Using Genetic Algorithms for Large Scale Optimization of Assignment, Planning and Rescheduling Problems

IRFAN YOUNAS

Doctoral Thesis in Electronics and Computer Systems
Stockholm, Sweden 2014

© Irfan Younas, March 2014

Tryck: Universitetsservice US AB
Abstract

There has always been a need to solve real-life large-scale problems, such as efficiently allocating limited resources, and other complex and conflicting situations related to combinatorial optimization genre. A class of combinatorial optimization problems is NP-hard and, among many well-known, several of them are assignment, planning and rescheduling problems. Assignment problems can deal with optimal assignment of teams of collaborating agents; planning problems can be effects-based planning that search for promising plans to get desired end states with minimal cost; rescheduling problems can be multi-criteria optimization of rescheduling resources that modify existing original schedule. These large scale optimization problems are complex with intractable and highly complex search spaces. Currently, there are no known algorithms with polynomial time complexity, which can solve these problems. Genetic Algorithms have been successfully applied to solve many complex optimization problems but not to the specific problems mentioned above.

The aim of the research, presented in this thesis, is to use Genetic Algorithms for large scale optimization of assignment, planning and rescheduling problems. More specifically, the contributions of the thesis are to: (i) adapt existing and develop new efficient Genetic Algorithms to solve large scale assignment problems, and (ii) adapt existing Genetic Algorithms to solve large scale effects-based planning, and multi-objective rescheduling optimization problems.

In case of assignment, we solve a team assignment problem and investigate specific regions in a solution space for assignment problems with huge search spaces. For the team assignment, an existing Genetic Algorithm is adapted and applied for optimal assignment of tasks to teams of collaborating agents. The algorithm is scalable, stable, robust and produces a near optimal solution. The results of the team assignment problem show that the existing Genetic Algorithms are not efficient for optimal assignment of tasks to teams of agents. Hence, to solve larger instances of the problem efficiently, new Genetic Algorithms are developed with emphasis on the construction of crossover operators. Since teams assignment can be multi-criteria, a multi-objective model is constructed and two widely used multi-objective evolutionary algorithms are applied. Further, for the assignment problems with huge search spaces, an existing Genetic Algorithm is adapted to extract possible combinations of input parameters from a specified solution space region. To solve the large scale effects-based planning, a multi-objective optimization problem is formulated for the evaluation of operational plans and a multi-objective Genetic Algorithm is adapted and applied to the problem. The results show that the suggested algorithm is much more efficient than A*. For the rescheduling problem, a multi-objective optimization model for rescheduling of resources is proposed and a multi-objective Genetic Algorithm is adapted and applied to obtain the Pareto-optimal solutions.

The research presented in this thesis confirms that Genetic Algorithms can be used for large scale assignment, planning and rescheduling problems since they have shown to be suitable in solving these problems efficiently.
I would like to pay my deepest gratitude to my supervisors Professor Rassul Ayani and Associate Professor Anne Håkansson for their devotion, academic guidance and support. Professor Rassul Ayani dedicated his time and effort throughout my PhD studies. His broad knowledge, inspiring enthusiasm, excellent technical and supervisory skills, and kind guidance have all made me blossom from a junior researcher into a good mature researcher. Associate Professor Anne Håkansson gave her valuable time and effort during the last year of my studies. Her encouraging and constructive supervision, precious advices and recommendations helped a lot in improving the contents and presentation of this dissertation. I am honored to work under the supervision of both of these personalities and without them this thesis would not have been possible.

I express sincere thanks to my co-supervisor Associate Professor Johan Schubert (FOI) for his guidance, constructive critiques and valuable suggestions to shape the direction of this work. I would like to extend my sincere thankfulness to Professor Christian Schulte for his thoughtful comments and precious suggestions on design decisions related to data structures of the algorithms and their asymptotic complexity. Many thanks go to Dr. Farzad Kamrani for his help, support and enjoyable research collaboration throughout my PhD studies. I got a lot of motivation and boosting from him as a senior member of our research group. I would also like to thank Dr. Farshad Moradi (FOI) for fruitful discussions, suggestions and research collaboration. Furthermore, I am also grateful to Hirad Asadi (FOI) for cooperation resulted in a joint publication.

