Thomas Stützle
Mauro Birattari
Holger H. Hoos (Eds.)

LNCS 4638

Engineering Stochastic Local Search Algorithms

Designing, Implementing and Analyzing Effective Heuristics

International Workshop, SLS 2007
Brussels, Belgium, September 2007
Proceedings

Springer

Thomas Stützle Mauro Birattari
Holger H. Hoos (Eds.)

Engineering Stochastic Local Search Algorithms

Designing, Implementing and Analyzing Effective Heuristics

International Workshop, SLS 2007
Brussels, Belgium, September 6-8, 2007
Proceedings

Springer

Volume Editors

Thomas Stützle
Mauro Birattari
Université Libre de Bruxelles (ULB)
CoDE, IRIDIA
Av. F. Roosevelt 50, CP 194/6, 1050 Bruxelles, Belgium
E-mail: {stuetzle,mbiro}@ulb.ac.be

Holger H. Hoos
University of British Columbia
Computer Science Department
2366 Main Mall, Vancouver, BC, V6T 1Z4, Canada
E-mail: hoos@cs.ubc.ca

Library of Congress Control Number: 2007933306

CR Subject Classification (1998): E.2, E.5, D.2, F.2, H.2, I.1.2, I.2.8, I.7

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-74445-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-74445-0 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12111459 06/3180 5 4 3 2 1 0

Lecture Notes in Computer Science 4638

Preface

Stochastic local search (SLS) algorithms enjoy great popularity as powerful and versatile tools for tackling computationally hard decision and optimization problems from many areas of computer science, operations research, and engineering. To a large degree, this popularity is based on the conceptual simplicity of many SLS methods and on their excellent performance on a wide gamut of problems, ranging from rather abstract problems of high academic interest to the very specific problems encountered in many real-world applications. SLS methods range from quite simple construction procedures and iterative improvement algorithms to more complex general-purpose schemes, also widely known as metaheuristics, such as ant colony optimization, evolutionary computation, iterated local search, memetic algorithms, simulated annealing, tabu search and variable neighborhood search.

Historically, the development of effective SLS algorithms has been guided to a large extent by experience and intuition, and overall resembled more an art than a science. However, in recent years it has become evident that at the core of this development task there is a highly complex engineering process, which combines various aspects of algorithm design with empirical analysis techniques and problem-specific background, and which relies heavily on knowledge from a number of disciplines and areas, including computer science, operations research, artificial intelligence, and statistics. This development process needs to be assisted by a sound methodology that addresses the issues arising in the various phases of algorithm design, implementation, tuning, and experimental evaluation. A similarly principled approach is key to understanding better which SLS techniques are best suited for particular problem types and to gaining further insights into the relationship between algorithm components, parameter settings, problem characteristics, and performance.

The aim of *SLS 2007, Engineering Stochastic Local Search Algorithms — Designing, Implementing and Analyzing Effective Heuristics* was to stress the importance of an integration of relevant aspects of SLS research into a more coherent engineering methodology and to provide a forum for research in this direction. The workshop brought together researchers working on various aspects of SLS algorithms, ranging from fundamental SLS methods and techniques to more applied work on specific problems or real-life applications. We hope that this event will lead to an increased awareness of the importance of the engineering aspects in the design and implementation of SLS algorithms, and that it will help to tie together existing activities and to seed new efforts in this promising research area.

The importance and the timeliness of the topic of SLS engineering is witnessed by the more than 50 submissions we received for this workshop. From these submissions, the 12 full and 9 short papers contained in this volume and presented at the workshop were chosen based on a highly selective and rigorous peer-reviewing process; each of them reports results of very promising, ongoing research efforts from, or highly related to, the budding area of SLS engineering. The workshop program was complemented by the Doctoral Symposium on Engineering Stochastic Local Search Algorithms, which was organized by Enda Ridge and Edward Curry, and five tutorials on important topics in SLS engineering given by well-known researchers in the field.

We gratefully acknowledge the contributions of everyone at IRIDIA who helped in organizing SLS 2007. Special thanks go to Enda Ridge and Edward Curry for their enthusiasm in organizing the doctoral symposium. We thank all researchers who submitted their work and thus provided the basis for the workshop program; the members of the Program Committee and the additional referees for their help with the paper selection process; the Université Libre de Bruxelles for providing the rooms and logistic support; and, more generally, all those who contributed to the organization of the workshop. Finally, we would like to thank COMP2SYS,[1] the Belgian National Fund for Scientific Research, and the French Community of Belgium for supporting the workshop.

