Marcus Randall
Hussein A. Abbass
Janet Wiles (Eds.)

LNAI 4828

Progress in Artificial Life

Third Australian Conference, ACAL 2007
Gold Coast, Australia, December 2007
Proceedings

Springer

Marcus Randall Hussein A. Abbass
Janet Wiles (Eds.)

Progress in Artificial Life

Third Australian Conference, ACAL 2007
Gold Coast, Australia, December 4-6, 2007
Proceedings

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Marcus Randall
Bond University
School of Information Technology
Robina, QLD 4229, Australia
E-mail: mrandall@bond.edu.au

Hussein A. Abbass
University of New South Wales
Australian Defence Force Academy
School of Information Technology & Electrical Engineering
Canberra, ACT 2600, Australia
E-mail: h.abbass@adfa.edu.au

Janet Wiles
The University of Queensland
School of Information Technology & Electrical Engineering
Division of Complex & Intelligent Systems
Brisbane, QLD 4072, Australia
E-mail: janetw@itee.uq.edu.au

Library of Congress Control Number: 2007939513

CR Subject Classification (1998): I.2, J.3, F.1.1-2, G.2, H.5, I.5, J.4, J.6

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN	0302-9743
ISBN-10	3-540-76930-7 Springer Berlin Heidelberg New York
ISBN-13	978-3-540-76930-9 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: 12195459 06/3180 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 4828
Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

The field of artificial life (Alife) is a rapidly emerging area that draws on expertise from computer science, biology, psychology, to name a few. In essence it is the study of systems related to life, its processes and evolution. These systems commonly use computer model simulations. The past decade has seen an increasing stream of scientific articles devoted to the exploration of Alife.

The Australian Conference on Artificial Life (ACAL) series is a testament to the above. It is a biannual event that originated in 2001 as the "Inaugral Workshop on Artificial Life" as part of the 14^{th} Joint Conference on Artificial Intelligence. ACAL 2007 received 70 quality submissions of which 34 were accepted for oral presentation in the conference. Each paper was peer reviewed by two or three members of the Program Committee. Apart from Australian researchers, the conference attracted participants from a number of countries across Europe, America, Asia-Pacific and Africa.

ACAL 2007 was fortunate to have four distinguished speakers in Alife to address the conference. They were David Abramson (Monash University), Kenneth A. De Jong (George Mason University), K.C. Tan (National University of Singapore) and Rodney Walker (Queensland University of Technology).

The organizers wish to thank a number of people and institutions for their support of this event and publication. Importantly we would like to acknowledge the effort and contributions of the Program Committee members and advisory board. Our sponsors were: The Australian Computer Society, the ARC Complex Open Systems Research Network, Bond University, The University of New South Wales (Australian Defence Force Academy), University of Canberra, Australian National University and the Gold Coast City Council. Their financial and in-kind support ensured the costs were minimized for attendees. Finally, the editors must pay tribute to the team at Springer.

We hope to repeat the success of ACAL 2007 with ACAL 2009. The venue of this event will be announced in 2008.

December 2007

Marcus Randall
Hussein A. Abbass
Janet Wiles

Organization

ACAL 2007 was organized by the School of Information Technology, Bond University in association with the University of New South Wales (Australian Defence Force Academy) and the University of Queensland.

Chairs

General Chair	Marcus Randall (Bond University, Australia)
Co-chairs	Hussein A. Abbass (University of New South Wales, Australia)
	Janet Wiles (University of Queensland, Australia)

