University ”Alexandru Ioan Cuza” of Iaşi
Faculty of Computer Science

Efficient Hybrid Metaheuristics for Solving Hard Combinatorial Problems

PhD Candidate
Cristian Rotaru

Supervisor

Prof. PhD Henri Luchian

Thesis committee
Prof. PhD Daniela Zaharie,
West University of Timişoara, România
Prof. PhD Dan Dumitrescu,
Babes-Bolyai University, Cluj-Napoca, România
Prof. PhD Cornelius Croitoru,
”Al. I. Cuza” University of Iaşi, România

A thesis submitted to ”Alexandru Ioan Cuza” University
for the degree of Philosophiæ Doctor (PhD)
September 2012

Abstract

Evolutionary Computation is a research field that provides general and high quality algorithms, proved efficient in tackling difficult optimization problems. A common trait of these methods is the self-adaptation to the search landscape defined by the objective function of the solved problem instance, without in-depth knowledge of the problem domain. The success of Evolutionary Algorithms greatly relies on their ability to balance between the exploration and exploitation of the search space.

Since Wolpert and Macready's No Free Lunch Theorem, in 1995, the general opinion seems to be that none of the many existing heuristic methods can be declared the best one for solving any optimization problem. Each method, heuristic or exact, has its advantages and disadvantages depending on the search landscape defined by the optimized objective function. A natural research direction in the field of Evolutionary Computation is the hybridization of Evolutionary Algorithms. The immediate motivation is the development of hybrid systems, in which good components from different approaches are complementary working together, in order to provide a better solution to the problem than any of the source algorithms alone.

The purpose of this thesis is to investigate some hybrid designs that try to address the problem of efficiently balancing between exploration and exploitation. All methods have as basic component a genetic algorithm. Other heuristic and exact techniques are used collaboratively or are integrated directly in the genetic algorithm.

A first proposed approach describes the grafting of a Branch-and-Bound technique on a genetic algorithm. The hybrid method evolves embryos, instead of chromosomes. An embryo is a representation of a partial solution to the problem and represents a set of complete solutions. By evolving sets of solutions the method performs a better exploration of the search space and then focuses the search in the most promising regions.

A second algorithm, based on Cellular Genetic Algorithms, implicitly creates, maintains and exploits niches in the search space. Genetic material exchange between high quality niches explored in parallel intensifies the search in the corresponding regions of the search space.

The last approach, loosely related to the Island Model Genetic Algorithm, explicitly maintains and exploits niches in the search space. The last two methods integrate the use of clustering algorithms, principles borough from Tabu Search, genetic material exchange through communication protocols. Both methods are also distributed methods, efficiently employing the high computing power of modern processors.

Extensive experimental evaluations analyze the behavior and prove the performance of the proposed approaches. The assembly line balancing problem and the problem of optimizing the variable order in reduced ordered binary decision diagrams are used as test cases. Both problems are of great importance in industry and research.

Contents

Chapter I

Introduction

I.1 Research context

The field of Evolutionary Computation offers general and high quality algorithms used to tackle difficult optimization problems. These algorithms have the common property that they try to self-adapt to the search landscape defined by the objective function of the solved problem instance. They usually provide good quality solutions in limited amounts of time. Most of the methods are easy, cheap, fast to implement, and have a high degree of reusability.

The general opinion in the scientific community, at least from Wolpert and Macready's No Free Lunch Theorem, in 1995, seems to be that none of the many existing heuristic methods can be declared the best one for solving any optimization problem. Each method, heuristic or exact, has its advantages and disadvantages depending on the search landscape defined by the optimized objective function.

A natural research direction in the field of Evolutionary Computation, which already starts to take contour as a research field, is the hybridization of Evolutionary Algorithms. An immediate motivation for hybridization in general, not only for Evolutionary Algorithms, is to create systems in which good components from different approaches are complementary working together, in order to provide a better solution to the problem than any of the source algorithms alone.

The hybridization of Evolutionary Algorithms is a new, emerging research field. A generally accepted definition of the term hybrid metaheuristic and a methodology for devising hybrid systems do not exist. The first book devoted entirely to this subject has been published in 2008 (Blum and Roli [8]). The authors argue that the lack of a more formal definition of the domain "is rather helpful, as strict borderlines between related fields of research are often a hindrance for creative thinking and the exploration of new research directions".

The success of Evolutionary Algorithms greatly relies on the ability of the methods to balance between the exploration and exploitation of the search space. This subject is intensively debated in the research community. Large amounts of research results are dedicated to this subject. Also, many of the existing hybridization methodologies seem to address this problem.

The purpose of this thesis is to investigate some hybrid designs that try to address the problem of efficiently balancing between exploration and exploitation. All methods have as basic component a genetic algorithm. Other heuristic and exact techniques are used collaboratively or are integrated directly in the genetic algorithm.

A constant concern is to keep the proposed approaches as general as possible. The hybrid methods should be able to provide better solutions as result of the synergy between the components, not by intensively using problem-specific knowledge. This preserves the quality of the hybrid systems of being general approaches, easily adaptable to multiple problem types or instances. The performance improvement obtained by hybridization should reflect in multiple applications of the algorithm, not only in the case of the problem for which the algorithm was initially designed.

As the last decades have shown, the computing power of the processors that appear on the market grows exponentially. Most of the personal computers have two or four cores processors. The price of an eight cores processor is also affordable, especially for companies in industry. The need of efficient distributed designs for Evolutionary Algorithms is evident. Two of the three approaches proposed in this thesis are distributed algorithms, that efficiently employ almost the entire computing power of modern processors. The distribution is not limited to parallelization and aggregation of the results. Communication protocols are used in order to boost the quality of the solution.

Extensive experimental evaluations analyze the behavior and prove the performance of the proposed approaches. The assembly line balancing problem and the problem of optimizing the variable order in reduced ordered binary decision diagrams are used as test cases. Both problems are of great importance in industry and research. Beside showing the advantages of the hybrid systems over the basic methods used in the hybridization, the experimental evaluation has produced results competitive with the results in the specific research fields concerned with solving the used test problems.

I.2 Structure of the thesis

The thesis is organized as follows.

