Tutorials at PPSN 2018

Gisele Lobo Pappa[1(✉)], Michael T. M. Emmerich[2], Ana Bazzan[3], Will Browne[4], Kalyanmoy Deb[5], Carola Doerr[6], Marko Đurasević[7], Michael G. Epitropakis[8], Saemundur O. Haraldsson[9], Domagoj Jakobovic[7], Pascal Kerschke[10], Krzysztof Krawiec[11], Per Kristian Lehre[12], Xiaodong Li[13], Andrei Lissovoi[14], Pekka Malo[15], Luis Martí[16], Yi Mei[4], Juan J. Merelo[17], Julian F. Miller[18], Alberto Moraglio[19], Antonio J. Nebro[20], Su Nguyen[21], Gabriela Ochoa[9], Pietro Oliveto[14], Stjepan Picek[22], Nelishia Pillay[23], Mike Preuss[10], Marc Schoenauer[24], Roman Senkerik[25], Ankur Sinha[26], Ofer Shir[27,28], Dirk Sudholt[14], Darrell Whitley[29], Mark Wineberg[30], John Woodward[31], and Mengjie Zhang[4]

[1] Federal University of Minas Gerais, Belo Horizonte, Brazil
glpappa@dcc.ufmg.br
[2] Leiden University, Leiden, Netherlands
[3] Federal University of Rio Grande do Sul, Porto Alegre, Brazil
[4] Victoria University of Wellington, Wellington, New Zealand
[5] Michigan State University, East Lansing, USA
[6] Sorbonne University, Paris, France
[7] University of Zagreb, Zagreb, Croatia
[8] University of Patras, Patras, Greece
[9] University of Stirling, Stirling, UK
[10] University of Müenster, Müenster, Germany
[11] Poznan University of Technology, Poznan, Poland
[12] University of Birmingham, Birmingham, UK
[13] RMIT University, Bundoora, Australia
[14] University of Sheffield, Sheffield, UK
[15] Aalto University, Helsinki, Finland
[16] Universidade Federal Fluminense, Niterói, Brazil
[17] University of Granada, Granada, Spain
[18] University of York, York, UK
[19] University of Exeter, Exeter, UK
[20] University of Malaga, Malaga, Spain
[21] La Trobe University, Melbourne, Australia
[22] TU Delft, Delft, Netherlands
[23] University of Pretoria, Pretoria, South Africa
[24] Institute for Research in Computer Science and Control, Ile de France, France
[25] Tomas Bata University, Zlín, Czech Republic
[26] Indian Institute of Management Ahmedabad, Ahmedabad, India
[27] Tel-Hai College, Tel Hai, Israel
[28] Migal-Galilee Research Institute, Kiryat Shmona, Israel
[29] Colorado State University, Fort Collins, USA
[30] University of Guelph, Guelph, Canada
[31] Queen Mary University of London, London, UK

A. Auger et al. (Eds.): PPSN 2018, LNCS 11102, pp. 477–489, 2018.
https://doi.org/10.1007/978-3-319-99259-4_38

Abstract. PPSN 2018 features a total of 23 free tutorials covering a broad range of topics in evolutionary computation and related areas. From theory and methods to applications and computer implementations, and from introductory to advanced, the PPSN 2018 tutorial program offers participants the opportunity to learn more about both well-established and ongoing research in this field.

1 Welcome from the Tutorial Chairs

PPSN 2018 received a surprisingly large number of high-quality tutorial proposals, from which 23 tutorials were selected for presentation at the conference. All tutorials are scheduled on the first two days of the conference, which are exclusively reserved for tutorial presentations and workshops.

The PPSN 2018 tutorial program features a wide range of topics, from theory to applications and computer implementations. Two hands-on tutorials cover the implementation of evolutionary algorithms (EAs) in the cloud and in the jMetal framework. Other tutorials focus on applications in security (cryptology, anomaly and intrusion detection), software engineering (genetic improvement), and dynamic scheduling.