June 2007

Thomas Stützle
Mauro Birattari
Holger H. Hoos

[1] A Marie Curie Early Stage Training Site funded by the European Commission; more information is available at http://iridia.ulb.ac.be/comp2sys.

Organization

SLS 2007 was organized by IRIDIA, CoDE, Université Libre de Bruxelles, Belgium.

Workshop Chairs

Thomas Stützle	Université Libre de Bruxelles, Belgium
Mauro Birattari	Université Libre de Bruxelles, Belgium
Holger H. Hoos	University of British Columbia, Canada

Program Committee

Thomas Bartz-Beielstein	Cologne University of Applied Sciences, Germany
Roberto Battiti	Università di Trento, Italy
Christian Blum	Universitat Politècnica de Catalunya, Spain
Marco Chiarandini	University of Southern Denmark, Denmark
Carlos Cotta	University of Málaga, Spain
Camil Demetrescu	Università La Sapienza, Italy
Luca Di Gaspero	Università degli Studi di Udine, Italy
Karl F. Doerner	Universität Wien, Austria
Marco Dorigo	Université Libre de Bruxelles, Belgium
Carlos M. Fonseca	University of Algarve, Portugal
Michel Gendreau	Université de Montréal, Canada
Jens Gottlieb	SAP AG, Germany
Walter J. Gutjahr	Universität Wien, Austria
Pierre Hansen	GERAD and HEC Montreal, Canada
Jin-Kao Hao	University of Angers, France
Richard F. Hartl	Universität Wien, Austria
Geir Hasle	SINTEF Applied Mathematics, Norway
David Johnson	AT&T Labs Research, USA
Joshua Knowles	University of Manchester, UK
Arne Løkketangen	Molde University College, Norway
Vittorio Maniezzo	Università di Bologna, Italy
Catherine C. McGeoch	Amherst College, USA
Daniel Merkle	Universität Leipzig, Germany
Peter Merz	Universität Kaiserslautern, Germany
Martin Middendorf	Universität Leipzig, Germany
Pablo Moscato	University of Newcastle, Australia
Luis Paquete	University of Algarve, Portugal
Steven Prestwich	University College Cork, Ireland

Günther Raidl	Vienna University of Technology, Austria
Celso Ribeiro	Pontifícia Univ. Católica do Rio de Janeiro, Brazil
Andrea Roli	Università degli Studi G. D'Annunzio, Italy
Jonathan Rowe	University of Birmingham, UK
Ruben Ruiz	Valencia University of Technology, Spain
Michael Sampels	Université Libre de Bruxelles, Belgium
Andrea Schaerf	Università degli Studi di Udine, Italy
El-Ghazali Talbi	University of Lille, France
Pascal Van Hentenryck	Brown University, USA
Stefan Voss	University of Hamburg, Germany
Jean-Paul Watson	Sandia National Labs, USA
Ingo Wegener	Universität Dortmund, Germany
David Woodruff	University of California, Davis, USA
Mutsunori Yagiura	Nagoya University, Japan

Local Arrangements

Prasanna Balaprakash	Université Libre de Bruxelles, Belgium
Carlotta Piscopo	Université Libre de Bruxelles, Belgium

Additional Referees

Prasanna Balaprakash
Frank Hutter
Marco A. Montes de Oca

Sponsoring Institutions

COMP2SYS, Marie Curie Early Stage Training Site
http://iridia.ulb.ac.be/comp2sys

National Fund for Scientific Research, Belgium
http://www.fnrs.be

French Community of Belgium (through the research project ANTS)
http://www.cfwb.be

Lecture Notes in Computer Science

For information about Vols. 1–4569

please contact your bookseller or Springer

Vol. 4708: L. Kučera, A. Kučera (Eds.), Mathematical Foundations of Computer Science 2007. XVIII, 764 pages. 2007.

Vol. 4707: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part III. XXIV, 1205 pages. 2007.

Vol. 4706: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part II. XXIII, 1129 pages. 2007.