Chapter II, ”Evolutionary Computation”, describes the Evolutionary Computation field. Traditional approaches in the field of Evolutionary Algorithms and newer fields, like Swarm Intelligence, are presented. Due to the great interest posed to this thesis, Genetic Algorithms are described in more detail. Some approaches based on Genetic Algorithms, Cellular Genetic Algorithms and the Island Model of Genetic Algorithms, are further presented, as they are sources of inspiration for some of the approaches proposed in the thesis.

Chapter III, ”Hybridization of Evolutionary Algorithms”, is dedicated to the hybridization of Evolutionary Algorithms. Motivation and general principles are described. Following, a survey of the most used methodologies found in literature is presented.

Chapter IV, ”Test cases”, describes the test problems. Sections IV.1 and IV.2 are dedicated to the Assembly Line Balancing Problem and to the problem of optimizing the variable order in Reduced Ordered Binary Decision Diagrams, respectively. Both problems are defined formally and existing approaches, both exact and heuristic, are presented.

Chapter V, ”Basic methods and algorithms”, presents some well-known methods used in the proposed hybrid approaches. Section V.1 describes the Branch-and-Bound paradigm, further integrated with a genetic algorithm in the method described in Chapter VII. Section V.2 presents the Tabu Search metaheuristic, used as source of inspiration in the algorithm proposed in Chapter IX. The clustering problem, the k-Means and DBSCAN clustering algorithms and a clustering solution quality measure, used in both approaches proposed in Chapter VIII and Chapter IX, are described in section V.3.

Chapter VI, ”Basic approaches to the test problems”, presents the basic methods for the test problems further used in the proposed hybrid algorithms. Section VI.1 describes the basic genetic algorithm for the assembly line balancing problems. The algorithm is the main component in the methods described in Chapter VII and Chapter IX. Section VI.2 presents a simple Branch-and-Bound approach for the assembly line balancing problems. Elements from this method are grafted on the basic genetic algorithm previously detailed, the obtained method being thoroughly described and analyzed in Chapter VII. Section VI.3 describes the basic genetic algorithm for the problem of optimizing the variable order in reduced ordered binary decision diagrams. The method is further used in the cellular genetic algorithm and the segregative genetic algorithm, detailed in Chapter VIII and Chapter IX, respectively.

Chapter VII, "Grafting a Branch and Bound method on a Genetic Algorithm", describes the methodology of grafting elements and principles from a Branch-and-Bound method on a Genetic Algorithm. The approach is detailed in the context of solving the variants of the assembly line balancing problem described in section IV.1. The basic methods used in the hybridization are the genetic algorithm presented in section VI.1 and the Branch-and-Bound algorithm from section VI.2. The main design feature is that the hybrid genetic algorithm evolves incomplete chromosomes, called embryos, instead of complete chromosomes, for a large part of the evolution process. The embryos represent sets of possible solutions to the problem, whereas chromosomes, or adults, represent single, complete solutions. The estimation function from the Branch-and-Bound method is used to compute the fitness for incomplete solutions. A newly introduced growing operator, based on the branching rules of the Branch-and-Bound method, is used to evolve the population toward complete solutions.

Chapter VIII, "Cellular Genetic Algorithm with Parallel Evolving and Communicating Grids", presents a cellular genetic algorithm, in the context of optimizing the variable order in reduced ordered binary decision diagrams. The evolution process is inspired by the basic genetic algorithm described in section VI.3. The population evolves on a bidimensional grid and is implicitly organized in geographical clusters that present a form of structural similarity between individuals. Two feature functions are used to measure the similarity between chromosomes. The approach considers multiple parallel evolving grids. A similarity-based communication protocol between clusters of individuals from parallel grids is defined. The exchange of genetic material proves to considerably boost the quality of the solution. The algorithm strives to achieve a better balance between exploration and exploitation by implicitly creating and preserving niches in the search space and intensifying the search in the corresponding high quality regions of the search space.

Chapter IX, "Segregative Genetic Algorithm", presents a segregative genetic algorithm used for solving both used test problems. The approach integrates the basic genetic algorithms described in sections VI.1 and VI.3, for both problems, respectively. Feature functions are used in order to define a similarity measure between chromosomes. Subpopulations of individuals, formed by applying a clustering procedure in the obtained feature space, are explored in parallel by multiple copies of the basic genetic algorithm. A communication protocol preserves the similarity inside each subpopulation during the evolution process. The redundant exploration of the search space is avoided by using a tabu search associative memory. Genetic material from yet unexplored regions of the search space is managed and organized in order to explicitly guide the search process to yet undiscovered local optima. The method has as main purpose to explicitly create and maintain niches in the search space and to intensify the search in the corresponding regions of the search space.

All described algorithms, presented for solving the two test problems, basic and hybrid, are extensively evaluated on classical and publicly available benchmark problem instances. The same quality measures are reported in all evaluations. All experiments are performed in exactly the same environment. Conclusive and fair comparisons are achieved between different methods. The conclusions are detailed and proved significant through statistical tests.

All algorithms have been implemented in order to produce the results presented in the thesis. In the case of the methods described in Chapter VIII and Chapter IX the implementation is distributed. The efficiency of resource utilization in a multiprocessor environment is analyzed.

I.3 Contributions of the thesis

The original contributions of the thesis are enumerated in the following paragraphs:

- an up-to-date survey on the most popular methodologies for the hybridization of Evolutionary Algorithms (Chapter III). Approaches from the literature are presented and examples are given.
- an up-to-date survey on existing approaches to the used test problems: the Assembly Line Balancing Problem (section IV.1.2) and the problem of optimizing the variable order in Reduced Ordered Binary Decision Diagrams (section IV.2.4).
- a genetic algorithm for solving the Simple Assembly Line Balancing Problem with I- and U-shaped line layouts (section VI.1). The algorithm integrates a mechanism for dynamically controlling the probabilities of the genetic operators, proved to give good results for other formulations of the Assembly Line Balancing Problem (section VI.1.3.1). The well-known permutation representation is used and some simple and efficient evaluation procedures

are proposed (section VI.1.1): one for the I-shaped line layout and two for the U-shaped line layout.