In addition to introductory tutorials on mathematical programming, two genetic programming (GP) variants, learning classifier systems, and multiagent systems, a number of advanced techniques also deserve attention. Adaptive parameter choices, deterministic search operator design in Gray Box Optimization, and automated algorithm design with hyper-heuristics are covered, as well as algorithms for bi-level and multi-modal optimization.

Finally, there is a group of tutorials oriented towards the modelling, analysis and visualization of search-space structures, fitness landscapes, and evolutionary algorithm processes and their dynamics. Three tutorials on the theoretical analysis of population-based EAs, parallel EAs, and GP complete the programme.

We invite all PPSN participants to explore the wide range of topics discussed in the selected tutorials, and wish them an enjoyable conference!

Gisele L. Pappa and Michael T. M. Emmerich
PPSN 2018 Tutorial Chairs.

2 Tutorial Abstracts

2.1 Adaptive Parameter Choices in Evolutionary Computation

Tutorial Speaker: Carola Doerr, *Sorbonne University (France).*

Tutorial Abstract: Evolutionary algorithms and other popular black-box optimization techniques are highly parametrized algorithms. To run these algorithms, we typically need to decide upon their population sizes, mutation strengths, crossover rates, selective pressure, etc. This parametrization allows to adjust the behavior of the algorithms to the problem at hand. The chosen

parameter values can have a decisive influence on performance. We thus need to select them with care.

Unfortunately, the identification of good parameter values still is one of the most challenging tasks in evolutionary computation. What complicates the parameter selection problem is the observation that different parameter values can be optimal in different stages of the optimization process. In the beginning of an optimization process, for example, one may want to allow for more exploration, while later on we may prefer a more focused search ("exploitation"). This observation calls for adaptive parameter choices, which automatically adjust the parameter values to the current state of the optimization process.

Adaptive parameter choices are today standard in continuous optimization. Quite surprisingly, however, this is not the case in discrete optimization, where they play only a rather marginal role. A paradigm change towards a more systematic use of non-static parameter choices is much needed. This tutorial aims to contribute to this goal, by providing an in-depth discussion of online parameter selection techniques. We survey both experimental and theoretical results, which demonstrate the unexploited potential of non-static parameter choices.

2.2 Applications of Genetic Programming in Dynamic Scheduling

Tutorial Speakers: Domagoj Jakobovic and Marko Đuraseviæ, *University of Zagreb, Croatia*, Yi Mei and Mengjie Zhang, *Victoria University of Wellington (New Zealand)* and Su Nguyen, *La Trobe University (Australia).*

Tutorial Abstract: Scheduling problems are encountered in many real-world situations and scenarios. In real world, the problem is often dynamic, and unpredicted new jobs arrive in real time. To solve scheduling problems under dynamic conditions various problem-specific heuristics, called dispatching rules, have been designed. However, manually designing such heuristics is a difficult and lengthy process. Therefore, a great deal of research is focused on automatically designing new scheduling heuristics. Genetic programming is usually the method of choice for generating new dispatching rules, since in numerous occasions it generated good dispatching rules for various difficult scheduling environments. The tutorial will cover recent developments in the automatic generation of dispatching rules, as well as outline several new research directions in this field, such as multi-objective heuristic generation, application of ensemble learning methods, construction of surrogate models, etc. The tutorial will help interested researchers to acquire an overview of this emerging and interesting research area and to understand the key ideas and challenges for future studies.

2.3 A Small World Hidden in Evolutionary Computation Techniques

Tutorial Speaker: Roman Senkerik, *Tomas Bata University (Czech Republic).*

Tutorial Abstract: This tutorial represents an insight into an attractive open research task, which is a novel method for visualizing the dynamics of evolutionary and swarm-based algorithms in the form of networks. The idea is based on the similarity in interactions between individuals in the metaheuristics algorithms and for example, users of social networks, linking between web pages, etc. The population is visualized as an evolving complex network that exhibits non-trivial features. The features like clustering, centralities, communities, and many more, offer a clear description of the population under evaluation. This tutorial shows the differences between the types of complex networks used, variations in building complex networks to capture the population dynamics of evolutionary algorithms or communication inside swarm-based algorithms, investigation on the time development of the network. It also shows several successful utilization of complex networks attributes for the performance improvements through the adaptive population as well as parameter/strategy control, further the possibility of controlling the evolution through complex network features.