Vol. 4705: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part I. XLIV, 1169 pages. 2007.

Vol. 4703: L. Caires, V.T. Vasconcelos (Eds.), CONCUR 2007 – Concurrency Theory. XIII, 507 pages. 2007.

Vol. 4697: L. Choi, Y. Paek, S. Cho (Eds.), Advances in Computer Systems Architecture. XIII, 400 pages. 2007.

Vol. 4685: D.J. Veit, J. Altmann (Eds.), Grid Economics and Business Models. XII, 201 pages. 2007.

Vol. 4683: L. Kang, Y. Liu, S. Zeng (Eds.), Intelligence Computation and Applications. XVII, 663 pages. 2007.

Vol. 4682: D.-S. Huang, L. Heutte, M. Loog (Eds.), Advanced Intelligent Computing Theories and Applications. XXVII, 1373 pages. 2007. (Sublibrary LNAI).

Vol. 4681: D.-S. Huang, L. Heutte, M. Loog (Eds.), Advanced Intelligent Computing Theories and Applications. XXVI, 1379 pages. 2007.

Vol. 4679: A.L. Yuille, S.-C. Zhu, D. Cremers, Y. Wang (Eds.), Energy Minimization Methods in Computer Vision and Pattern Recognition. XII, 494 pages. 2007.

Vol. 4673: W.G. Kropatsch, M. Kampel, A. Hanbury (Eds.), Computer Analysis of Images and Patterns. XX, 1006 pages. 2007.

Vol. 4671: V. Malyshkin (Ed.), Parallel Computing Technologies. XIV, 635 pages. 2007.

Vol. 4660: S. Džeroski, J. Todorovski (Eds.), Computational Discovery of Scientific Knowledge. X, 327 pages. 2007. (Sublibrary LNAI).

Vol. 4657: C. Lambrinoudakis, G. Pernul, A M. Tjoa (Eds.), Trust and Privacy in Digital Business. XIII, 291 pages. 2007.

Vol. 4651: F. Azevedo, P. Barahona, F. Fages, F. Rossi (Eds.), Recent Advances in Constraints. VIII, 185 pages. 2007. (Sublibrary LNAI).

Vol. 4649: V. Diekert, M.V. Volkov, A. Voronkov (Eds.), Computer Science – Theory and Applications. XIII, 420 pages. 2007.

Vol. 4647: R. Martin, M. Sabin, J. Winkler (Eds.), Mathematics of Surfaces XII. IX, 509 pages. 2007.

Vol. 4645: R. Giancarlo, S. Hannenhalli (Eds.), Algorithms in Bioinformatics. XIII, 432 pages. 2007. (Sublibrary LNBI).

Vol. 4644: N. Azemard, L. Svensson (Eds.), Integrated Circuit and System Design. XIV, 583 pages. 2007.

Vol. 4643: M.-F. Sagot, M.E.M.T. Walter (Eds.), Advances in Bioinformatics and Computational Biology. XII, 177 pages. 2007. (Sublibrary LNBI).

Vol. 4642: S.-W. Lee, S.Z. Li (Eds.), Advances in Biometrics. XX, 1216 pages. 2007.

Vol. 4639: E. Csuhaj-Varj'u, Z. Ésik (Eds.), Fundamentals of Computation Theory. XIV, 508 pages. 2007.

Vol. 4638: T. Stützle, M. Birattari, H.H. Hoos (Eds.), Engineering Stochastic Local Search Algorithms. X, 223 pages. 2007.

Vol. 4637: C. Kruegel, R. Lippmann, A. Clark (Eds.), Recent Advances in Intrusion Detection. XII, 337 pages. 2007.

Vol. 4635: B. Kokinov, D.C. Richardson, T.R. Roth-Berghofer, L. Vieu (Eds.), Modeling and Using Context. XIV, 574 pages. 2007. (Sublibrary LNAI).

Vol. 4634: H.R. Nielson, G. Filé (Eds.), Static Analysis. XI, 469 pages. 2007.

Vol. 4633: M. Kamel, A. Campilho (Eds.), Image Analysis and Recognition. XII, 1312 pages. 2007.