- a feature or characteristic vector, used to define a similarity measure between chromosomes that represent solutions in the genetic algorithm for the Assembly Line Balancing Problem (section VI.1.6). A real-valued vector is attached to each chromosome and the Euclidean distance is used to easily define the similarity between permutation-based representations.
- two feature vectors used in a similar manner to define the similarity measure between the chromosomes based on order representation, employed by the genetic algorithm for optimizing the variable order in Reduced Ordered Binary Decision Diagrams (section VI.3.6).
- a novel and efficient approach to the Simple I/U-shaped Assembly Line Balancing problems, based on the hybridization of a basic genetic algorithm with elements of the Branch-and-Bound paradigm (Chapter VII). The idea of partial solution representation from previous research is used. Original evaluation procedures for both problem variants, based on estimation functions specific to Branch-and-Bound, are proposed (section VII.3). A growing operator based on simple heuristics specific to the tackled problems is introduced in order to simulate the branching rules from Branch-and-Bound (section VII.4.2). The search strategy employed by the exact approach is simulated by dividing the evolution process performed by the hybrid genetic algorithm in epochs and by devising growing strategies, that guide partial solutions towards complete ones (section VII.5).
- a novel and efficient approach to the problem of optimizing the variable order in Reduced Ordered Binary Decision Diagrams, based on Cellular Genetic Algorithms (Chapter VIII). The proposed approach integrates elements from a basic genetic algorithm with principles from the Predator-Prey Cellular Genetic Algorithm. An original method of placing the initial population in the evolution environment, based on self-organizing Kohonen networks is described (section VIII.2). Placing rules for genetic operator offspring are proposed (sections VIII.3.1 and VIII.3.2). A well-known density-based clustering procedure is used to organize the population in the environment of evolution and to guide the predators in order to achieve the selection process (section VIII.4). Simple and natural survival techniques are introduced in order protect high quality individuals (section VIII.4.3). A multi-grid distributed approach is presented (section VIII.6) and the quality of the solution is significantly boosted by employing the known idea of

genetic material exchange, via a communication protocol between parallel evolving populations. The previously mentioned similarity measure between chromosomes and clustering techniques are used to devise new variants for the communication protocol.

- a novel and efficient approach to both used test problems, called Segregative Genetic Algorithm, loosely based on the Island Model Genetic Algorithm (Chapter IX). As in the IMGA, multiple populations are evolved in parallel, but in the case of the proposed approach the search is more focused on clearly defined regions of the search space (section IX.2). This is achieved by a communication protocol, that preserves structural similarity inside the evolved subpopulations (section IX.2.3). The evolution process is performed by the simple genetic algorithms previously presented. The similarity measure is computed by using the proposed feature vectors. An efficient memory structure, based on the Tabu Search paradigm, is introduced in order to avoid redundant exploration of the search space (section IX.2.2.1). The search space is systematically explored by guiding the evolution process and focusing the search in newly discovered local optima (section IX.2.2.2).
- extensive experimental evaluations of all methods proposed in section VI.1, section VI.3, Chapter VII, Chapter VIII and Chapter IX. All procedural choices and their effect on the algorithms behavior are thoroughly analyzed. Performance results are reported in a systematical manner, by using the same quality indicators and well-known benchmark problem instances from literature. The results are compared with existing approaches from literature and the superior quality of the proposed algorithms is underlined. In-depth comparisons between the basic methods and the proposed hybrid methods are presented in order to prove the performance boost achieved by hybridization.
- a high quality framework, in which all the used and proposed methods are implemented. The library is written in the C and C++ programming languages and was used to produce all the results in the experimental evaluation sections in all chapters. The library is quite general and can be easily used to apply the proposed methods on other difficult optimization problems. All implementations are efficient and, where is the case, are parallel or distributed and make use of almost the entire computing power of modern multi-core processors.

I.4 Publications

All research in this thesis has been published in the following papers:

- Cristian Rotaru, Octav Brudaru, 2012. *Dynamic Segregative Genetic Algorithm for Optimizing the Variable Ordering of ROBDDs.* In Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference, GECCO '12, p. 657 – 664 (Philadelphia, PA, USA). ACM, New York, NY, USA. ISBN: 978-1-4503-1177-9. doi: 10.1145/2330163.2330256. URL: `http://doi.acm.org/10.1145/2330163.2330256`.
- Cristian Rotaru, Octav Brudaru, 2012. *Multi-grid cellular genetic algorithm for optimizing variable ordering of ROBDDs.* In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, IEEE CEC 2012, p. 1 – 8 (Brisbane, QLD, Australia). IEEE Press. E-ISBN: 978-1-4673-1508-1. Print ISBN: 978-1-4673-1510-4. doi: 10.1109/CEC.2012.6256590. URL: `http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6256590`.
- Octav Brudaru, Cristian Rotaru, 2011. *Grafting a branch and bound method on a genetic algorithm for balancing I / U-shaped assembly lines.* Memoriile Sectiilor Stiintifice / Memoirs of Scientific Sections, Series IV, Tome XXXIV, p. 137 – 160. Publishing House of the Romanian Academy. ISSN: 1224-1407. URL: `http://www.academiaromana-is.ro/Reviste/mem_sc_st_2011/09_MSS_Brudaru_Rotaru.pdf`.
- Octav Brudaru, Cristian Rotaru, Iulian Furdu, 2011. *Static Segregative Genetic Algorithm for Optimizing Variable Ordering of ROBDDs.* In Proceedings of the 2011 13th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC '11, p. 222 – 229 (Timisoara, Romania). IEEE Computer Society, Washington, DC, USA. ISBN: 978-0-7695-4630-8. doi: 10.1109/SYNASC.2011.54. URL: `http://dl.acm.org/citation.cfm?id=2195576`.
- Octav Brudaru, Cristian Rotaru, 2010. *Dynamic Segregative Genetic Algorithm for Assembly Lines Balancing.* In Proceedings of the 2010 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC '10, p. 229 – 236 (Timisoara, Romania). IEEE Computer Society, Washington, DC, USA. ISBN: 978-0-7695-4324-6. doi: 10.1109/SYNASC.2010.39. URL: `http://dl.acm.org/citation.cfm?id=1957189`.

Chapter II

Evolutionary Computation and Hybridization of Evolutionary Algorithms

II.1 Evolutionary Computation

Evolutionary Computation is a general term that describes the class of optimization techniques mainly inspired by the principles of natural evolution. These algorithms, usually called Evolutionary Algorithms (EAs), are methods that simulate the natural evolution of organisms. During this process, a number of individuals inhabit the same environment and compete for common resources. Natural laws such as the "survival of the fittest" principle, proposed by Charles Darwin in 1859, guide the natural selection process, that will dictate which organism will have descendants and the amount of these descendants in future generations. Organism that are capable of better acquiring resources will pass on their characteristics with a high probability. Lesser organism will be slowly removed from the population.