2.4 Bio-inspired Approaches to Anomaly and Intrusion Detection

Tutorial Speakers: Luis Martí, *Universidade Federal Fluminense (Brazil)* and Marc Schoenauer, *Institute for Research in Computer Science and Control (France).*

Tutorial Abstract: Intrusion detection systems (IDSs) have gained a substantial attention because of its high-impact safety and security applications. Two main approaches are used when building those systems: (i) misuse-based and (ii) anomaly-based detection. While the former focuses on detecting attacks that follow a known pattern or signature, the latter is interested in building a model representing the system's nominal behavior while assuming all deviated activities to be anomalous or intrusions, and, therefore provide a more robust solution. Bio-inspired approaches have been proposed to address the problem of anomaly-based intrusion detection, with artificial immune systems (AISs) being the most recognizable approach of all. However, recent developments in the area of single and multi-criterion evolutionary computing, adversarial co-evolutionary modeling and simulation have served a foundation for novel and better performing bio-inspired IDS that have yielded competitive results.

This tutorial will present the anomaly detection topic, its peculiarities and revise the current state of the art on this topic, departing from classical machine learning approaches, presenting the current state-of-the-art methods and analyze how and why those methods have been shown to outperform many of the currently established approaches.

2.5 Cartesian Genetic Programming

Tutorial Speaker: Julian F. Miller, *University of York (UK).*

Tutorial Abstract: Cartesian Genetic Programming (CGP) is a well-known and respected form of Genetic Programming. It uses a very simple integer address-based genetic representation of a program in the form of a directed graph. In a number of studies, CGP has been shown to be comparatively efficient to other GP techniques. The classical form of CGP has undergone a number of developments which have made it more useful, efficient and flexible in various ways. These include self-modifying CGP (SMCGP), cyclic connections (recurrent-CGP), encoding artificial neural networks and automatically defined functions (modular CGP). SMCGP uses functions that cause the evolved programs to change themselves as a function of time. Recurrent-CGP allows evolution to create programs which contain cyclic, as well as acyclic, connections. CGP encoded artificial neural networks represent a powerful training method for neural networks. CGP has been applied successfully to a variety of real-world problems, such as digital circuit design, visual object recognition and classification.

2.6 Cloud-y Evolutionary Algorithms

Tutorial Speaker: J.J. Merelo, *University of Granada (Spain).*

Tutorial Abstract: This tutorial will describe how cloud computing is a new paradigm that changes the way applications are designed and deployed, and how it can be put to use in a scientific computing environment. It will be a practical tutorial with examples and tools that are used nowadays by companies and institutions. The examples used will be taken from evolutionary computation, although its application is widespread.

2.7 Computational Complexity Analysis of Genetic Programming

Tutorial Speaker: Pietro Oliveto and Andrei Lissovoi, *University of Sheffield (UK).*

Tutorial Abstract: Genetic Programming is an evolutionary computation paradigm that aims to evolve computer programs. Compared to the great number of successful applications of GP that have been reported, the theoretical understanding of its underlying working principles lags far behind. In particular, the identification of which classes of computer programs can be provably evolved efficiently via GP has progressed slowly compared to the understanding of the performance of traditional evolutionary algorithms (EAs) for function optimisation. The main reason for the slow progress is that the analysis of GP systems is considerably more involved due to the variable length of programs compared to the fixed solution representation used in EAs and because understanding candidate program quality over all possible inputs is unfeasible. Nevertheless, nowadays it is possible to analyse the time and space complexity of GP algorithms for evolving proper programs with input/output relationships where

the fitness of candidate solutions is evaluated by comparing their accuracy on input/output samples of a polynomially-sized training set (e.g., Boolean Functions). In this tutorial, we give an overview of the recent results outlining the techniques used and the challenges involved.