Vol. 4632: R. Alhajj, H. Gao, X. Li, J. Li, O.R. Zaïane (Eds.), Advanced Data Mining and Applications. XV, 634 pages. 2007. (Sublibrary LNAI).

Vol. 4628: L.N. de Castro, F.J. Von Zuben, H. Knidel (Eds.), Artificial Immune Systems. XII, 438 pages. 2007.

Vol. 4627: M. Charikar, K. Jansen, O. Reingold, J.D.P. Rolim (Eds.), Approximation, Randomization, and Combinatorial Optimization. XII, 626 pages. 2007.

Vol. 4626: R.O. Weber, M.M. Richter (Eds.), Case-Based Reasoning Research and Development. XIII, 534 pages. 2007. (Sublibrary LNAI).

Vol. 4624: T. Mossakowski, U. Montanari, M. Haveraaen (Eds.), Algebra and Coalgebra in Computer Science. XI, 463 pages. 2007.

Vol. 4622: A. Menezes (Ed.), Advances in Cryptology - CRYPTO 2007. XIV, 631 pages. 2007.

Vol. 4619: F. Dehne, J.-R. Sack, N. Zeh (Eds.), Algorithms and Data Structures. XVI, 662 pages. 2007.

Vol. 4618: S.G. Akl, C.S. Calude, M.J. Dinneen, G. Rozenberg, H.T. Wareham (Eds.), Unconventional Computation. X, 243 pages. 2007.

Vol. 4617: V. Torra, Y. Narukawa, Y. Yoshida (Eds.), Modeling Decisions for Artificial Intelligence. XII, 502 pages. 2007. (Sublibrary LNAI).

Vol. 4616: A. Dress, Y. Xu, B. Zhu (Eds.), Combinatorial Optimization and Applications. XI, 390 pages. 2007.

Vol. 4615: R. de Lemos, C. Gacek, A. Romanovsky (Eds.), Architecting Dependable Systems IV. XIV, 435 pages. 2007.

Vol. 4613: F.P. Preparata, Q. Fang (Eds.), Frontiers in Algorithmics. XI, 348 pages. 2007.

Vol. 4612: I. Miguel, W. Ruml (Eds.), Abstraction, Reformulation, and Approximation. XI, 418 pages. 2007. (Sublibrary LNAI).

Vol. 4611: J. Indulska, J. Ma, L.T. Yang, T. Ungerer, J. Cao (Eds.), Ubiquitous Intelligence and Computing. XXIII, 1257 pages. 2007.

Vol. 4610: B. Xiao, L.T. Yang, J. Ma, C. Muller-Schloer, Y. Hua (Eds.), Autonomic and Trusted Computing. XVIII, 571 pages. 2007.

Vol. 4609: E. Ernst (Ed.), ECOOP 2007 – Object-Oriented Programming. XIII, 625 pages. 2007.

Vol. 4608: H.W. Schmidt, I. Crnkovic, G.T. Heineman, J.A. Stafford (Eds.), Component-Based Software Engineering. XII, 283 pages. 2007.

Vol. 4607: L. Baresi, P. Fraternali, G.-J. Houben (Eds.), Web Engineering. XVI, 576 pages. 2007.

Vol. 4606: A. Pras, M. van Sinderen (Eds.), Dependable and Adaptable Networks and Services. XIV, 149 pages. 2007.

Vol. 4605: D. Papadias, D. Zhang, G. Kollios (Eds.), Advances in Spatial and Temporal Databases. X, 479 pages. 2007.

Vol. 4604: U. Priss, S. Polovina, R. Hill (Eds.), Conceptual Structures: Knowledge Architectures for Smart Applications. XII, 514 pages. 2007. (Sublibrary LNAI).

Vol. 4603: F. Pfenning (Ed.), Automated Deduction – CADE-21. XII, 522 pages. 2007. (Sublibrary LNAI).

Vol. 4602: S. Barker, G.-J. Ahn (Eds.), Data and Applications Security XXI. X, 291 pages. 2007.

Vol. 4600: H. Comon-Lundh, C. Kirchner, H. Kirchner (Eds.), Rewriting, Computation and Proof. XVI, 273 pages. 2007.

Vol. 4599: S. Vassiliadis, M. Berekovic, T.D. Hämäläinen (Eds.), Embedded Computer Systems: Architectures, Modeling, and Simulation. XVIII, 466 pages. 2007.