I am very grateful for Higher Education Commission of Pakistan for financing my PhD studies. I am also thankful to Mr. Thomas Sjöland, the head of department for Software and Computer Systems (SCS), for providing the good environment and office facilities for research.

I would also like to thank the staff at Software and Computer Systems at ICT. Thank you Ms. Sandra Gustavsson and Mrs. Marriane Hellmin for being available all the time for taking care of all the administrative matters.

Many thanks are also due to my colleagues and friends, too numerous to mention here, for their support and good company over the years. My special thanks go to my parents, brothers and sisters for their affection and encouragement throughout my life. Last but not least, I would like to thank my wife for her love, patience, care, and support throughout the whole journey of my PhD studies.
# Contents

<table>
<thead>
<tr>
<th>Contents</th>
<th>vii</th>
</tr>
</thead>
<tbody>
<tr>
<td>I 1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Research Problem</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Research Question</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Contributions and Results</td>
<td>5</td>
</tr>
<tr>
<td>1.4 Research Methodology</td>
<td>6</td>
</tr>
<tr>
<td>1.5 Structure of the Thesis</td>
<td>10</td>
</tr>
<tr>
<td>1.6 Thesis Author’s Contributions</td>
<td>10</td>
</tr>
<tr>
<td>2 Extended Background</td>
<td>13</td>
</tr>
<tr>
<td>2.1 Optimization Problems</td>
<td>13</td>
</tr>
<tr>
<td>2.2 Assignment Problems</td>
<td>16</td>
</tr>
<tr>
<td>2.3 Scheduling/Rescheduling and Planning Problems</td>
<td>17</td>
</tr>
<tr>
<td>2.4 Genetic Algorithms</td>
<td>18</td>
</tr>
<tr>
<td>3 Related Work</td>
<td>21</td>
</tr>
<tr>
<td>3.1 Literature Review of Assignment Problems</td>
<td>21</td>
</tr>
<tr>
<td>3.2 Literature Review of Scheduling/Rescheduling and Planning Problems</td>
<td>24</td>
</tr>
<tr>
<td>3.3 Literature Review of Genetic Algorithms</td>
<td>27</td>
</tr>
<tr>
<td>4 Thesis Contributions</td>
<td>31</td>
</tr>
<tr>
<td>4.1 Optimization of task assignment to collaborating agents</td>
<td>31</td>
</tr>
<tr>
<td>4.2 Efficient genetic algorithms for optimal assignment of tasks to teams of agents</td>
<td>32</td>
</tr>
<tr>
<td>4.3 Optimization of assignment of tasks to teams using multi-objective metaheuristics</td>
<td>33</td>
</tr>
<tr>
<td>4.4 Using Genetic Algorithms for Investigating Specific Regions of the Solution Space</td>
<td>34</td>
</tr>
<tr>
<td>4.5 Using genetic algorithms in effects-based planning</td>
<td>35</td>
</tr>
</tbody>
</table>
4.6 Solving battalion rescheduling problem using multi-objective genetic algorithms ........................................... 36

5 Conclusions and Future Work ........................................... 39
  5.1 Result of the research ............................................. 39
  5.2 Conclusions ..................................................... 40
  5.3 Evaluation ..................................................... 41
  5.4 Discussion ..................................................... 42
  5.5 Future Work ..................................................... 43

Bibliography .......................................................... 45

II ................................................................. 53

1 Paper 1 .......................................................... 55
  1.1 Introduction ..................................................... 57
  1.2 Problem Formulation ........................................... 59
  1.3 The Proposed GA Heuristic ................................... 60
  1.4 Implementation Overview ..................................... 63
  1.5 Experimental Results ......................................... 63
  1.6 Summary and Conclusions ..................................... 72