Program Committee

Mark Bedau (Reed College, USA)
Alan Blair (UNSW, Australia)
Eric Bonabeau (Icosystem, USA)
Juergen Branke (University of Karlsruhe, Germany)
Angelo Cangelosi (Plymouth University, UK)
Stephan Chalup (Newcastle University, Australia)
Tan Kay Chen (NUS, Singapore)
Vic Ciesielski (RMIT, Australia)
Oscar Cordon (University of Granada, Spain)
David Cornforth (UNSW@ADFA, Australia)
Marco Dorigo (ULB, Belgium)
Alan Dorin (Monash University, Australia)
Daryl Essam (UNSW@ADFA, Australia)
David Green (Monash University, Australia)
Tim Hendtlass (Swinburne University, Australia)
Takashi Ikegami (Tokyo University, Japan)
Christian Jacob (Calgary University, Canada)
Ray Jarvis (Monash University, Australia)
Graham Kendall (Nottingham University, UK)
Kevin Korb (Monash University, Australia)
Xiaodong Li (RMIT, Australia)
Frederic Maire (QUT, Australia)
Bob McKay (Seoul National University, Korea)
James Montgomery (Swinburne University, Australia)
Chrystopher Nehaniv (University of Hertfordshire, UK)
David Newth (CSIRO, Australia)
Stefano Nolfi (CNR-ISTC, Italy)
Marcus Randall (Bond University, Australia)
Alex Ryan (DSTO, Australia)

Russell Standish (UNSW, Australia)
Charles Taylor (UCLA, USA)
Jason Teo (Universiti Malaysia Sabah, Malaysia)
Athanasios Vasilakos (University of Western Macedonia, Greece)
Peter Wills (University of Auckland, NZ)
Janet Wiles (University of Queensland, Australia)

Additional Reviewers

Lucas Hope (Monash University)
Antony Iorio (RMIT, Australia)
Irene Moser (Swinburne University, Australia)
Andy Song (RMIT, Australia)
Gayan Wijesinghe (RMIT, Australia)
Owen Woodberry (Monash University)
Qinying Xu (RMIT, Australia)

International Advisory Committee

Mark Bedau (Reed College, USA)
Eric Bonabeau (Icosystem, USA)
David Fogel (Natural Selection, USA)
Peter Stadler (Leipzig, Germany)
Masanori Sugisaka (Oita, Japan)

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 4828: M. Randall, H.A. Abbass, J. Wiles (Eds.), Progress in Artificial Life. XII, 402 pages. 2007.

Vol. 4827: A. Gelbukh, Á.F. Kuri Morales (Eds.), MICAI 2007: Advances in Artificial Intelligence. XXIV, 1234 pages. 2007.

Vol. 4798: Z. Zhang, J.H. Siekmann (Eds.), Knowledge Science and Engineering and Management. XVI, 669 pages. 2007.

Vol. 4795: F. Schilder, G. Katz, J. Pustejovsky (Eds.), Annotating, Extracting and Reasoning about Time and Events. VII, 141 pages. 2007.

Vol. 4790: N. Dershowitz, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 562 pages. 2007.

Vol. 4788: D. Borrajo, L. Castillo, J.M. Corchado (Eds.), Current Topics in Artificial Intelligence. XI, 280 pages. 2007.

Vol. 4775: A. Esposito, M. Faundez-Zanuy, E. Keller, M. Marinaro (Eds.), Verbal and Nonverbal Communication Behaviours. XII, 325 pages. 2007.

Vol. 4772: H. Prade, V.S. Subrahmanian (Eds.), Scalable Uncertainty Management. X, 277 pages. 2007.

Vol. 4766: N. Maudet, S. Parsons, I. Rahwan (Eds.), Argumentation in Multi-Agent Systems. XII, 211 pages. 2007.

Vol. 4755: V. Corruble, M. Takeda, E. Suzuki (Eds.), Discovery Science. XI, 298 pages. 2007.

Vol. 4754: M. Hutter, R.A. Servedio, E. Takimoto (Eds.), Algorithmic Learning Theory. XI, 403 pages. 2007.

Vol. 4737: B. Berendt, A. Hotho, D. Mladenic, G. Semeraro (Eds.), From Web to Social Web: Discovering and Deploying User and Content Profiles. XI, 161 pages. 2007.

Vol. 4733: R. Basili, M.T. Pazienza (Eds.), AI*IA 2007: Artificial Intelligence and Human-Oriented Computing. XVII, 858 pages. 2007.

Vol. 4724: K. Mellouli (Ed.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty. XV, 914 pages. 2007.