The field of EC and EAs emerged as a response to the apparition of difficult and large instances of real-world problems, that are no longer tractable through classic, deterministic methods. In simulating the natural process of evolution all EAs have an intrinsic probabilistic behavior. The main advantage of these algorithms is that they try to adapt to the fitness landscape of the given problem. EAs are general heuristics, applicable to a large class of optimization problems.

II.1.1 Evolutionary Algorithms

EAs are iterative methods, that use a population of candidate solutions to explore the search space of a problem. The members of the population are called individuals or chromosomes. An individual is an internal representation or encoding of an actual solution to the given problem. A decoding procedure is used to translate from the space of internal representations, called genotype, to the space of actual solutions, called phenotype.

The population evolves over a number of generations. The iterative process stops when a halting criterion is satisfied. A fitness function is used to evaluate the quality of each individual in the population. Depending on their quality, some individuals are chosen to be parents. The parents produce offspring by means of reproduction or genetic operators. The classical reproduction operators are mutation and crossover. A selection method, also based on fitness, is applied to decide which individuals remain in the population of the next generation. The procedures used for choosing parents and selection usually involve some degree of randomness, making EAs probabilistic methods.

EAs are general heuristics, without in-depth knowledge of the given problem instance. They try to adapt to the search space by exploring multiple regions. The principles inspired from natural evolution are intended to guide the search process toward high quality regions of the search space. EAs, as any probabilistic algorithms, do not always guarantee the discovery of the optimal solution. Usually, at least high quality sub-optimal solutions are returned. General purpose EAs are usually easy to develop and can be used to successfully tackle a large class of optimization problems.

The traditional classes of Evolutionary Algorithms are: genetic algorithms, evolutionary programming, evolutionary strategies and genetic programming. The differences between these approaches are given by the choice of solution representation, the implementation of genetic operators and the used procedures for selection. Detailed reviews of research directions in EAs can be found in: Bäck et al. [3], Bäck et al. [4], Eiben and Smith [21]. The thesis briefly describes all approaches and underlines the main features particular to each method.

II.1.2 Genetic Algorithms

Genetic Algorithms (GAs) represent the most popular class of Evolutionary Algorithms. The method was introduced by Holland in 1975 (Holland [29]). Since then, many variations to the traditional algorithm were proposed, but the principles of the method remained unchanged.

GAs employ a population of solution candidates to explore the search space of the given problem. The performed iterative process mimics the process of natural evolution, based on the survival of the fittest principle. The main components of a GA are: a representation or encoding of the solution, a fitness function used to evaluate individuals, a selection procedure, genetic operators and a stopping condition.

The thesis describes in more detail each of the components of a genetic algorithm and briefly presents the most popular approaches in literature to each subject. Some more advanced approaches regarding genetic algorithms are also described, as being of great interest for the thesis: the concept of niching (Deb and Goldberg [18], De Jong [16]), Cellular Genetic Algorithms (Alba and Dorronsoro [2]) and the Island Model Genetic Algorithm (Whitley and Starkweather [45]).

II.2 Hybridization of Evolutionary Algorithms

When confronted with the question whether one of the many existing heuristics can be declared the best one for solving any problem, the scientific community seems to have accepted that the answer is no. Wolpert and Macready showed in 1995 that all algorithms that optimize an objective function perform the same when averaged over all possible objective functions (the No Free Lunch Theorem, Wolpert and Macready [46]). A first motivation for hybridization in general, not only for Evolutionary Algorithms, is to create systems in which good components from different approaches are complementary working together, in order to provide a better solution to the problem than any of the source algorithms alone.

The field of Evolutionary Computation offers general and high quality algorithms for difficult optimization problems. These algorithms self-adapt to the given problem instance and usually provide a good enough result in limited time. The methods are usually easy, cheap, fast to implement, and have a high degree of reusability. Decades of research in Evolutionary Computation in general, and in Evolutionary Algorithms in particular, produced a high variety of nature inspired metaheuristics, used to tackle with great success a large range of optimization problems. Even so, these methods are mostly outperformed by algorithms tailored for the given problem specifically. The main reason for this issue is that the later approaches exploit particularities of the search space, by incorporating problem-specific knowledge. As a result, a direction in the hybridization of EAs is to devise more problem-specific procedures and operators.

A good practice should be to keep the newly obtained problem-aware EAs as general as possible. The hybrid methods should be able to provide better solutions as result of the synergy between the components, not by intensively using problem-specific knowledge. This preserves the quality of the hybrid systems of being general approaches, easily adaptable to multiple problem types or instances. The performance improvement obtained by hybridization should reflect in multiple applications of the algorithm, not only in the case of the problem for which the algorithm was initially designed. The success of EAs greatly relies on the ability of the methods to balance between the exploration and exploitation of the search space. Many developed hybrid systems address this problem.

The hybrid algorithms proposed in the thesis make use of problem-specific information, but the integration in the system is quite well delimited. The methods are general and the performance increase is due to the quality features appended by all used ingredients. The application of the algorithms to other problems than the ones used as test subjects in the thesis is easily achievable. The problem-specific knowledge is introduced in the system by devising some clearly defined procedures, that do not interfere directly with the execution of the hybrid algorithms. The main goals achieved by the hybrid approaches are: better solution quality and stability. Both goals are achieved by devising a more systematic search process, with a good control of the balance between exploration and exploitation. The presentation of each hybrid algorithm underlines all benefits of the hybrid systems in comparison with the used basic approaches.

The thesis presents a detailed review of the most popular hybridization techniques in literature: EAs & EAs, EAs & Swarm Intelligence, EAs & local search (Memetic Algorithms), EAs & exact methods, EAs & problem knowledge. Examples are given for each combination.

Chapter III

Test cases and basic methods

III.1 Test cases

The thesis presents three hybrid evolutionary algorithms. A main focus is to thoroughly describe the methods and to prove the advantages of the hybrid approaches, when compared to some of the basic components used in the hybridization. Other important concerns are the comparison between the hybrid methods themselves and the comparison with other similar approaches from the literature.

In order to achieve these comparisons, some difficult combinatorial problems where chosen as test cases for all algorithms: the Assembly Line Balancing Problem (ALBP) and the problem of optimizing the variable order in Reduced Ordered Binary Decision Diagrams (ROBDD). Both problems are NP-complete and have important theoretical and practical applications. The presented methods are novel approaches for the chosen problems.

