Raymond BISDORFF

Algorithmic Decision Making with Python Resources

From Multicriteria Performance Records to Decision Algorithms via Bipolar-Valued Outranking Digraphs

January 20, 2022

Springer Nature

This book is dedicated to my colleague and dear friend, the late Prof. Marc ROUBENS.

Preface

The reader will find in this monograph a series of tutorials and advanced topics originally written over the last decade as documentation parts for the DIGRAPH3 collection of Python modules. These programming resources —like the `outrankingDigraphs` module— were essentially used both for the computational verification of decision algorithms and for the preparation and illustration of a Master Course on Algorithmic Decision Theory taught at the University of Luxembourg from 2010 to 2020. Some resources, like the `randomNumbers` module, served for preparing and illustrating the Lectures of a Computational Statistics Course. Curious readers will also discover some resources —the `arithmetics` module— used for preparing and illustrating a first Semester Course on Discrete Mathematics.

The DIGRAPH3 Python programming resources are useful in the field of Algorithmic Decision Theory and more specifically for the outranking approach of Multiple-Criteria Decision Aiding (MCDA). In this latter scientific field, we address essentially three kinds of usage.

First, we present algorithms and illustrate computing tools for solving either, a multiple-criteria first choice selection or, a dual last choice rejection problem. We also tackle the problem of how to list a set of items with multiple incommensurable performance criteria either, from the best to the worst (ranking problem), or from the worst to the best (ordering problem). Finally, we present order-statistical algorithms for relative or absolute quantiles-rating of multiple-criteria performance records.

It is necessary to mention that the DIGRAPH3 resources do not provide a professional Python software library. The collection of Python modules, I describe in this book, was not built following any professional software development methodology. The design of classes and methods was kept as simple and elementary as was opportune for the author. Sophisticated and cryptic overloading of classes, methods and variables is more or less avoided all over. A simple copy, paste and ad hoc customisation development strategy was generally preferred. As a consequence, the DIGRAPH3 modules keep a large part of independence. Furthermore, the development of the DIGRAPH3 modules being spread over two decades, the programming style did evolve with growing experience and the changes and enhancement coming up with the ongoing new releases of the standard Python3 libraries.

The purpose of this book is to present in a single monograph the methodological and scientific contributions the author made over the past two decades to the multiple-criteria decision aiding field and that are either left unpublished or solely published in very specialised media difficult to access.

This monograph should provide the reader with a self-contained series of tutorials which explain how to solve multiple-criteria selection, as well as ranking or rating decision problems. If successful in this aim, the curious reader will effectively install the DIGRAPH3 programming resources on their laptop and try out and redo for themselves the proposed computations.

The material in this book is valuable for master students and doctoral candidates in Computer Science, Mathematics, Engineering Sciences or Computational Management Sciences taking a course on Algorithmic Decision Theory, Multiple-Criteria Decision Aiding or Decision Analysis. Some experience in computer programming, in particular with Python, will assist the reader, but it is not a prerequisite. The many coding examples shown throughout the text are purposely kept elementary from a programming point of view.

Chapters presenting algorithms for ranking multiple-criteria performance records from best to worst —especially when facing big performance tableaux— will be of computational interest for designers of web recommender systems.

Similarly, the relative and absolute quantiles-rating algorithms, discussed and illustrated in several chapters, will be of practical interest for public or private performance auditors.

Finally, the monograph does not provide any mathematical developments or proofs. Those readers interested in the mathematical background of our decision algorithms are invited to consult the references provided at chapter level. Full texts of most of these references may be downloaded from the open access https://orbilu.uni.lu/ repository of the University of Luxembourg.

Acknowledgments

This monograph contains many ideas, methods and tools that are not only the author's. They have been shared and enhanced with friends, colleagues and many students. To name the most relevant: *Pascal Bouvry*, *Denis Bouyssou*, *Luis C Dias*, *Claude Lamboray*, *Patrick Meyer*, *Vincent Mousseau*, *Alex Olteanu*, *Marc Pirlot*, the late *Bernard Roy*, *Ulrich Sorger*, *Alexis Tsoukiàs*, *Thomas Veneziano* and especially, the late *Marc Roubens*. The UL HPC team, and more specifically *Valentin Plugaru* and *Sébastien Varette*, helped with mastering the multiprocessing experiments on the HPC platforms. The Springer publishing team very kindly assisted me eventually with the manuscript preparation.

The help of everybody is gratefully acknowledged.

Luxembourg, Autumn 2021 *Raymond Bisdorff*

Contents

Introduction

The editing strategy

In the five parts of this monograph, the reader will find several series of tutorials and advanced topics that present and illustrate computational methods and tools mainly useful in the field of Multiple-Criteria Decision Aiding and Decision Analysis. These methods and tools were designed and implemented first in Python2 and then in Python3 by the author over the last two decades for supporting both the computational verification and validation of decision algorithms as well as the preparation and illustration of a Master Course on Algorithmic Decision Theory.