2.8 Evolutionary Algorithms and Hyper-Heuristics

Tutorial Speaker: Nelishia Pillay, *University of Pretoria (South Africa).*

Tutorial Abstract: Evolutionary algorithms have played a pivotal role in the advancement of hyper-heuristics. The aim of the tutorial is to firstly provide an introduction to evolutionary algorithm hyper-heuristics. The tutorial will examine each of the four categories of hyper-heuristics, namely, selection constructive, selection perturbative, generation constructive and generation perturbative, showing how evolutionary algorithms can be used for each type of hyper-heuristic. A case study will be presented for each type of hyper-heuristic. The EvoHyp library will be used to demonstrate the implementation of evolutionary algorithm hyper-heuristics for the case studies. Challenges in the implementation of evolutionary algorithm hyper-heuristics will be highlighted. An emerging research direction is using hyper-heuristics for the automated design of computational intelligence techniques. The tutorial will look at the synergistic relationship between evolutionary algorithms and hyper-heuristics in this area. The tutorial will end with a discussion session on future directions in evolutionary algorithms and hyper-heuristics.

2.9 Evolutionary Bilevel Optimization (EBO): An Emerging Area for Research and Application in EC

Tutorial Speakers: Kalyanmoy Deb, *Michigan State University (USA)*, Ankur Sinha, *Indian Institute of Management Ahmedabad (India)*, and Pekka Malo, *Aalto University (Finland).*

Tutorial Abstract: Many practical optimization problems are better posed as hierarchical optimization problems in which different optimization tasks are put into different levels. The simplest of these hierarchical problems is known as "Bilevel" optimization problems which contain two levels of optimization tasks in a nested manner. A solution at the upper level is considered feasible only if it is optimal to the corresponding lower level problem. These problems are too complex to be solved using classical optimization methods simply due to the "nestedness" of one optimization task into another. Evolutionary algorithms provide amenable ways to address such problems due to their flexibility and ability to handle constrained search spaces efficiently. In this tutorial, we will introduce principles of bilevel optimization for both single and multiple objectives, and

discuss the difficulties in solving such problems in general. A number of applications of evolutionary bilevel optimization (EBO) will also be highlighted. A recent review on EBO is available in Sinha et al. (2018).

Sinha, A., Malo, P., and Deb, K. (2018). A Review on Bilevel Optimization: From Classical to Evolutionary Approaches and Applications. IEEE Transactions on Evolutionary Computation, Vol 22, No. 2, pp. 276–295.

2.10 Evolutionary Computation and Machine Learning in Cryptology

Tutorial Speaker: Stjepan Picek, *TU Delft (Netherlands).*

Tutorial Abstract: Evolutionary Computation (EC) has been successfully applied to various real-world problems. One domain rich with difficult problems is cryptology. This tutorial starts with a brief introduction on cryptology intended for general audience. Next, we examine several topics from cryptology that are successfully tackled up to now with EC and discuss why those topics are suitable to apply EC. We discuss the choice of appropriate EC techniques (GA, GP, CGP, ES, multi-objective optimization) for various problems and evaluate on the importance of that choice. We discuss the gap between the crypto community and EC community and what does it mean for the results. By doing that, we give a special emphasis on the perspective that cryptology can represent a source of interesting benchmark problems for EC. We finish with a more general overview on artificial intelligence applications in security. This tutorial will present live demos of EC in action when dealing with cryptology problems.

2.11 Exploratory Landscape Analysis

Tutorial Speakers: Pascal Kerschke and Mike Preuss, *University of Müenster (Germany).*

Tutorial Abstract: Exploratory Landscape Analysis (ELA) has been conceived as an automated approach for characterizing optimization problems by extracting – not necessarily intuitively understandable – landscape features, based on a rather small initial sample from the underlying optimization problem. Within this tutorial we will introduce the general concept of (automated) algorithm selection, which is one of the main use cases of ELA, followed by a presentation of examples from different optimization domains, in which ELA has successfully been used to improve algorithm selection processes. After presenting the general idea of ELA and providing a detailed overview of its status quo (including recently published extensions), we will show how ELA can improve our understanding of (a) the characteristics of different problem landscapes, and (b) the behavior of optimization algorithms, which are executed on these problems.

The remainder of the tutorial will be used for an interactive live-demo, in which our participants will perform ELA on some continuous optimization problems.