Assembly lines are one of the most popular means to achieve mass and large-scale series production. The basic idea is to split the production process of some object into multiple indivisible operations. The product units are transported by mechanical means, a conveyor belt, for example, through a series of consecutive stations. At each station, the same operations, or tasks, are repeatedly performed on pieces from different products. All operations have some processing time. The maximum processing time of any station on the line dictates the production rate of the line. The production rate is the time interval at which a new final product exits the assembly line.

As a general formulation, the assembly line balancing problem consists in optimally partitioning the set of tasks among the workstations, with regard to some objective function. All constraints must be met. Early formulations of the assembly line balancing problem can be found in Salveson [38]. The basic variant of the problem, which is considered in the thesis, called Simple Assembly Line Balancing Problem (SALBP) is clearly formulated in Baybars [5]. A few of the many comprehensive reviews and surveys on the formulations of the problem and methods, that appeared over the years, can be found in: Kilbridge and Wester [31], Mastor [34], Johnson [30], Talbot et al. [41], Baybars [5], Ghosh and Gagnon [24], Becker and Scholl [7]. The last two works contain systematic classifications of all problem variants available in literature. Wee and Magazine [43] shows that the assembly line balancing problem is NP-hard.

Binary Decision Diagrams (BDDs) are data structures used to represent switching functions (boolean functions). They were introduced in Lee [33] and Akers [1]. Many types of BDDs were proposed, the most known ones being the Ordered BDDs (OBDDs) and Reduced OBDDs (ROBDDs), described in Bryant [13]. Applications of BDDs can be mentioned in: digital circuit design, symbolic model checking, formal verification of combinatorial circuits, analysis of sequential systems, graph algorithms, complexity theory. Examples for the mentioned fields and many others can be found in: McMillan [35], Clarke et al. [14], Minato [37], Wegener [44], Drechsler and Becker [19], Hachtel and Somenzi [28], Meinel and Theobald [36].

An ROBDD is a single-rooted directed acyclic graph. The size of the ROBDD is defined as the number of non-terminal nodes in the graph and is highly dependent on the order of the variables of the boolean function. The goal is to minimize the size. For example, when a digital circuit is represented by a switching function, a smaller size of the BDD directly transfers to a smaller chip area.

The thesis formally describes the two problems and presents up-to-date reviews of the methods, exact and heuristic, used to tackle the problems.

III.2 Well-known methods

The well-known methods used as sources of inspiration or directly in the hybrid approaches are the following:

- Branch-and-Bound – a deterministic method, used to systematically explore the search space of a problem, by building the tree representation of the search space. Entire branches can be excluded by using estimation and bounding functions. The tree is build by applying branching rules. Search strategies guide the search process. The method was proposed in 1960 in Land and Doig [32] and further developed in: Beale and Small [6], Dakin [15], Driebeck [20], Forest et al. [22], Tomlin [42].

- Tabu Search – an iterative local search metaheuristic, proposed by Glover in 1986 (Glover [25]) and further developed in Glover [26], Glover [27] and de Werra and Hertz [17]. The Tabu search metaheuristic introduces the idea of keeping track of the trajectory of the solution through the search space. The main feature of the approach consists in the use of a memory to guide the search process. The content of the memory is dynamically modified, depending on the number of the iterations and the solutions visited in the search process.
- clustering techniques – the k-Means algorithm, a centroid-based clustering method, the DBSCAN algorithm, a density-based method, and the Davies-Bouldin Index for the quality of the clustering solution are thoroughly presented.

The thesis formally describes all methods. Examples of applications are given. Many literature references, for further details, are indicated.

III.3 Basic methods

The thesis describes two basic methods for the ALB problem, further used in the hybrid methods. First, a basic genetic algorithm (BGA) is presented. The method uses the permutation representation. Original fitness evaluation procedures are proposed. Well-known genetic operators from literature, suitable to the chosen representation are applied. A mechanism based on the variability of the population dynamically controls the operator probabilities. The BGA for ALB is the basic component of the hybrid methods presented in Chapter VII and Chapter IX in the thesis.

The second basic method is a very simple Branch-and-Bound approach to ALB. An idealistic estimation function is used. Elements of the method are grafted on the BGA for ALB. The resulting method is thoroughly described in Chapter VII.

In the case of optimizing the variable order in ROBDDs, the thesis describes and further uses in the hybrid approaches, a simplified version of the genetic algorithm from Furdu and Brudaru [23]. The method is integrated in the methods proposed in Chapter VIII and Chapter IX in the thesis.

The thesis details all aspects of the methods. The algorithms have been implemented and the experimental results are reported and further used in the comparisons with the hybrid approaches.

Chapter IV

Proposed approaches

The thesis describes three hybrid approaches based on the presented basic and well-known methods and algorithms. All methods are concerned with achieving a better balance between exploration and exploitation of the search space. Chapters VII, VIII and IX, in the thesis, that present the proposed methods are structured as follows. In the first part, all elements of the algorithms are thoroughly detailed. The specific procedures and architectural choices are presented. The second part of each chapter contains the experimental evaluation of the methods. The effect of the procedural choices and the behavior of the methods is analyzed. The performance evaluation follows the same steps, in the same environment and reports the same quality indicators on the same well-known and publicly available problem instances. The methods are compared with other methods from literature, with the basic methods used in the hybridization, and between themselves. The advantages achieved by the three different hybrid architectures are clearly underlined.

IV.1 Grafting a Branch-and-Bound method on a Genetic Algorithm

Chapter VII in the thesis presents a hybrid method that combines a genetic algorithm and a branch-and-bound technique. The elements of the approach are described in the context of solving the two variants of the assembly line balancing problem, presented in section IV.1, in the thesis: SALBP-1 and UALBP-1. The basic ingredients of the approach are the genetic algorithm for I- and U-shaped assembly lines balancing described in section VI.1 and some elements from the branch-and-bound method outlined in section VI.2. The methodology of grafting elements from the branch-and-bound method on the genetic algorithm is quite general and can be applied for solving any combinatorial problem, for which some specific procedures used in the hybridization can be provided.