Bibliography .......................................................... 73

2 Paper 2 .......................................................... 75
  2.1 Introduction ..................................................... 77
  2.2 Problem Formulation ........................................... 80
  2.3 Genetic Algorithms for Assignment of Tasks to Teams of Agents ........................................... 82
  2.4 Implementation Overview ..................................... 83
  2.5 Analysis of the Genetic Algorithm ........................................... 84
  2.6 Genetic Crossover Operators ................................... 89
  2.7 Experimental Results ........................................... 99
  2.8 Summary and Conclusions ..................................... 111

Bibliography .......................................................... 115

3 Paper 3 .......................................................... 119
  3.1 Introduction ..................................................... 121
  3.2 Multi-objective model for collaborating teams assignment ........................................... 123
  3.3 Multi-objective Optimization Algorithms ................. 127
  3.4 Experimental Results ........................................... 129
  3.5 Conclusions ..................................................... 137

Bibliography .......................................................... 139
Part I
Chapter 1

Introduction

Solving real-life large problems of efficiently allocating limited resources, as well as
developing and managing complicated systems, and designing strategies for decision
makers to cope with different conflicting situations have always been demanding
for individuals and organizations [1]. To cope with these kind of hard problems,
optimization and operations research are the key disciplines that can facilitate de-
cision makers with quantitative decision-making and decision analysis by applying
scientific methods and technologies [1]. Optimization is the sub-discipline of ap-
plied mathematics, where the aim is either to minimize or maximize the output of a
function over a set of input variables subject to a set of constraints. Operations Re-
search, which is “problem and system oriented management approach”, deals with
the phases of problem formulation, mathematical modeling, and implementation of
the model solution in real life problem scenario. [1]

Many of the optimization problems are combinatorial optimization [3], where
the desired optimal solutions are certain combinations of variable settings (possibil-
ities) from a finite pool [4, 5]. Though some combinatorial optimization problems
have well-known polynomial time algorithms, there is a family of combinatorial
optimization problems which is NP-hard. The NP-hard [27] problems are very
complex with huge search spaces and no exact algorithm is known to solve them in
polynomial time.

Many of the real-life NP-hard optimization problems are assignment, planning
and rescheduling. Assignment Problems (APs) deal with optimal matching of two
or more sets, where each matching may have different weight. The classical Assignment
Problem (AP) is the assignment of $m$ agents to $n$ tasks such that each task
is performed by exactly one agent and the aim is to minimize the cost of assign-
ments. There is a variety of assignment problems, which need optimization in many
real-life scenarios. For instance, improving decision making for business processes
to increase product quality at a lower cost and time [2], and assigning soldiers to
military operations are some examples of assignment optimization problems.

Planning problems can be defined in different ways depending on the related
field. In Artificial Intelligence (AI), the planning problem can be defined as: “The AI planning problem is to find the sequence of actions that will take the planning world from a defined initial condition to a given goal state.” Instead of focusing on which activities need to be performed, scheduling problems are concerned with timely allocation of limited resources to a sequence of activities. Due to dynamic nature of the problems, new activities can appear over time and it may require modifications in the initial schedule, which is known as rescheduling. There are many real-life applications of planning and rescheduling such as rescheduling of operating rooms, nurses to shifts rescheduling, job shop rescheduling, human resources rescheduling, activities planning and rescheduling in military operations. These complex optimization problems require specific attention to support decision makers.

1.1 Research Problem

Solving large scale optimization problems have always been very challenging and demanding. In this thesis, the focus is on large-scale assignment, planning and rescheduling optimization problems. The class of assignment problems, focused in this thesis, deals with the optimal assignment of tasks to teams of collaborating agents. Assuming that each task has a set of requirements and each agent has a set of skills, the aim is to assign a team of agents to each task such that the overall benefit is maximized. Changing one of the team members may have a vital impact on the results. Since the changes affect the teams, collaborations within these teams make the problem complex by introducing non-linearity in the search space. Moreover for assignment problems with huge search spaces, instead of focusing on the whole solution space, decision makers may be more interested in specific region of the solution space.