2.12 Genetic Improvement: Taking Real-World Source Code and Improving It Using Genetic Programming

Tutorial Speakers: John Woodward, *Queen Mary University of London (UK)* and Saemundur O. Haraldsson, *University of Stirling (UK).*

Tutorial Abstract: Genetic Programming (GP) has been around for 25 years. Genetic Improvement (GI) is new. GI evolves source code, rather than a simulation of code. In other words, GI operates directly on Java or C, for example, whereas GP operates on a tiny instruction set defined by the function set and terminal set. Another fundamental difference is that GI starts with real-world software, whereas GP typically tries to evolve programs from scratch. These differences may not seem important; however this subtle difference opens new possibilities for research. Furthermore we can optimize the physical properties of code such as power consumption, size of code, bandwidth, and other non-functional properties, including execution time.

This tutorial is of interest to people with a GP background interested in applying their techniques to real source code, and software practitioners interested in using automated techniques to improve software. We will not assume prior knowledge of GP.

2.13 Introduction to Statistical Modeling of EC Systems and Experiments: A Visual Approach

Tutorial Speaker: Mark Wineberg, *University of Guelph (Canada).*

Tutorial Abstract: This tutorial is a follow-up to the statistics tutorial given at GECCO. While still at an introductory level (attendance of the GECCO tutorial is not assumed), the material covered is typically found in more upper year undergraduate courses on statistical modeling. We will move beyond simple comparisons using T tests, to the understanding and modeling of the underlying behaviors of various factors that may affect a system, even when those behaviors are obscured by the noise encountered in a stochastic system. Topics include: factor models, two factor linear regression confidence bands, multiple regression, polynomial regression, the relationship to GP-style symbolic regression, post-hoc analysis, and possibly one-way and multi-way ANOVA, time permitting. As with the GECCO tutorial, this tutorial takes a very visual approach to statistics; relying on graphics and animation to provide an intuitive understanding of the subject, instead of the traditional equations, which cater to only the initiated.

2.14 Learning Classifier Systems as Learning Cognitive Systems

Tutorial Speaker: Will Browne, *Victoria University of Wellington (New Zealand).*

Tutorial Abstract: Learning classifier systems (LCSs) are an often overlooked class of rule-based machine learning algorithms with a unique and flexible set of features that sets them apart from other strategies. The original LCS from 40 years ago, CS-1, stood for Cognitive System - One. Subsequently, LCSs have become powerful evolutionary machine learning techniques, but there is still much to be gained by exploring their cognitive systems capabilities. The tutorial will begin with a gentle introduction to LCSs based on the recent textbook 'Introduction to Learning Classifier Systems', which is co-authored by the presenter. The second part of this tutorial will show examples of Learning Classifier Systems in terms of cognitive systems. In-depth understanding will be provided regarding improved perception, representation, transfer learning and embodied LCSs. Examples will be discussed of solving previously intractable problems in a human-like manner as well as unique applications in robotics.

2.15 Mathematical Programming as a Complement to Bio-inspired Optimization

Tutorial Speaker: Ofer Shir, *Tel-Hai College and Migal-Galilee Research Institute (Israel).*

Tutorial Abstract: Global optimization of complex models has been for several decades approached by means of formal algorithms as well as Mathematical Programming (MP), and simultaneously has been treated by a wide range of dedicated heuristics - where nature-inspired approaches are placed. These two branches complement each other, yet practically studied under two independent CS disciplines. The claim that education within the scope of problem-solving from nature should encompass basic MP is untenable at present times, and this tutorial aims at bridging the gap for our scholars and students. The tutorial comprises two parts. The first part presents the fundamentals of MP. It overviews mathematical optimization in light of convex optimization versus combinatorial optimization. It discusses some of the theoretical aspects, such as polyhedra and the duality theorem. The second part focuses on MP in practice, particularly on modeling, and covers selected algorithms: Simplex, Ellipsoid, and Branch-and-Bound. The tutorial is planned for all PPSN participants, assuming no prior knowledge in mathematical optimization.