Vol. 4722: C. Pelachaud, J.-C. Martin, E. André, G. Chollet, K. Karpouzis, D. Pelé (Eds.), Intelligent Virtual Agents. XV, 425 pages. 2007.

Vol. 4720: B. Konev, F. Wolter (Eds.), Frontiers of Combining Systems. X, 283 pages. 2007.

Vol. 4702: J.N. Kok, J. Koronacki, R. Lopez de Mantaras, S. Matwin, D. Mladenič, A. Skowron (Eds.), Knowledge Discovery in Databases: PKDD 2007. XXIV, 640 pages. 2007.

Vol. 4701: J.N. Kok, J. Koronacki, R. Lopez de Mantaras, S. Matwin, D. Mladenič, A. Skowron (Eds.), Machine Learning: ECML 2007. XXII, 809 pages. 2007.

Vol. 4696: H.-D. Burkhard, G. Lindemann, R. Verbrugge, L.Z. Varga (Eds.), Multi-Agent Systems and Applications V. XIII, 350 pages. 2007.

Vol. 4694: B. Apolloni, R.J. Howlett, L. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. XXIX, 1126 pages. 2007.

Vol. 4693: B. Apolloni, R.J. Howlett, L. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. XXXII, 1380 pages. 2007.

Vol. 4692: B. Apolloni, R.J. Howlett, L. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LV, 882 pages. 2007.

Vol. 4687: P. Petta, J.P. Müller, M. Klusch, M. Georgeff (Eds.), Multiagent System Technologies. X, 207 pages. 2007.

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

Vol. 4676: M. Klusch, K.V. Hindriks, M.P. Papazoglou, L. Sterling (Eds.), Cooperative Information Agents XI. XI, 361 pages. 2007.

Vol. 4667: J. Hertzberg, M. Beetz, R. Englert (Eds.), KI 2007: Advances in Artificial Intelligence. IX, 516 pages. 2007.

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

Vol. 4659: V. Mařík, V. Vyatkin, A.W. Colombo (Eds.), Holonic and Multi-Agent Systems for Manufacturing. VIII, 456 pages. 2007.

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

Vol. 4648: F. Almeida e Costa, L.M. Rocha, E. Costa, I. Harvey, A. Coutinho (Eds.), Advances in Artificial Life. XVIII, 1215 pages. 2007.

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

Vol. 4629: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XVII, 663 pages. 2007.

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

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

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

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

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

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

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

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

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

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

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

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

Vol. 4565: D.D. Schmorrow, L.M. Reeves (Eds.), Foundations of Augmented Cognition. XIX, 450 pages. 2007.

Vol. 4562: D. Harris (Ed.), Engineering Psychology and Cognitive Ergonomics. XXIII, 879 pages. 2007.

Vol. 4548: N. Olivetti (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. X, 245 pages. 2007.

Vol. 4539: N.H. Bshouty, C. Gentile (Eds.), Learning Theory. XII, 634 pages. 2007.

Vol. 4529: P. Melin, O. Castillo, L.T. Aguilar, J. Kacprzyk, W. Pedrycz (Eds.), Foundations of Fuzzy Logic and Soft Computing. XIX, 830 pages. 2007.

Vol. 4520: M.V. Butz, O. Sigaud, G. Pezzulo, G. Baldassarre (Eds.), Anticipatory Behavior in Adaptive Learning Systems. X, 379 pages. 2007.

Vol. 4511: C. Conati, K. McCoy, G. Paliouras (Eds.), User Modeling 2007. XVI, 487 pages. 2007.

Vol. 4509: Z. Kobti, D. Wu (Eds.), Advances in Artificial Intelligence. XII, 552 pages. 2007.

Vol. 4496: N.T. Nguyen, A. Grzech, R.J. Howlett, L.C. Jain (Eds.), Agent and Multi-Agent Systems: Technologies and Applications. XXI, 1046 pages. 2007.

Vol. 4483: C. Baral, G. Brewka, J. Schlipf (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 327 pages. 2007.