Each chapter illustrates a specific preference modelling aspect, like building a best choice recommendation, ranking or rating a set of potential decision alternatives, or computing the winner of an election. In order to keep parts and chapters more or less self-contained, definitions and explanations of major concepts, like bipolar-valued digraphs, multiple-criteria performance tableaux and outranking situations, may appear several times in the monograph.

Explicit Python programming examples, purposely kept elementary, are shown in numerous terminal session style listings. A complete list of the numbered listings, shown over all the chapters, is printed in the Appendix. These programming examples were all checked against errors with the `doctest` module of the standard Python3 library and should work effectively as such either, in a Python3 interactive terminal console, or for sure in an `ipython` console[1]. Note that the layout of console `print(...)` outcomes has been edited in some listings for easing their reading. Some chapters will rely on a given data file that is made available in the `examples` directory of the DIGRAPH3 resources.

For similarly easing their reading, most chapters do not provide mathematical developments and proofs. Readers interested in such details are invited to consult the references listed separately at the end of each chapter. The author's references provide full text access to preprints on the open access https://orbilu.uni.lu/ repository of the University of Luxembourg.

[1] IP[i]: IPython interactive computing, https://ipython.org

Readers interested in the technical aspects of the organisation and implementation of the collection of DIGRAPH3 Python modules are invited to consult the extensive reference manual: https://digraph3.readthedocs.io/en/latest/techDoc.html, assisted by a search page https://digraph3.readthedocs.io/en/latest/search.html covering the whole DIGRAPH3 documentation.

Organisation of the book

The content of the monograph is divided into five parts.

Part I presents three chapters introducing the DIGRAPH3 programming resources and the main formal objects discussed in this book, namely *bipolar-valued digraphs* and, in particular, *outranking digraphs*.

In Chapter 1, the reader will gain contact with the DIGRAPH3 Python resources. First are given the installation instructions and the list of the main DIGRAPH3 Python modules with their purpose. A Python terminal session using the root `digraphs` module eventually illustrates how to generate, save and inspect a random crisp digraph.

Chapter 2 introduces the bipolar-valued digraph model—the root type of all our digraph models. A randomly bipolar-valued digraph instance is generated. Drawing the digraph, separating its asymmetric and symmetric parts, or its border and inner parts, is illustrated. The initial digraph instance may be reconstructed by epistemic disjunctive fusion from these respective parts. Dual, converse and codual transforms, as well as symmetric and transitive closures are presented. Complete, empty and indeterminate digraphs are eventually presented.

Chapter 3 presents the bipolar-valued outranking digraph—the main formal object used and discussed in this monograph. We first notice its hybrid type: it is conjointly a multiple-criteria performance tableau and a bipolar-valued relation modelling outranking situations between the given performance records. Pairwise comparisons of decision alternatives and the recoding of the digraph characteristic valuation are then illustrated. The codual transform of the outranking digraph renders the corresponding strict outranking digraph, i.e. its asymmetric part.

Part II illustrates in eight methodological chapters multiple-criteria performance evaluation models and decision algorithms. These chapters are mostly problem oriented.

Chapter 4 presents the RUBIS best choice recommender system. The approach is illustrated with building a best office-location recommendation. We show how to explore a given performance tableau and compute the corresponding outranking digraph. After presenting the pragmatic principles that govern our best choice recommendation algorithm we solve the best office-location choice problem.

Chapter 5 illustrates a way of creating a new multiple-criteria performance tableau by editing a given template file containing 5 decision alternatives, 3 decision objectives and 6 performance criteria. We discuss in detail how to edit the decision alternatives, the decision objectives, the family of performance criteria, and finally, the evaluations of the decision alternatives on the performance criteria.

Chapter 6 describes the DIGRAPH3 resources for generating random multiple-criteria performance tableaux. These random performance tableaux instances, mainly meant for illustration and training purposes, were serving the preparation and illustration of the Algorithmic Decision Theory Course lectures. The random generators propose several useful models like a Cost-Benefit tableau, a three Objectives —economic, societal and environmental–– tableau, and an academic performance tableau.

Chapter 7 is more specifically devoted to handling linear voting profiles and computing the winner of such election results like the simple majority or the instant runoff winner. By following CONDORCET's recipe, we consider pairwise comparisons of election candidates and balance the number of times the first beats the second against the number of times the second beats the first in order to obtain a majority margins digraph, in fact a bipolar-valued digraph. When the voters express contradictory linear voting profiles one may naturally observe cyclic social preferences without seeing any paradox in this situation. Finally, the chapter presents a more politically realistic generator for random linear voting profiles by taking into account pre-election polls.