The main design feature is that the genetic algorithm evolves incomplete chromosomes, or embryos, instead of complete chromosomes, for a large part of the evolution process. The embryos are prefixes or pairs of prefixes–suffixes of permutations and represent sets of possible solutions to the problem. The estimation function from the branch-and-bound method is used to compute the fitness for incomplete solutions. A newly introduced growing operator, based on the branching rules of the branch-and-bound method, is used to evolve the population toward complete solutions. Growing and survival schemes are proposed for preventing the elimination of long embryos, with more accurate, but not good enough fitness values.

The experimental results show that the proposed embryonic genetic algorithm dominates the basic genetic algorithm, in terms of solution quality, solution stability and computational efficiency. This is mainly due to a better trade-off between the exploration and exploitation or the search space, achieved by the hybrid approach. A good exploration is achieved in the first part of the evolution process, by the growing population of embryos. Managing subsets of solutions, rather than single solutions, provides the means of disregarding large low-quality regions of the search space entirely. The search is guided toward promising regions from early stages of the process. The exploitation is achieved by evolving complete solutions, after the complete maturation of the population.

Previous attempts to graft a branch-and-bound technique to a genetic algorithm can be found in: Brudaru and Valmar [9], for assembly line balancing with fuzzy times; Brudaru and Valmar [10], for a delivery problem; Brudaru et al. [11], for the k-repairmen problem. The proposed approach inherits some principles from these methods, but the design is modified and introduces new and efficient components.

Future work is envisioned in the following directions:

- One or more tighter lower bounds can be used instead of the idealistic and very poorly performing one included in the current approach. Some strong and computationally inexpensive lower bounds, used in the most successful current approaches, can be found in Scholl and Becker [39].
- The basic growing operator employs three simple rules adapted from constructive priority-based heuristics. Current reviews concerning this research area describe around 20 – 25 rules (Scholl and Voß [40]). The growing operator is also very similar to the branching in station-oriented branch-and-bound methods. Scholl and Becker [39] describes high qual-

ity branching procedures along with domination rules, that are successfully used to prioritize the newly grown solutions. More attention can be focused in improving the quality of the chromosomes generated by growing.

- The staircase growing strategy accomplishes its purpose in guiding the maturation of chromosomes, but it is quite simplistic, taking into account the decisive role it plays in guiding the search process. A common sense idea would be to make the now fixed growing period variable and dependent on the moment of evolution. The evolution of similar length chromosomes corresponds to searching high quality subtrees in some small range of levels of the branch-and-bound search tree. As the length of embryos grows, the levels of the tree become more larger. The algorithm may benefit from concentrating more computational resources on higher levels of the tree.

IV.2 Cellular Genetic Algorithm with Parallel Evolving and Communicating Grids

Chapter VIII in the thesis presents a hybrid approach based on the Cellular Genetic Algorithm design. The main goal of the approach is to achieve a better balance between exploration and exploitation by creating niches in the search space and intensifying the search in the corresponding high quality regions of the search space. This effect is achieved by encouraging the geographical grouping of individuals in the environment of evolution, in clusters with a high degree of structural similarity between individuals. The clusters are created and maintained implicitly, by exploiting design features of the genetic operators and procedures specific to the cellular approach.

A distributed variant of the proposed approach, in which multiple copies of the algorithm are executed in parallel, is described. A communication protocol is introduced in order to share genetic material from high quality niches between populations evolved in parallel. The effect consists in the intensification of the search process in the corresponding regions of the search space. A great performance boost is observed for the variant of the algorithm that uses the communication protocol. The algorithm offers an efficient means of using the available computational resources, in order to produce better results, in terms of both stability and quality of the solution.

The approach starts with a basic genetic algorithm and adapts it in order to achieve the cellular design. The procedures specific to the cellular design are thoroughly described. New and original improvements are introduced in order to obtain better performance. The whole approach can be viewed as a collaborative effort involving the basic genetic algorithm, clustering techniques and a local search technique, based on the communication protocol.

The approach is presented in the context of solving the problem of optimizing the variable ordering in Reduced Ordered Binary Decision Diagrams. The design of the algorithm is general and can be easily adapted in order to solve other optimization problems. An approach based on the cellular design to the Assembly Line Balancing Problem can be found in Brudaru et al. [12].

The results prove the high performance of the proposed approach. A thorough comparison shows the improvements achieved by the cellular algorithm when compared with the basic genetic algorithm. Comparisons with other methods in the literature are also provided. Another more in-depth comparison is achieved with the segregative approach proposed in Chapter IX. The cellular approach can be viewed as an intermediate design between the basic and the segregative genetic algorithm, in which the niching mechanism is achieved explicitly, by evolving multiple populations in parallel.

Future work is focused on:

- improving the quality of the feature functions, to better discern between highly similar chromosomes;
- hybridization with a branch-and-bound method, by evolving embryos in the early stages of the process;
- improving the quality of the selection phase (devouring) by allowing the predators to take part in the evolution process (coevolution).

IV.3 Segregative Genetic Algorithm

Chapter IX describes a hybrid genetic algorithm with the main goal of explicitly creating and maintaining niches in the search space during the evolution process, in order to achieve a better balance between exploration and exploitation. The approach is related to the Island Model Genetic Algorithm, in the sense that it evolves multiple populations in parallel.

The proposed approach is a collaborative effort of several algorithms. Subpopulations of individuals are explored in parallel by copies of a basic genetic algorithm. The subpopulations are obtained by applying a clustering algorithm in the feature space defined by the feature vectors attached to each chromosome. A communication protocol between parallel evolved subpopulations preserves the similarity inside each subpopulation. Organizing the individuals in subpopulations, each corresponding to a niche in the search space, augments the exploratory performance of the algorithm. The evolution process performed by the copies of the basic genetic algorithm on each subpopulation represents an intensification of the search in the corresponding regions of the search space. A memory inspired from the Tabu Search paradigm is used to keep track of the exploratory process, in order to avoid redundant exploration of the search space. New genetic material produced during the evolution process is organized and prioritized for exploration.

Another goal of the approach is to describe a general design, easily adaptable to a large class of optimization problems. The algorithm is tested on both used problems. Given a basic population-based heuristic and a similarity measure between individuals, the proposed approach can be applied on any problem without further modifications. The great potential for parallel computation makes the segregative algorithm a good choice if computational resources are available. The organized search process, guided by well-defined characteristics of the search space, proves to be superior in performance than restarting or parallel evolving of populations with the basic genetic algorithm and aggregating the results. The algorithm scales well with the size of the tackled problem instance. The performance of the algorithm in terms of quality and/or execution time can be boosted without modifying the design, only by adding computational resources to the system and exploring more subpopulations in parallel.