Vol. 4482: A. An, J. Stefanowski, S. Ramanna, C.J. Butz, W. Pedrycz, G. Wang (Eds.), Rough Sets, Fuzzy Sets, Data Mining and Granular Computing. XIV, 585 pages. 2007.

Vol. 4481: J. Yao, P. Lingras, W.-Z. Wu, M. Szczuka, N.J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology. XIV, 576 pages. 2007.

Vol. 4476: V. Gorodetsky, C. Zhang, V.A. Skormin, L. Cao (Eds.), Autonomous Intelligent Systems: Multi-Agents and Data Mining. XIII, 323 pages. 2007.

Vol. 4460: S. Aguzzoli, A. Ciabattoni, B. Gerla, C. Manara, V. Marra (Eds.), Algebraic and Proof-theoretic Aspects of Non-classical Logics. VIII, 309 pages. 2007.

Vol. 4457: G.M.P. O'Hare, A. Ricci, M.J. O'Grady, O. Dikenelli (Eds.), Engineering Societies in the Agents World VII. XI, 401 pages. 2007.

Vol. 4456: Y. Wang, Y.-m. Cheung, H. Liu (Eds.), Computational Intelligence and Security. XXIII, 1118 pages. 2007.

Vol. 4455: S. Muggleton, R. Otero, A. Tamaddoni-Nezhad (Eds.), Inductive Logic Programming. XII, 456 pages. 2007.

Vol. 4452: M. Fasli, O. Shehory (Eds.), Agent-Mediated Electronic Commerce. VIII, 249 pages. 2007.

Vol. 4451: T.S. Huang, A. Nijholt, M. Pantic, A. Pentland (Eds.), Artifical Intelligence for Human Computing. XVI, 359 pages. 2007.

Vol. 4442: L. Antunes, K. Takadama (Eds.), Multi-Agent-Based Simulation VII. X, 189 pages. 2007.

Vol. 4441: C. Müller (Ed.), Speaker Classification II. X, 309 pages. 2007.

Vol. 4438: L. Maicher, A. Sigel, L.M. Garshol (Eds.), Leveraging the Semantics of Topic Maps. X, 257 pages. 2007.

Vol. 4434: G. Lakemeyer, E. Sklar, D.G. Sorrenti, T. Takahashi (Eds.), RoboCup 2006: Robot Soccer World Cup X. XIII, 566 pages. 2007.

Vol. 4429: R. Lu, J.H. Siekmann, C. Ullrich (Eds.), Cognitive Systems. X, 161 pages. 2007.

Vol. 4428: S. Edelkamp, A. Lomuscio (Eds.), Model Checking and Artificial Intelligence. IX, 185 pages. 2007.

Vol. 4426: Z.-H. Zhou, H. Li, Q. Yang (Eds.), Advances in Knowledge Discovery and Data Mining. XXV, 1161 pages. 2007.

Vol. 4411: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 249 pages. 2007.

Vol. 4410: A. Branco (Ed.), Anaphora: Analysis, Algorithms and Applications. X, 191 pages. 2007.

Vol. 4399: T. Kovacs, X. Llorà, K. Takadama, P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. XII, 345 pages. 2007.

Vol. 4390: S.O. Kuznetsov, S. Schmidt (Eds.), Formal Concept Analysis. X, 329 pages. 2007.

Vol. 4389: D. Weyns, H. Van Dyke Parunak, F. Michel (Eds.), Environments for Multi-Agent Systems III. X, 273 pages. 2007.

Vol. 4386: P. Noriega, J. Vázquez-Salceda, G. Boella, O. Boissier, V. Dignum, N. Fornara, E. Matson (Eds.), Coordination, Organizations, Institutions, and Norms in Agent Systems II. XI, 373 pages. 2007.