2.16 Multiagent Systems and Agent-Based Modeling and Simulation

Tutorial Speaker: Ana Bazzan, *Federal University of Rio Grande do Sul (Brazil).*

Tutorial Abstract: Multiagent systems (MAS) and agent-based modeling and simulation (ABMS) deal with social interactions among intelligent actors (agents). These two disciplines study neither just physical systems nor agents in isolation, but the agent as part of a social space.The goal of this tutorial is twofold: (a) about half of the time will cover basic material about MAS and ABMS (since this simulation paradigm is highly used by the computational intelligence community), and hence provide the audience a sense of the basic principles; and (b) about half of the time will cover the most recent advances in MAS, including the highly relevant topic of multiagent learning, one of the obvious interfaces between these two communities, as well evolutionary game theory.

2.17 Multi-objective Optimization with the jMetal Framework

Tutorial Speaker: Antonio J. Nebro, *University of Malaga (Spain).*

Tutorial Abstract: jMetal is a Java-based framework for multi-objective optimization with metaheuristics which has become popular in many disciplines (engineering, economics, bioinformatics, etc.). The journal paper describing jMetal has more than 775 citations according to Google Scholar, and it has been used by research groups, industry and academia. In this tutorial, we give a practical overview of the main jMetal components (algorithms, encodings, problems, operators, experiments, quality indicators), focusing on how to configure and run some of the included metaheuristics and also on how to incorporate new solution representations and problems. We give examples of classical algorithms but also more modern techniques, including preference-based metaheuristics. Special attention will be paid to the definition of experimental studies to statistically assess the performance of algorithms. The main goal is that the attendants can replicate all the examples presented, and the material needed to follow the tutorial will be available in a public repository (https://github.com/jMetal/PPSN2018Tutorial).

2.18 Next Generation Genetic Algorithms

Tutorial Speaker: Darrell Whitley, *Colorado State University (USA).*

Tutorial Abstract: New developments in Gray Box Optimization makes it possible to construct new forms of Genetic Algorithms that do not use random mutation or random recombination. Instead, for certain classes of NP Hard problems, it is possible to exactly compute the location of improving moves in constant time. In some domains, this makes random mutation obsolete. Deterministic "Partition Crossover" can be applied to optimization problems such as MAXSAT and the Traveling Salesman Problem. Partition Crossover locally decomposes a recombination graph into q subgraphs in $O(n)$ time. It can then identify the best of 2^q possible offspring. If the parents are local optima, the

offspring are guaranteed to be locally optimal in the largest hyperplane subspace containing both parents. Local decomposition has also been used to solve multiply constrained scheduling problems with unto 1 billion variables.

The book chapter "Next Generation Genetic Algorithms" will accompany the tutorial.

2.19 Runtime Analysis of Population-Based Evolutionary Algorithms

Tutorial Speaker: Per Kristian Lehre, *University of Birmingham (UK).*

Tutorial Abstract: Populations are at the heart of evolutionary algorithms (EAs). They provide the genetic variation which selection acts upon. A complete picture of EAs can only be obtained if we understand their population dynamics. A rich theory on runtime analysis of EAs has been developed over the last 20 years. This theory provides insights into how the performance of EAs depends on their parameter settings and the characteristics of the underlying fitness landscapes. Early studies were mostly concerned with EAs without populations, such as the $(1+1)$ EA. This tutorial introduces recent techniques that enable runtime analysis of EAs with realistic populations. To illustrate the application of these techniques, we consider fundamental questions such as: When are populations necessary for efficient optimisation? What is the appropriate balance between exploration and exploitation and how does this depend on relationships between mutation and selection rates? What determines an EA's tolerance for uncertainty?