Vol. 4598: G. Lin (Ed.), Computing and Combinatorics. XII, 570 pages. 2007.

Vol. 4597: P. Perner (Ed.), Advances in Data Mining. XI, 353 pages. 2007. (Sublibrary LNAI).

Vol. 4596: L. Arge, C. Cachin, T. Jurdziński, A. Tarlecki (Eds.), Automata, Languages and Programming. XVII, 953 pages. 2007.

Vol. 4595: D. Bošnački, S. Edelkamp (Eds.), Model Checking Software. X, 285 pages. 2007.

Vol. 4594: R. Bellazzi, A. Abu-Hanna, J. Hunter (Eds.), Artificial Intelligence in Medicine. XVI, 509 pages. 2007. (Sublibrary LNAI).

Vol. 4592: Z. Kedad, N. Lammari, E. Métais, F. Meziane, Y. Rezgui (Eds.), Natural Language Processing and Information Systems. XIV, 442 pages. 2007.

Vol. 4591: J. Davies, J. Gibbons (Eds.), Integrated Formal Methods. IX, 660 pages. 2007.

Vol. 4590: W. Damm, H. Hermanns (Eds.), Computer Aided Verification. XV, 562 pages. 2007.

Vol. 4589: J. Münch, P. Abrahamsson (Eds.), Product-Focused Software Process Improvement. XII, 414 pages. 2007.

Vol. 4588: T. Harju, J. Karhumäki, A. Lepistö (Eds.), Developments in Language Theory. XI, 423 pages. 2007.

Vol. 4587: R. Cooper, J. Kennedy (Eds.), Data Management. XIII, 259 pages. 2007.

Vol. 4586: J. Pieprzyk, H. Ghodosi, E. Dawson (Eds.), Information Security and Privacy. XIV, 476 pages. 2007.

Vol. 4585: M. Kryszkiewicz, J.F. Peters, H. Rybinski, A. Skowron (Eds.), Rough Sets and Intelligent Systems Paradigms. XIX, 836 pages. 2007. (Sublibrary LNAI).

Vol. 4584: N. Karssemeijer, B. Lelieveldt (Eds.), Information Processing in Medical Imaging. XX, 777 pages. 2007.

Vol. 4583: S.R. Della Rocca (Ed.), Typed Lambda Calculi and Applications. X, 397 pages. 2007.

Vol. 4582: J. Lopez, P. Samarati, J.L. Ferrer (Eds.), Public Key Infrastructure. XI, 375 pages. 2007.

Vol. 4581: A. Petrenko, M. Veanes, J. Tretmans, W. Grieskamp (Eds.), Testing of Software and Communicating Systems. XII, 379 pages. 2007.

Vol. 4580: B. Ma, K. Zhang (Eds.), Combinatorial Pattern Matching. XII, 366 pages. 2007.

Vol. 4579: B. M. Hämmerli, R. Sommer (Eds.), Detection of Intrusions and Malware, and Vulnerability Assessment. X, 251 pages. 2007.

Vol. 4578: F. Masulli, S. Mitra, G. Pasi (Eds.), Applications of Fuzzy Sets Theory. XVIII, 693 pages. 2007. (Sublibrary LNAI).

Vol. 4577: N. Sebe, Y. Liu, Y.-t. Zhuang, T.S. Huang (Eds.), Multimedia Content Analysis and Mining. XIII, 513 pages. 2007.

Vol. 4576: D. Leivant, R. de Queiroz (Eds.), Logic, Language, Information and Computation. X, 363 pages. 2007.

Vol. 4575: T. Takagi, T. Okamoto, E. Okamoto, T. Okamoto (Eds.), Pairing-Based Cryptography – Pairing 2007. XI, 408 pages. 2007.

Vol. 4574: J. Derrick, J. Vain (Eds.), Formal Techniques for Networked and Distributed Systems – FORTE 2007. XI, 375 pages. 2007.

Vol. 4573: M. Kauers, M. Kerber, R. Miner, W. Windsteiger (Eds.), Towards Mechanized Mathematical Assistants. XIII, 407 pages. 2007. (Sublibrary LNAI).