Table of Contents

Biological Systems I

Networks

Heuristics II

Complex Systems II

Heuristics III

Biological Systems II

Alternative Solution Representations for the Job Shop Scheduling Problem in Ant Colony Optimisation

James Montgomery

Complex Intelligent Systems Laboratory
Centre for Information Technology Research
Faculty of Information & Communication Technologies
Swinburne University of Technology
Melbourne, Australia
jmontgomery@ict.swin.edu.au

Abstract. Ant colony optimisation (ACO), a constructive metaheuristic inspired by the foraging behaviour of ants, has frequently been applied to shop scheduling problems such as the job shop, in which a collection of operations (grouped into jobs) must be scheduled for processing on different machines. In typical ACO applications solutions are generated by constructing a permutation of the operations, from which a deterministic algorithm can generate the actual schedule. An alternative approach is to assign each machine one of a number of alternative dispatching rules to determine its individual processing order. This representation creates a substantially smaller search space biased towards good solutions. A previous study compared the two alternatives applied to a complex real-world instance and found that the new approach produced better solutions more quickly than the original. This paper considers its application to a wider set of standard benchmark job shop instances. More detailed analysis of the resultant search space reveals that, while it focuses on a smaller region of good solutions, it also excludes the optimal solution. Nevertheless, comparison of the performance of ACO algorithms using the different solution representations shows that, using this solution space, ACO can find better solutions than with the typical representation. Hence, it may offer a promising alternative for quickly generating good solutions to seed a local search procedure which can take those solutions to optimality.

Keywords: Ant colony optimisation, job shop scheduling, solution representation.

1 Introduction

Ant colony optimisation (ACO) is a constructive metaheuristic, inspired by the foraging behaviour of ant colonies, that produces a number of solutions over successive iterations of solution construction. During each iteration, a number of artificial ants build solutions by probabilistically selecting from problem-specific

M. Randall, H.A. Abbass, and J. Wiles (Eds.): ACAL 2007, LNAI 4828, pp. 1–12, 2007.

solution components, influenced by a parameterised model of solutions (called a pheromone model in reference to ant trail pheromones). The parameters of this model are updated at the end of each iteration using the solutions produced so that, over time, the algorithm learns which solution components should be combined to produce the best solutions. When adapting ACO to suit a problem an algorithm designer must first decide how solutions are to be represented and built (i.e., what base *components* are to be combined to form solutions) and then what characteristics of the chosen representation are to be modelled.

Shop scheduling problems consist of a number of jobs, made up of a set of operations, each of which must be scheduled for processing on one of a number of machines. Precedence constraints are imposed on the operations of each job. The majority of ACO algorithms for these problems represent solutions as permutations of the operations to be scheduled (operations are the base components of solutions), which determines the relative order of operations that require the same machine (see, e.g., [1,2,3,4]). A deterministic algorithm can then produce the best possible schedule given the precedence constraints established by the permutation. This approach is more generally referred to as the *list scheduler algorithm* [2].

An alternative approach is to assign different heuristics to each machine which determine the relative processing order of operations, thereby searching the reduced space of schedules that can be produced by different combinations of the heuristics. Building solutions in this manner may offer an advantage by concentrating the search on heuristically good solutions. A previous study compared these two solution representations in ACO algorithms for a real-world job shop scheduling problem (JSP) with staggered release and due dates modelled using fuzzy sets [5]. Applied to that single real-world instance the alternative approach performed extremely well, finding better solutions than the list scheduler ACO in considerably less time. An open question was whether the same relative performance would be observed on other, benchmark JSP instances.

This paper examines, in greater detail than in [5], the search space produced by the alternative solution representation when applied to a number of commonly used benchmark JSP instances (Section 4). An empirical comparison is subsequently made of ACO algorithms using the typical and alternative solution construction approaches (Sections 5–6). Section 7 describes the implications of the results for the future application of ACO to such problems. A formal description of the JSP and further details of the typical solution construction approach are given first.

2 Job Shop Scheduling

The JSP examined in this study is of the $n \times m$ form, with a set of n jobs $J_1, \ldots, J_n$ and m machines $M_1, \ldots, M_m$. Each job consists of a predetermined sequence of m operations, each of which requires one of the m machines. Only one operation from a job may be processed at any given time, only one operation may use a machine at any given time and operations may not be pre-empted.