2.20 Semantic Genetic Programming

Tutorial Speakers: Alberto Moraglio, *University of Exeter (UK)* and Krzysztof Krawiec, *Poznan University of Technology (Poland).*

Tutorial Abstract: Semantic genetic programming is a recent, rapidly growing trend in Genetic Programming (GP) that aims at opening the 'black box' of the evaluation function and make explicit use of more information on program behavior in the search. In the most common scenario of evaluating a GP program on a set of input-output examples (fitness cases), the semantic approach characterizes program with a vector of outputs rather than a single scalar value (fitness). The past research on semantic GP has demonstrated that the additional information obtained in this way facilitates designing more effective search operators. In particular, exploiting the geometric properties of the resulting semantic space leads to search operators with attractive properties, which have provably better theoretical characteristics than conventional GP operators. This in turn leads to dramatic improvements in experimental comparisons. The aim of the tutorial is to give a comprehensive overview of semantic methods in genetic programming,

illustrate in an accessible way a formal geometric framework for program semantics to design provably good mutation and crossover operators for traditional GP problem domains, and to analyze rigorously their performance (runtime analysis). A number of realworld applications of this framework will be also presented. Other promising emerging approaches to semantics in GP will be reviewed. In particular, the recent developments in the behavioral programming, which aims at characterizing the entire program behavior (and not only program outputs) will be covered as well. Current challenges and future trends in semantic GP will be identified and discussed.

Efficient implementation of semantic search operators may be challenging. We will illustrate very efficient, concise and elegant implementations of these operators, which are available for download from the web.

2.21 The Cartography of Computational Search Spaces

Tutorial Speaker: Gabriela Ochoa, *University of Stirling (UK).*

Tutorial Abstract: The performance of heuristic search algorithms crucially depends on the underlying fitness landscape structure. Most fitness landscapes analysis techniques study their local structure; there is a lack of tools to study instead their global structure, which is known to impact algorithms' performance. This tutorial will describe local optima networks (LONs), a model of fitness landscapes suited to analyse their global structure by bringing tools from complex networks. LONs provide new insight into the structural organisation and the connectivity pattern of a search space. We will cover the relevant definitions, extraction methodologies, metrics, and visualisation techniques to thoroughly characterise the global structure of computational search spaces. We will consider the landscapes induced by both evolutionary and local-search algorithms and will show results on combinatorial problems (binary and permutation spaces) as well as on computer program search spaces. An interactive demo will allow attendees to analyse and visualise realistic computational search spaces.

2.22 The Most Recent Advances on Multi-Modal Optimization

Tutorial Speakers: Michael G. Epitropakis, *University of Patras (Greece)*, Mike Preuss, *University of Müenster (Germany)*, and Xiaodong Li, *RMIT University (Australia).*

Tutorial Abstract: Multi-Modal optimization (MMO) is currently undergoing many changes, and becoming established as an active research area that collects approaches from various domains of operational research, swarm intelligence and evolutionary computation. Typically MMO strives for delivering multiple optimal (or close to optimal) solutions in a single optimization run. This tutorial will cover several scenarios and list currently employed and potentially available

performance measures. Furthermore, many state-of-the-art as well as more classic MMO methods are compared and put into a rough taxonomy. We will also discuss recent relevant competitions and their results and outline the possible future developments in this area. In brief, the tutorial will cover the following topics (in syllabus form):

- Multi-modal optimization - what for?
- Niching? Biological inspiration and optimization reality
- Towards theory: high-level modelling
- Suggested taxonomy and critical review of methods (with Demos)
- Spotlight: Clustering, multiobjectivization, surrogates and archives
- 'Measuring and different scenarios'
- Developing challenging competition benchmark function sets
- Discussion on current competition results, and available software
- Expected future developments.

2.23 Theory of Parallel Evolutionary Algorithms

Tutorial Speaker: Dirk Sudholt, *University of Sheffield (UK).*

Tutorial Abstract: Evolutionary algorithms (EAs) have given rise to many parallel variants, fuelled by the rapidly increasing number of CPU cores and the ready availability of computation power through GPUs and cloud computing. A very popular approach is to parallelize evolution in island models, or coarse-grained EAs, by evolving different populations on different processors. These populations run independently most of the time, but they periodically communicate genetic information to coordinate search. Many applications have shown that island models can speed up computation time significantly, and that parallel populations can further increase solution diversity. However, there is little understanding of when and why island models perform well, and what impact fundamental parameters have on performance. This tutorial will give an overview of recent theoretical results on the runtime of parallel evolutionary algorithms. These results give insight into the fundamental working principles of parallel EAs, assess the impact of parameters and design choices on performance, and contribute to the design of more effective parallel EAs.