Vol. 4572: F. Stajano, C. Meadows, S. Capkun, T. Moore (Eds.), Security and Privacy in Ad-hoc and Sensor Networks. X, 247 pages. 2007.

Vol. 4571: P. Perner (Ed.), Machine Learning and Data Mining in Pattern Recognition. XIV, 913 pages. 2007. (Sublibrary LNAI).

Vol. 4570: H.G. Okuno, M. Ali (Eds.), New Trends in Applied Artificial Intelligence. XXI, 1194 pages. 2007. (Sublibrary LNAI).

¥452.00元

Table of Contents

Short Papers

The Importance of Being Careful

Arne Løkketangen

Molde University College, Molde, Norway
Arne.Lokketangen@himolde.no

Abstract. Metaheuristic (and other) methods are often implemented using standardized parameter values. This is most often seen when comparing a favorite method with competing methods, but also when inexperienced researchers implement a method for the first time. Often the (hidden) correlations between the search method components and parameters are neglected or ignored, using only standardized templates. This paper looks at some of these pitfalls or hidden correlations, using the mechanisms of tabu search (TS) as examples. The points discussed are illustrated by examples from the authors experience.

1 Introduction

When designing and implementing heuristics for discrete optimization problems, there are many choices to be made. These include search paradigms, search mechanisms, search parameters, test sets, etc. Often these choices are treated as independent of each other, even though most researchers acknowledge some interdependence between the different search mechanisms and associated parameter settings. Our experience is that the search mechanisms and parameters often interact, and at times in unforeseen ways. This is a difficult problem to disentangle, and often even to discover.

This paper attempts to highlight some of the interactions experienced by the author, while implementing the Tabu Search (TS) metaheuristic [3]. Others will have similar experiences. The reader should at least learn that caution is required when designing heuristics, and testing should be carried out to verify the hypothesis underlying each mechanism and parameter choice.

One early example is the publicity in the early days of TS by people claiming that the best value for Tabu Tenure (TT) was 7. This was mainly due to the (somewhat erroneous) linkage to the article *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information* by Miller [1], creating the impression that TS was governed by some of the same rules as human cognitive abilities. The claim of 7 being the best TT has long since been abandoned, noticing that good TT values are also correlated to instance size and neighborhood structure.

Many articles on how to design and test heuristics have been published, indicating that these processes are far from being well-defined. Two much-cited works are *How not to do it* by Gent and Walsh [2] and *Testing Heuristics: We have it all wrong* by Hooker [4]. These articles contains much good advice, and

T. Stützle, M. Birattari, and H.H. Hoos (Eds.): SLS 2007, LNCS 4638, pp. 1–15, 2007.

the interested reader should acquaint themselves with them. From the article by Gent and Walsh, the following advice stands out: Make fast, repeatable code. That is, make the code fast, so that more tests can be done in the same amount of cpu-time, and make the code repeatable, as there is very often need to run tests anew, due to some error, or referee comment, or other. One of Hooker's main points is that we should forget much of the competition aspect, as this leads us to waste time on low-level engineering instead of inventing or discovering new mechanisms. The engineering part is hardly research, but is often required by referees who will only accept articles presenting new *best* results.

One example of unnecessary slow code too often seen is when the TT mechanisms are implemented as a (circular) list. This is cumbersome, has a time complexity of $|TT|$ and is thus very time consuming, especially as the TT can be implemented in constant time with very little overhead.

The examples used for illustrations in the following sections are only explained in sufficient detail for understanding the arguments. To get more detail and background, the authors are referred to the original papers.

This paper is organized as follows. In Section 2 we look at values for the *Tabu Tenure*, while the usefulness of *Aspiration Criteria* is illustrated in Section 3. The question on whether to search in infeasible space or not is addressed in Section 4. Section 5 treats the *Move Selection*, Section 6 discusses *Learning and Forgetting*, and Section 7 discusses some tradeoff issues when implementing search methods. Finally the conclusions are summarized in Section 8.

2 The Tabu Tenure

The value for the Tabu Tenure is often given little consideration, and often it seems like it is taken out of thin air, or by reference to another article. (Occasionally one might even see the 7 +/- 2 argument). There is some controversy as to wether some randomization in the value of TT should be used, or not, with current consensus leaning towards some randomization. There are clear indicators, however, that the actual value of TT required is linked to the other mechanisms used, especially move evaluation, diversification and learning.