The extensive experimental evaluation proves the performance of the algorithm in terms of both stability and quality of the solution, when compared with the basic genetic algorithm and with other approaches in the literature. An in-depth comparison with the cellular genetic algorithm described in Chapter VIII is provided. The behavior of the approach is thoroughly analyzed.

Future research directions are focused on:

- devising a better feature vector, in order to better discriminate between highly similar individuals;
- integration of the branch-and-bound paradigm by evolving embryos instead of chromosomes;
- parameter tuning.

Chapter V

Conclusions and future work

The focus of the thesis is on the development of new and powerful methods, based on the hybridization of Evolutionary Algorithms, in particular Genetic Algorithms. The hybridization, both integrative and collaborative, of Genetic Algorithms with elements from other optimization methods, both exact and heuristic, is the starting point in addressing the problem of the balance between exploration and exploitation of the search space. This balance is one of the key points for the success of metaheuristics in finding good solutions for the tackled problems.

The introductory chapters of the thesis describe the Evolutionary Computation research field. A greater interest is given to the Evolutionary Algorithms area and to Genetic Algorithms, especially. Some advanced approaches, that address the balance between exploration and exploitation are described. The most popular existing methodologies of hybridizing Evolutionary Algorithms are presented and examples of algorithms and their achievements are given.

The performance of the proposed methods is analyzed in the context of solving two difficult problems: the Assembly Line Balancing Problem and the optimization of variable order in Reduced Ordered Binary Decision Diagrams. Both problems are formulated and the current state of the research literature is presented.

The first proposed method is a hybridization between a genetic algorithm and elements from the exact optimization method Branch-and-Bound. The genetic algorithm evolves, for a period of time, sets of solution representations, rather than single solutions. The search is more explorative in the beginning of the evolution process, by identifying promising regions, and becomes more and more focused as the algorithm converges, by intensifying the search in the high quality found local optima.

The second proposed method is based on the Predator-Prey Cellular Genetic Algorithm and achieves to implicitly integrate the idea of niching, by organizing the population in ”pockets” corresponding to different regions of the search space. Genetic material exchange between high quality parallel evolving niches intensify the search in the corresponding regions of the search space.

The last method, called Segregative Genetic Algorithm, organizes the search process such that subpopulations corresponding to different regions of the search space are explicitly created and maintained. The exploration is achieved by organizing large quantities of individuals in subpopulations focused on niches in the search space. The search is intensified in high quality regions by evolving the corresponding subpopulations by means of a basic genetic algorithm. A Tabu Search inspired memory is used to avoid the redundant exploration of the search space.

All proposed hybrid methods have as basic component a genetic algorithm and integrate elements from other optimization methods, such as: Branch-and-Bound, Tabu Search, clustering techniques, self organizing Kohonen networks. The success of some of the methods is also dependent on a measure of similarity between chromosomes based on permutation or order representation. Simple distance measures between such chromosomes are proposed and analyzed.

All methods are benchmarked against classic test problem instances and are compared with state-of-the-art algorithms in literature. Extensive experimental evaluations give a better insight on how the methods work and prove their efficiency in terms of quality and stability of the solution. In-depth comparisons between the hybrid approaches and basic ingredients used in the hybridization are presented in order to prove the advantages of the hybridization process.

Constant concerns are also the degree of generality of the methods and the efficient use of computing resources. All algorithms are quite general and can be easily adapted to solve other optimization problems, given a few simple prerequisites. The proposed Cellular Genetic Algorithm and Segregative Genetic Algorithm are highly efficient models for parallel and distributed computing. Both implementations are distributed. The experimental evaluations prove the very good usage of computing resources, in modern multi-core processors.

Future work can address most of the procedural choices made in different key points of each of the proposed algorithms. Many procedures are very simple and were used to prove the concept that the hybrid methods can offer significantly better results than the basic methods used in hybridization. Integrating more complex methods, some of them even problem specific, can further boost the quality of the results obtained by the proposed algorithms.

More complex and in-depth analysis is required to better understand the behavior of complex algorithms such as the Segregative Genetic Algorithm. The benefit of each component of the algorithms must be evaluated in order to estimate their importance and focus the research on the most promising features.

All parameters of the approaches are usually set to best known values in literature or to values obtained by means of a relatively limited statistical study. A more in-depth statistical analysis is required in order to better tune the parameters of each basic component in the context of the hybrid methods.

A study specific to the used test problems can be envisioned in order to obtain better feature vectors, that can better discriminate between highly similar chromosomes. The successful organization of the population in the cases of the Cellular And Segregative Genetic Algorithms is highly dependent on the quality of the similarity measure.

Other difficult optimization problems can be tackled in order to further prove the high efficiency of the proposed algorithms. Of high interest is multimodal optimization, which is very common in current real-world problems. The achievement of the niching effect, by means of the Cellular and Segregative Genetic Algorithms, make the extension to multi-modal optimization and this research direction natural.

Bibliography

[1] S. B. Akers. Binary Decision Diagrams. *IEEE Trans. Comput.*, 27(6):509–516, June 1978. ISSN 0018-9340. doi: 10.1109/TC.1978.1675141. URL `http://dx.doi.org/10.1109/TC.1978.1675141`.

[2] E. Alba and B. Dorronsoro. *Cellular Genetic Algorithms.* Springer Publishing Company, Incorporated, 1st edition, 2008. ISBN 0387776095, 9780387776095.

[3] T. Bäck, D. B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation.* Oxford University Press, 1997.

[4] T. Bäck, D. B. Fogel, and Z. Michalewicz. *Evolutionary Computation 1 + 2.* Institute of Physics Publishing, 2000.

[5] I. Baybars. A survey of exact algorithms for the simple assembly line balancing problem. *Manage. Sci.*, 32(8):900–932, August 1986. ISSN 0025-1909. URL `http://dl.acm.org/citation.cfm?id=9579.9580`.

[6] E. Beale and R. Small. Mixed integer programming by a branch and bound technique. In *Proceedings of the Third IFIP Congress*, pages 450–451, 1965.

[7] C. Becker and A. Scholl. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 168(3):694–715, 2006.