Chapter 8 introduces several algorithms for solving multiple-criteria ranking problems via bipolar-valued outranking digraphs. The COPELAND, NETFLOWS, KEMENY, SLATER, KOHLER and the RANKEDPAIRS ranking rules are illustrated with the help of a random outranking digraph. The fitness of their respective ranking result is measured with a bipolar-valued version of KENDALL's ordinal correlation index.

Chapter 9 applies order statistics for sorting a set X of n potential decision alternatives, evaluated on m incommensurable performance criteria, into q quantile equivalence classes. The sorting algorithm is based on pairwise outranking characteristics involving the quantile class limits observed on each criterion. Thus we may implement a weak ordering algorithm of complexity $O(nmq)$.

Chapter 10 addresses the problem of rating multiple-criteria performance records of a set of potential decision alternatives with respect to performance quantiles learned from similar decision alternatives observed in the past. We show how to incrementally compute performance quantiles from incoming performance tableaux. New performance records may now be rated with respect to such historical quantile norms.

Chapter 11 tackles the ranking of big multiple-criteria performance tableaux with thousands or millions of records. To effectively compute rankings from performance tableaux of these sizes, the chapter proposes a collection of C-compiled and optimised modules that may be run on Linux Debian HPC clusters as available, for instance, at the University of Luxembourg.

Part III delivers three realistic algorithmic decision making case studies.

Chapter 12 presents a case study concerning the building of a best choice recommendation for Alice, a German student who wants some advice concerning the choice of her future University studies. We present Alice's performance tableau —

potential foreign language study programs, her decision objectives, performance criteria and performance evaluations— and build a best choice recommendation for her. A thorough robustness analysis confirms a very best choice.

In Chap. 13 we are resolving with our DIGRAPH3 resources a ranking decision problem based on published data from the Times Higher Education (THE) World University Rankings 2016 by Computer Science (CS) subject. We first have a look into the THE multiple-criteria ranking data with short Python scripts. In a second section, we relax the commensurability hypothesis of the ranking criteria and show how to similarly rank with multiple incommensurable performance criteria of solely ordinal significance. A third section is finally devoted to introduce quality measures for qualifying ranking results.

Chapter 14 presents and discusses how to rate with the help of our DIGRAPH3 resources the apparent student enrolment quality of higher education institutions. The multiple-criteria performance tableau, we use, is inspired by a 2004 student survey published by DER SPIEGEL magazine and concerning nearly 50,000 students, enrolled in one of fifteen popular academic subjects, like German Studies, Life Sciences, Psychology, Law or Computer Science.

In Chapter 15, we propose a series of decision problems of various difficulties which may serve as exercises and exam questions for an Algorithmic Decision Theory or Multiple-Criteria Decision Analysis course. They cover selection, ranking and rating problems.

Part IV presents in five chapters more advanced topics showing some pearls of bipolar-valued epistemic logic.

Starting from a motivating decision problem about how to list, from the best to the worst, a set of movies that are star-rated by journalists and movie critics, Chapter 16 shows that KENDALL 's ordinal correlation index tau can be extended to a relational bipolar-valued equivalence measure of bipolar-valued digraphs. This finding gives way, on the one hand, to measure the fitness and fairness of multiple-criteria ranking rules. On the other hand, it provides a tool for illustrating preference divergences between decision objectives and/or performance criteria.

We illustrate in Chapter 17, first, the concept of graph kernel, i.e. maximal independent set of vertices. In non-symmetric digraphs the kernel concept becomes richer and separates into initial and terminal kernels. In, furthermore, lateralized outranking digraphs, initial and terminal kernels become separate and may deliver suitable first, respectively, last choice recommendations. After commenting the tractability of kernel computations, we close the chapter with the solving of bipolar-valued kernel equation systems.

In Chapter 18 we propose to link a qualifying significance majority for outranking situations with a required α%-confidence level. We model therefore the significance weights as random variables following more or less widespread distributions around a mean value that corresponds to the given deterministic significance weights. As the bipolar-valued random credibility of an outranking situation hence results from the simple sum of positive or negative independent random variables, we can apply the Central Limit Theorem (CLT) for computing the bipolar-valued likelihood that the expected significance majority margin will indeed be positive, respectively negative.

In Chapter 19 we study the robustness of the outranking digraph when the criteria significance weights faithfully indicate solely an order of importance. The required cardinal significance weights of the performance criteria represent actually the 'Achilles' heel of the outranking approach. Rarely will indeed a decision maker be cognitively competent for suggesting precise decimal-valued criteria significance weights. This approach leads furthermore to the concept of unopposed or Pareto efficient multiobjective choices.