The planning problem, considered in this thesis, is an effects-based planning, where the aim is to search for optimal plans to reach the desired end state such that the cost is minimized. Effects-based planning is to achieve a desired effect through a series of actions, where each action may have several alternative ways of performing that action. A plan is a sequence of action alternatives. Many possible plans need to be evaluated to find out the optimal plans.

The rescheduling problem, discussed in this thesis, is a multi-criteria optimization of rescheduling resources. Assuming the limited resources, on the emergence of new activities (tasks), rescheduling is done in such a way that (i) the number of performed tasks are maximized, (ii) the disruptions in the initial schedule are minimized, and (iii) the high priority tasks that are missed are also minimized. The considered cases are combinatorial optimization problems, where the search space is highly complex and finding exact optimal solutions is computationally intractable.

So far, there is not any existing algorithm, with polynomial time complexity, that can solve these optimization problems. Genetic Algorithms (GAs) are widely used meta-heuristics for sampling intractably large and highly complex
search spaces. The GAs have frequently been used for solving many assignment, planning and scheduling optimization problems [9–14]. To the best of our knowledge, GAs have not been used for the optimization of the class of assignment, planning and rescheduling problems discussed in this thesis. The GAs can be used to find those assignments which are mapped to a specific region of interest (subset of solution space).

1.2 Research Question

To find optimal solutions to large scale assignment, planning and rescheduling problems, we investigate how GAs can be adapted and applied to these optimization problems with intractably large and highly complex search spaces. More specifically, the thesis aims at answering the following research question:

**How to use Genetic Algorithms for large scale optimization of assignment, planning and rescheduling problems?**

To answer the research question, the research focuses on adapting already existing and developing new efficient GAs to solve large scale assignment problems of tasks applied to teams of collaborating agents. Moreover, the focus is on adapting and applying an existing GA to assignment problems with huge search spaces to find specific assignments, which are mapped onto a specific region of the solution space. In addition, already existing GAs are adapted to solve effects-based planning and rescheduling resources problems.

1.3 Contributions and Results

The contribution of this thesis is to illustrate how GAs can be used for large scale optimization of assignment, planning and rescheduling problems.

More specifically, the contributions are to:

(i) adapt existing and develop new efficient GAs to solve large scale assignment problems,

(ii) adapt existing GAs to solve large scale effects-based planning, and multi-objective rescheduling optimization problems.

For the part of the first contribution, a specific type of assignment problem, where tasks are assigned to teams of collaborating agents, is considered. Mathematical formulation of the optimization problem is presented and an existing GA is adapted and applied for finding a near optimal solution. An objective function of the suggested GA is formulated using a formal method for measuring the performance of teams. Simulation experiments are performed on large scale problems and the results are discussed with respect to accuracy, scalability, efficiency, robustness and stability. These results are presented and explained in the paper titled “Optimization of task assignment to collaborating agents” [59]. Further, to show how GAs to solve large scale team assignment problems, a multi-objective model
for the optimal assignment of teams to tasks is proposed and Non-dominated Sorting Genetic Algorithm-II (NSGA-II) and Strength Pareto Evolutionary Algorithm 2 (SPEA2), which are two widely used multi-objective algorithms, are adapted to solve large instances of the problem. In order to analyze the quality of the obtained results (Pareto-optimal solutions), simulation experiments are performed and evaluated. The algorithm and evaluation is presented in the paper titled “Optimization of assignment of tasks to teams using multi-objective metaheuristics” [61]. The results show that the existing GAs are not efficient in solving the larger problem instances of the team assignment. In order to develop efficient GAs for large scale team assignment problems, several widely used crossover operators are modified and two new crossover operators are designed to enhance the performance of GAs. The simulation experiments are performed and the results are evaluated by comparing them with the already existing crossover operators. The details of efficient GAs are given in the paper titled “Efficient genetic algorithms for optimal assignment of tasks to teams of agents” [60].