[8] C. Blum and A. Roli. *Hybrid Metaheuristics: An Introduction*, volume 114 of *Studies in Computational Intelligence.* Springer, 2008.

[9] O. Brudaru and B. Valmar. Genetic algorithm with embryonic chromosomes for assembly line balancing with fuzzy processing times. In *The 8th International Research/Expert Conference Trends in the Development of Machinery and Associated Technology (TMT)*, pages 875–878, Neum, Bosnia and Herzegovina, 15-19 September 2004.

[10] O. Brudaru and B. Valmar. Hybrid genetic algorithm / branch & bound technique to solve a time-dependent transportation problem. In B. Zupancic, R. Karba, and S. Blazic, editors, *Proceedings of 6th Eurosim Congress on Modelling and Simulation*, volume 2, pages 1–7, Ljubljana, Slovenia, 9-13 September 2007.

[11] O. Brudaru, E. Moraru, and A. Damaschin. Embryonic hybrid genetic algorithm for k-repairmen problem. In *MOPGP'10 – The 9th International Conference on Multiple Objective Programming and Goal Programming*, Sousse, Tunisia, 24-26 May 2010.

[12] O. Brudaru, D. Popovici, and C. Copaceanu. Cellular genetic algorithm with communicating grids for ALB problems. *Advances in Electrical and Computer Engineering*, 10(2):87–93, 2010.

[13] R. E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Trans. Comput.*, 35(8):677–691, August 1986. ISSN 0018-9340. doi: 10.1109/TC.1986.1676819. URL `http://dx.doi.org/10.1109/TC.1986.1676819`.

[14] E. Clarke, O. Grumberg, and D. Peled. *Model Checking.* MIT Press, 1999.

[15] R. Dakin. A tree search algorithm for mixed integer programming problems. *Computer Journal*, 8:250–255, 1965.

[16] K. De Jong. *An analysis of the behavior of a class of genetic adaptive systems.* PhD thesis, Ann Arbor, MI, USA, 1975.

[17] D. de Werra and A. Hertz. Tabu search techniques: A tutorial and an application to neural networks. *OR Spektrum*, pages 131–141, 1989.

[18] K. Deb and D. E. Goldberg. An investigation of niche and species formation in genetic function optimization. In *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 42–50, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc. ISBN 1-55860-066-3. URL `http://dl.acm.org/citation.cfm?id=645512.657099`.

[19] R. Drechsler and B. Becker. *Binary Decision Diagrams: Theory and Implementation.* Kluwer Academic Publishers, 1998.

[20] N. J Driebeck. An algorithm for the solution of mixed integer programming problems. *Management Science*, 12:576–587, 1966.

[21] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computating.* Springer, 2003.

[22] J. J. H. Forest, J. P. H. Hirst, and J. A. Tomlin. Practical solution of large mixed integer programming problems with umpire. *Management Science*, 20(5): 736–773, January 1974.

[23] I. Furdu and O. Brudaru. New hybrid genetic algorithm with adaptive operators and variability target for optimizing variable order in OBDD. *"Vasile Alecsandri" University of Bacău. Scientific Studies and Research. Series Mathematics and Informatics*, 19(2):241–256, 2009.

[24] S. Ghosh and R. J. Gagnon. A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research*, 27(4):637–670, 1989. doi: 10.1080/00207548908942574.

[25] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13:533–549, 1986.

[26] F. Glover. Tabu Search, Part I. *ORSA Journal on Computing*, 1:190–206, 1989.

[27] F. Glover. Tabu Search, Part II. *ORSA Journal on Computing*, 2:4–32, 1990.

[28] G. Hachtel and F. Somenzi. *Logic Synthesis and Verification Algorithms.* Kluwer Academic Publishers, 1996.

[29] J. H. Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, 1975.

[30] R. V. Johnson. Assembly line balancing algorithms: computation comparisons.

International Journal of Production Research, 19(3):277–287, 1981.

[31] M. D. Kilbridge and L. Wester. A review of analytical systems of line balancing. *Operations Research*, 10(5):626–638, September/October 1962.

[32] A. H. Land and A. G. Doig. An automatic method of solving discrete programming problems. *Econometrica*, 28(3):497–520, July 1960.

[33] C. Y. Lee. Representation of switching circuits by binary-decision programs. *Bell System Technical Journal*, 38:985–999, July 1959.

[34] A. A. Mastor. An experimental investigation and comparative evaluation of production line balancing techniques. *Management Science*, 16(11):728–746, July 1970.

[35] K. L. McMillan. *Symbolic Model Checking*. Kluwer Academic Publishers, 1993.

[36] C. Meinel and T. Theobald. *Algorithms and Data Structures in VLSI Design: OBDD Foundations and Applications*. Springer-Verlag, 1998.

[37] S. Minato. *Binary decision diagrams and applications for VLSI CAD*. Kluwer Academic Publishers, Norwell, MA, USA, 1996. ISBN 0-7923-9652-9.

[38] M. E. Salveson. The assembly line balancing problem. *The Journal of Industrial Engineering*, 6(3):18–25, 1955.

[39] A. Scholl and C. Becker. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*, 168(3):666–693, February 2006.

[40] A. Scholl and S. Voß. Simple assembly line balancing - heuristic approaches. *Journal of Heuristics*, 2(3):217–244, 1996.

[41] F. B. Talbot, J. H. Paterson, and W. V. Gehrlein. A comparative evaluation of heuristic line balancing techniques. *Management Science*, 32(4):430–454, April 1986. ISSN 0025-1909. doi: 10.1287/mnsc.32.4.430. URL `http://dx.doi.org/10.1287/mnsc.32.4.430`.

[42] J. A. Tomlin. An improved branch and bound method for integer programming. *Operations Research*, 19:1070–1075, 1971.

[43] T. S. Wee and M. J. Magazine. Assembly line balancing as generalized bin packing. *Operations Research Letters*, 1(2):56–58, April 1982.

[44] I. Wegener. *Branching Programs and Binary Decision Diagrams: Theory and Applications*. Monographs on Discrete Mathematics and Applications. SIAM, 2000.

[45] D. Whitley and T. Starkweather. GENITOR II.: a distributed genetic algorithm. *J. Exp. Theor. Artif. Intell.*, 2(3):189–214, October 1990. ISSN 0952-813X. doi: 10.1080/09528139008953723. URL `http://dx.doi.org/10.1080/09528139008953723`.

[46] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.