It is often stated that the purpose of the tabu status is in stopping the reversal of recent moves. This is clearly only part of the picture. An important effect of the tabu status is to block off most of the search space. As an example, in a bit-flip neighborhood, and a TT of 20, then the search space is reduced by a factor of 2^{20}. It is easy to construct tabu-criteria that are very powerful. As a different example, consider an edge-exchange neighborhood, often employed in graph problems. If e.g. the instance size is $|N|$ (same as the number of nodes) and the total number of edges is $|N|^2$, then both of the following tabu criteria might seem natural if both edges involved in a move becomes tabu:

1. A move is tabu if both edges of the move are tabu.
2. A move is tabu if either edge of the move is tabu.

The first criterion is very weak, while the second is very strong. These considerations are very often not present when research is presented in articles, but has

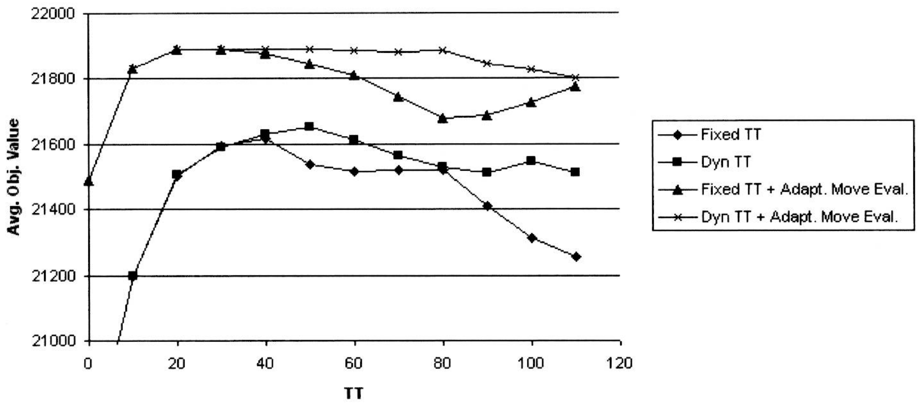

Fig. 1. Tabu Tenure

a large effect on the required values of the search parameters and the resulting search trajectories.

2.1 Varying Length TT

This subsection gives an example of when a dynamic TT is better than a fixed TT. The example is taken from Hvattum et al. [5]. The problem specification is given below to give the necessary background. The *Boolean Optimization Problem* can be regarded as a *Satisfiability Problem* with a profit on the variables. This profit is to be maximized.

$$max \qquad z = \sum_{j \in \mathbf{N}} c_j x_j \tag{1}$$

$s.t.$

$$\sum_{j \in \mathbf{N}} a_{ij} x_j \geq 1, \qquad i \in \mathbf{M} \tag{2}$$

$$x_j \in \{0, 1\}, \qquad j \in \mathbf{N} \tag{3}$$

The TS metaheuristic developed in [5] to solve this problem uses a simple 1-bit flip neighborhood. In Figure 1 is shown the effect, using a single instance, on the search for various values of the TT, and keeping it either fixed or varying. For the varying TT, it was set to be a random number between 10 and the number on the x-axis. With a fixed move evaluation function, the results are as shown in the bottom pair of curves in the figure. As can be seen, having a dynamic TT is clearly beneficial, as better results are obtained over a larger range of values. The same conclusion is supported by the top two curves, giving the corresponding results using an adaptive move evaluation function (see 2.2). (Similar results were observed for the other test-cases).

2.2 Interaction of TT with Move Evaluation

In Figure 1 is also shown how the form of the move evaluation function influences both the TT and the overall search results. The search in [5] was designed to be able to search both in feasible and infeasible space. This is done by modifying the move evaluation function F_M with a component reflecting the resulting level of infeasibility. $F_M = \Delta V + w \cdot \Delta z$. The range of change in the objective function value component ΔV is scaled to ± 1, and the infeasibility component is the number of violated constraints multiplied by an adaptive weight, w. The two components are combined so as to give a balanced view to maintaining primal feasibility and a good objective function value. The emphasis between the two components is changed dynamically to keep the search around the feasibility boundary.

