157/03, SFB 531, Universität Dortmund, Germany, (2003).

[4] M. Birattari, *The Problem of Tuning Metaheuristics as Seen from a Machine Learning Perspective*, Ph.D. dissertation, Université Libre de Bruxelles, Belgium, 2004.

[5] E. Burke, J. Newall, and R. Weare, 'A memetic algorithm for university exam timetabling', in *Proc. of ICPTAT-95*, pp. 241–250, (1995).

[6] M. Caramia, P. Dell'Olmo, and G. F. Italiano, 'New algorithms for examination timetabling', in *Proc. of WAE 2000*, eds., S. Nher and D. Wagner, vol. 1982 of *LNCS*, pp. 230–241. Springer-Verlag, (2000).

[7] M. W. Carter. Examination timetabling dataset. URL: `ftp://ftp.mie.utoronto.ca/pub/carter/testprob/`, 2005. Viewed: January 19, 2006, Updated: June 7, 2005.

[8] M. Chiarandini, D. Basso, and T. Stützle, 'Statistical methods for the comparison of stochastic optimizers', in *Proc. of MIC2005*, eds., K. F. Doerner, M. Gendreau, P. Greistorfer, W. J. Gutjahr, R. F. Hartl, and M. Reimann, Vienna, Austria, (August 2005).

[9] W.J. Conover, *Practical Nonparametric Statistics*, John Wiley & Sons, New York, USA, third edn., 1999.

[10] A. Dean and D. Voss, *Design and Analysis of experiments*, Springer Texts in Statistics, Springer-Verlag, New York, NY, USA, 1999.

[11] L. Di Gaspero and A. Schaerf, 'Tabu search techniques for examination timetabling', in *Proc. of PATAT-2000*, eds., E. Burke and W. Erben, vol. 2079 of *LNCS*, 104–117, Springer-Verlag, (2001).

[12] L. Di Gaspero and A. Schaerf, 'Neighborhood portfolio approach for local search applied to timetabling problems', *J. of Math. Modeling and Algorithms*, **5**(1), 65–89, (2006).

[13] R. Dorne and J. Hao, 'A new genetic local search algorithm for graph coloring', in *Proc. of PPSN V, 5th Int. Conference*, eds., A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, vol. 1498 of *LNCS*, 745–754, Springer-Verlag, (1998).

[14] M. R. Garey and D. S. Johnson, *Computers and Intractability—A guide to NP-completeness*, W.H. Freeman & Co., San Francisco, 1979.

[15] Fred Glover and Manuel Laguna, *Tabu search*, Kluwer Ac. Publ., 1997.

[16] A. Hertz and D. de Werra, 'Using tabu search techniques for graph coloring', *Computing*, **39**, 345–351, (1987).

[17] J. N. Hooker, 'Needed: An empirical science of algorithms', *Operations Research*, **42**(2), 201–212, (1996).

[18] H. H. Hoos and T. Stützle, *Stochastic Local Search Foundations and Applications*, Morgan Kaufmann Publishers, San Francisco, 2005.

[19] D. S. Johnson, 'A theoretician's guide to the experimental analysis of algorithms', in *Data Structures, Near Neighbor Searches, and Methodology: 5^{th} and 6^{th} DIMACS Impl. Challenges*, eds., M. H. Goldwasser, D. S. Johnson, and C. C. McGeoch, 215–250, AMS, (2002).

[20] F. T. Leighton, 'A graph coloring algorithm for large scheduling problems', *J. Res. Natl. Bur. Standards*, **84**, 489–506, (1979).

[21] L. Merlot, N. Boland, B. Hughes, and P. Stuckey, 'A hybrid algorithm for the examination timetabling problem', in *Proc. of PATAT-2002*, eds., E. Burke and P. De Causmaecker, vol. 2740 of *LNCS*, pp. 207–231. Springer-Verlag, (2003).

[22] S. Minton, M. D. Johnston, A. B. Philips, and P. Laird, 'Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems', *Artificial Intelligence*, **58**, 161–205, (1992).

[23] D. C. Montgomery, *Design and Analysis of Experiments*, John Wiley & Sons, sixth edn., 2005.

[24] B.M.E. Moret, 'Towards a discipline of experimental algorithmics', in *Data Structures, Near Neighbor Searches, and Methodology: 5^{th} and 6^{th} DIMACS Impl. Challenges*, eds., M.H. Goldwasser, Johnson D.S., and C.C. McGeoch, num. 59 in DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 197–213, AMS, (2002).

[25] R. L. Rardin and R. Uzsoy, 'Experimental evaluation of heuristic optimization algorithms: A tutorial', *J. of Heur.*, **7**(3), 261–304, (2001).

[26] A. Schaerf, 'A survey of automated timetabling', *AI Review*, **13**(2), 87–127, (1999).

[27] I. Whittley and G. Smith, 'Evaluating some lesser used features of the tabu search algorithm', in *4^{th} EU/ME Workshop on Design and Evaluation of Advanced Hybrid Meta-Heuristics*, (2004).

[28] J. Xu, S. Chiu, and F. Glover, 'Fine-tuning a tabu search algorithm with statistical tests', *Intl. Tran. in OR*, **5**(3), 233–244, (1998).