Another part of contribution 1 of the thesis is to show how an existing GA is adapted and applied to large scale assignment problems to find the specific combinations of input parameter values. The values are mapped onto a desired part of the solution space. An outsourcing example, with m subprojects each having n subcontractors, is studied in which decision makers may be interested in all the assignments (subprojects to subcontractors) that complete all the subprojects within given time and cost frame. The GA is used to investigate any part of the solution space. The paper “Using Genetic Algorithms for Investigating Specific Regions of the Solution Space” [67] illustrates how GA is used to find the mapping from the solution space to the input space.

For the second contribution, to show how GAs to solve large scale effects-based planning problems, a multi-objective optimization model is proposed and a multi-objective GA is adapted and applied to evaluate the operational plans within effects-based planning. The Pareto-optimal solutions obtained by the algorithm are compared with A* and the results are discussed with respect to efficiency and effectiveness. The adapted algorithm, results and evaluation is given in the paper titled “Using genetic algorithms in effects-based planning” [79]. Moreover, to show how GAs to solve large scale rescheduling problems, the battalion rescheduling is formulated as a multi-objective optimization problem and is solved using NSGA-II. The algorithm is implemented and the test cases for the evaluation of the experimental results are designed. The details are presented in the paper titled “Solving battalion rescheduling problem using multi-objective genetic algorithms” [68].

1.4 Research Methodology

The two basic and well-known research methodologies are quantitative and qualitative research methodologies [50–52]. The quantitative research method emphasizes on numbers handling large data sets and supports finding solutions by testing and
proves hypotheses [50, 52]. Research work which involves statistics, mathematical modeling and simulations should be carried out using quantitative research. On the other hand, qualitative research method is used to help researchers to reach a tentative hypothesis and theory by understanding people, their actions and environment using collected data from different means like observations, interviews and questionnaires [50, 52]. Depending on the nature of the project or problem, the research method should be selected. In order to get complete overview of the research, two methods can also be used as complement and the combined research method is known as triangulation [53].

In order to answer our research question “How to use Genetic Algorithms for large scale optimization of assignment, planning and rescheduling problems” quantitative research is conducted. The reason of choosing quantitative research is that we want to reach a conclusion by studying how existing GAs can be adapted and some new GAs can be developed to solve different combinatorial optimization problems provided in this thesis. In our research work, simulation experiments are performed and results are discussed and evaluated.

The main components of quantitative and qualitative research are philosophical assumptions, research methods, research approaches, research strategy/design, data collection and data analysis [50–52].

1.4.1 Philosophical Assumptions and Research Approach

A philosophical assumption is the starting point of research and the four common philosophical perspectives are positivism, realism, interpretivism, and criticalism [50, 54]. Positivism is the most dominant philosophical assumption underlying quantitative methodology. “Positivist researchers generally assume that reality is objectively given and can be described by measurable properties, which are independent of the observer (researcher) and his or her instruments” [50]. In order to get predictive understanding of the phenomena, positivist studies generally test theories, most of the time using deductive approach. The reality is comprised of units and those units can be objectively classified into subjects and predicates. The research is classified positivist if there is evidence of hypothesis testing, quantifiable measures of variables, and phenomenon inference from the sample to a stated population [50]. Realism is the philosophical assumption, which assumes that the things are independent of the persons who are perceiving or thinking about things, and the things are known in reality [54]. To get convincing and reasonable facts and data, the phenomenon is observed. The realists analyze the collected data for knowledge. Interpretivism is the philosophical assumption underlying qualitative methodology in which the phenomenon is studied and explored in-depth to find out the meanings assigned to the phenomenon by people [50, 54].