In a social choice context, where decision objectives would match different political parties, such Pareto efficient choices represent in fact multipartisan social choices. Chapter 20 shows that they may judiciously deliver the primary selection in a two stage election system. The outranking model is based on bipolar approvals-disapprovals of "*at least as well evaluated as*" statements. A similar approach is put into practice with bipolar approval-disapproval voting systems. When converting such approval-disapproval voting ballots into corresponding performance records, one obtains a $(-1,0,1)$-valued evaluative voting system. We eventually show in this chapter that in such bipolar voting systems, the election winner tends to be among the more or less multipartisan candidates.

Part V illustrates in three chapters computational resources for working with simple undirected graphs.

Chapter 21 introduces bipolar-valued undirected graphs and illustrates several special graph models and algorithms like Q-coloring, maximal independent set (MIS) and clique enumeration, line graphs and maximal matchings, grid graphs, and n-cycle graphs with their non-isomorphic MISs.

Chapter 22 specifically addresses working with tree graphs and graph forests. We illustrate how to generate and recognise random tree graphs and how to compute the centres of a tree and draw a rooted and oriented tree. Finally, algorithms for computing spanning trees and forests are presented.

Chapter 23 eventually presents some famous classes of BERGE graphs, namely comparability, interval, permutation and split graphs. We first present an example of an interval graph which is at the same time a triangulated, a comparability, a split and a permutation graph. The importance of being an interval graph is illustrated with *Claude Berge*'s mystery story. We discuss furthermore the generation of permutation graphs and close with how to recognise that a given graph is in fact a permutation graph.

Highlights

Contrary to what is generally thought, it is the preparation of the multiple-criteria performance tableau that takes most of the decision analysis time, not running any decision algorithms. Designing adequate performance evaluating criteria functions for each decision objective and collecting meaningful and precise evaluations is crucial for the success of the decision making. This is a very critical and essential step. Chapters 4, 5 and 12 illustrate and discuss in detail coherent multiple-criteria performance tableaux. In order to discover more examples of potential performance tableaux, we provide in Chap. 6 random generators for several common kinds of performance tableaux.

Once the multiple-criteria performance tableau is ready, starts the thrilling step of discovering the resulting outranking relation. Are there many chordless outranking circuits? What is its degree of symmetry? What is its degree of transitivity? If the number of potential decision alternatives is small—less than 30, one can try, in the case of a selection problem, to compute prekernels in order to find potential first or last choice decision alternatives? Chapters 4, 12 and 17 are illustrating and discussing this challenging computational problem.

Comparing various ranking rules working on bipolar-valued outranking relations constructed from performance tableaux of various kinds: Cost-Benefit, 3-Objectives, academic a.-o., has made us confident about the fact that convincing criteria for judging the quality of a ranking result may not to be found alone by mathematical properties, like KEMENY optimality or CONDORCET consistency. More useful seams to be the fair balancing of decision objectives and performance criteria. In this respect it is the NETFLOWS ranking rule which appears to be most effective and often gives fairly balanced multiple-criteria rankings. Chapter 8 on ranking rules, the ranking and rating case studies of Chaps. 13 and 14, and Chap. 16 on bipolar-valued relational equivalence of digraphs illustrate and discuss this important topic.

The bipolar-valued epistemic logic, in which our decision algorithms are computing and expressing their decision solutions, provides effective assistance for coping with missing data and imprecise performance evaluations. Chapters 14 and 16 illustrate this advantage. An efficient robustness analysis becomes furthermore available for handling, on the one side, uncertain criteria significance weights leading in Chapter 18 to $\alpha\%$-confident outranking digraphs. On the other side, Chapter 19 illustrates how to compute robust outranking digraphs and decision solutions when solely ordinal criteria significance weights are given. In Chapter 20, the same kind of robustness analysis proposes strategies for tempering plurality tyranny effects in social choice problems by favouring multipartisan candidates, like two-stage elections with multipartisan primary selection of candidates or bipolar approval-disapproval voting systems.

Noticing the efficiency of the bipolar-valued epistemic logical framework for handling outranking digraphs, we could not resist making in Chaps. 21 and 22 an excursion into the domain of simple undirected graphs and tree graphs. The beautiful book on Algorithmic Graph Theory and Perfect Graphs by *M. Ch. Golumbic* gave eventually the opportunity to tackle in the last Chapter 23 some famous classes of BERGE graphs.

It is my hope that the reader, by going on, will find the same astonishment and enchantment as I experienced when discovering the simplicity, efficiency and elegance of handling bipolar-valued outranking digraphs and graphs with Python programming resources. Extending the bipolar-valued epistemic logical framework to other computational science domains will prove valuable, I am sure, for many future scientific works.