Table 1. JSP instances used in this study

Instance	Best known	n	m
abz5	1234	10	10
abz6	943	10	10
abz7	656	20	15
abz8	669	20	15
abz9	679	20	15
ft10	930	10	10
ft20	1165	20	5
la21	1046	15	10
la24	935	15	10
la25	977	15	10
la27	1235	20	10
la29	1152	20	10
la38	1196	15	15
la40	1222	15	15
orb08	899	10	10
orb09	934	10	10

The objective is to schedule operations for processing on machines such that the total time to complete all jobs, the *makespan*, is minimised. The makespan of a solution s is denoted $C_{max}(s)$.

Table 1 describes the instances used in this study to compare the alternative solution representations. They are commonly used benchmarks in the ACO and wider operations research literature and are all available from the OR-Library [6].

3 Typical Solution Construction for the JSP

To generate a solution to the JSP it is sufficient to determine the relative processing order of operations that require the same machine. A deterministic algorithm can then produce the best possible schedule given those constraints. Indeed, it is common in ACO applications for the JSP and other related scheduling problems to generate a permutation of the operations, which implicitly determines this relative order (e.g., [1,2,3,4,7]). These algorithms are restricted to creating permutations that respect the required processing order of operations within each job, which can consequently be called *feasible permutations*.

Different approaches to constructing solutions produce different search spaces. The space of feasible permutations of operations for a JSP is very large (a weak upper bound is $O(k!)$, where $k = n \cdot m$ is the number of operations) and is certainly much larger than the space of feasible schedules [8]. This space also has a slight bias towards good solutions, which can be exploited by some pheromone models and proves disastrous for others. Another notable feature of this search space is that while all solutions can be reached, solutions (schedules) are represented by differing numbers of permutations. These issues are discussed in some detail by Montgomery, Randall and Hendtlass [8,9].

4 Search Space Created by Dispatching Rules

An alternative approach to building solutions is to assign different *dispatching rules* (i.e., ordering heuristics) to each machine, which subsequently build the actual schedule. The search space then becomes the space of all possible combinations of rules assigned to machines, which is $O(|D|^m)$ where D is the set of rules and m the number of machines. Given a small number of dispatching rules this search space will correspond to a subset of the space of all feasible schedules. Further, given that dispatching rules are chosen with the aim of minimising the makespan or number of tardy jobs, this is probably the case even for large sets of rules. However, if the dispatching rules individually perform well it is expected that this reduced space largely consists of good quality schedules.

Clearly, such an approach is inappropriate for single machine scheduling problems or problems in which too few criteria are available to heuristically determine the processing order of competing operations, as in either situation the search space is reduced by too great an amount. It is, however, entirely appropriate for problems with multiple machines and various criteria upon which to judge competing operations. This study examines its application to a number of common benchmark JSPs using four dispatching rules. The remainder of this section examines whether, for these instances using these four rules, the approach is appropriate.

The four rules used in this study are Earliest Starting Time (EST), Shortest Processing Time (SPT), Longest Processing Time (LPT) and Longest Remaining Processing Time (LRPT). SPT and LPT relate to an individual operation's processing time while LRPT refers to the remaining processing time of a candidate operation's containing job. EST is perhaps the simplest heuristic, choosing the operation that can start the soonest, with ties broken randomly. Note that the three other rules are not followed blindly: the earliest available operation is always chosen except when there are two or more such operations, in which case the rule determines which is given preference.

For small instances and a set of four rules it is possible to completely enumerate the set of assignment solutions.[1] This was performed for the test instances with up to 200 operations to discover the distribution of the cost of schedules described. The distributions for the larger instances were estimated by sampling 4×10^6 randomly generated solutions. Note that as the EST rule breaks ties randomly, there is some degree of error in the lower and upper bounds presented, although it is likely the distributions described here are good approximations of the true distributions. Fig. 1 presents box-plots of the distributions discovered, expressed in terms of the relative percentage deviation (RPD) from the best known cost, defined as

[1] Although complete enumeration of the search space obviates the need for a metaheuristic, on any moderate-sized instance or as the number of rules grows it quickly becomes impractical.