The value of w, the adaptive component, is initially set to 1. It is adjusted after each move as follows:

- If the current solution is feasible: $w = w + \Delta w_{inc}$
- If the current solution is not feasible, and $w > 1$: $w = w - \Delta w_{dec}$

As is illustrated in Figure 1, the effect of using an adaptive move evaluation function, capable of guiding the search into, and out of, infeasible space, has a large impact on the acceptable values for TT. The most important aspect of this is that the TT can be shorter, and good results are produced over a larger range of TT, making the search much more robust with respect to this parameter. (For the actual values used for Δw_{inc} and Δw_{dec}, see Section 4).

2.3 Localized Tabu Tenure

This subsection deals with the case when the global constraints can be regarded as weak. A typical case of this is described in *Interactive Planning for Sustainable Forest Harvesting* by Hasle et al. [6]. The object here is to device a sustainable plan for the treatment of each of a set of connected forest stands, or areas, in a 200 year perspective. Each stand has its own type of wood, age profile, soil quality, inclination, etc. giving rise to different growth rates and suitability for different types of trees. In Figure 2 is shown a typical area of forest, with around 700 stands. A treatment can typically be *cut* or *leave*, but there are also numerous other treatments like *thinning* and *gathering seeds.* There are global constraints of the type requiring yearly harvesting to be approximately equal each year (or follow a given profile). There are also neighboring constraints, like the *2-meter-constraint* that requires all neighboring stands to be at least 2 meters tall before harvesting of a given stand. A move is the changing of a treatment for a given stand, and the tabu criterion is not to make a move in a recently treated stand.

Preliminary testing of a TS heuristic for this problem showed that the best TT was nearly equal to the number of stands in the instance. This implies that most stands will have a tabu status at any given time, leaving very little room for the local search.

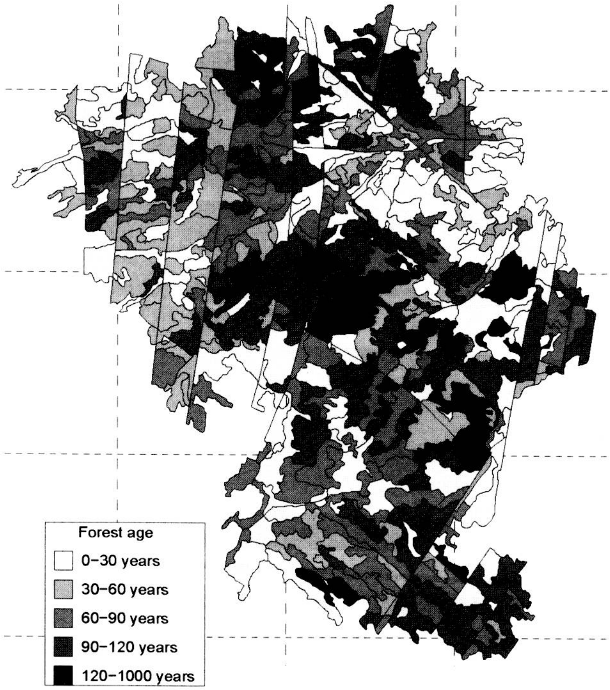

Fig. 2. Example of forest area and its subdivision into stands

The reason for this behavior is the weak nature of the global constraints, like balancing the harvesting over the years. Whenever the treatment plan for a given stand has been changed, the search moves to distant (in the geographical sense) stands. As there are many distant stands, all (or most) of them need to be tabu for the neighbor constraints to take effect, resulting in the extremely long required TT.

To remedy this, a device that might be called *Localized Tabu Tenure* can be employed. In this case the TT of a given stand is only counted for moves in the *vicinity* of the stand, where vicinity can be one or two stands distant. In this way, the TT will have a more natural length, and the neighboring constraints dealt with in the usual way.

3 Aspiration Criteria – Why Not Ignore It?

The inclusion of *Aspiration Criteria* (AC) in a TS is of crucial importance. The reason for this is that the basic TS mechanism of keeping certain attributes of the solution tabu for a certain time (i.e. iterations) often is too powerful, and blocks off attractive solutions. The mechanism called AS is designed to overcome this, by releasing the tabu status from moves leading to solutions having some predefined feature. The most common AC is to allow moves that leads to a new,