To address the problems defined in this thesis, the choice of positivism philosophical assumption is the natural choice. The reason of choosing positivism is that in our research all the considered problems are defined objectively and the
variables involved are quantifiable. Instead of hypothesis, general idea of our research is tested and concluded by collecting data using simulation experiments.

The research approaches help in producing new knowledge or getting deep understanding and are used to draw conclusions. The most common research approaches are Deductive and Inductive. Deductive approach [50, 55, 56] works from more general to more specific and it is also often said to be top-down approach. It starts from a general theory or hypothesis and move towards more specific solution which can be evaluated to verify the theory or hypothesis. Inductive approach [50, 55, 56], on the other hand, moves from more specific observations to a general theory and it is known as bottom-up approach. To carry out the research in this thesis, the deductive approach is used. The research is a top-down research, where the general theory is that GAs are suitable for solving large scale assignment, planning and rescheduling optimization problems. The observations are collected using experiments and the results of the collected data are evaluated to confirm the general idea.

1.4.2 Research Strategy/Design

The research strategies and designs are methodologies for carrying out the research. The quantitative research designs are experimental research, ex post factor research, case study, and surveys, and in qualitative research, the designs are action research, exploratory research, grounded theory, case study and surveys.

Experimental research design is concerned with the evaluation of the hypothesis or general ideas and provides correlation between dependent and independent variables. Managing control of all factors, which may affect the experimental results, it provides "cause and effect" - relationship between variables. [52]

Ex post factor research or after-the-fact-research method does not change or control the independent variables since data has been already collected before it is carried out [52]. After-the-fact means that investigation starts after the fact has been occurred without any researcher’s interference [57]. This research design finds back in time the causal factors which are plausible. Similar to experimental research, it also evaluates theories and provides correlation between variables but does not provide safeguards to make strong inferences [52, 58].

Case study research is a design in which a phenomenon is investigated empirically [52].

Surveys are descriptive research designs that describe those phenomenon, which are not directly observable, and examines the frequency and relationships between variables [52, 57].

In qualitative research, action research strategy and design is concerned with actions to improve the strategies, knowledge and problem solving techniques of people. In case of problematic situations, it helps by contributing to practical matters. Exploratory research strategy and design focuses on exploring all the possible relationships between involved variables and provides basis for general findings. It helps
in finding key variables and issues which assist in defining objectives. Grounded theory intends to discover or develop a theory based on collected data. [52]

In the thesis, experimental research is carried out. The reason for choosing this research design is that a general idea is used for the research, which in our case becomes the theory for the research. The theory is that GAs are suitable for solving large scale assignment, planning and rescheduling optimization problems. We formulate the optimization problems in the form of mathematical models. The modeling is a process of formalization and abstraction which gives solvable mathematical model and it reveals the important cause and effect relationships [1]. To solve the formulated optimization problems, existing GAs are adapted, modified and new efficient GAs are developed.

1.4.3 Data Collection and Data Analysis

Experiments, case study, and questionnaire are the most common data collection methods in quantitative research. Experiments collect a large data set for involved factors (variables). In case study, in-depth analysis of small number of participants is carried out. Questionnaire collects data using different types of questions. [52]

The qualitative data collection methods are case study, observations, interviews, language and text, and questionnaire [52]. Observations focus on culture and participation to observe behavior [52]. Language and text illustrate meanings of texts and interpret reports, conversations, dialogs and dissertations [52]. In our work, the synthetic input data, belonging to different distribution families, is generated. We generate data such that it is close to real scenarios. For data collection, we use experiments for our quantitative research. Simulation experiments are performed to collect large data sets, which are used to evaluate the accuracy and performance of the implemented algorithms.

The most widely used data analysis techniques for quantitative research methods are statistics and computational mathematics and for qualitative research methods, the most common are coding, grounded theory, and analytical induction. Coding transforms qualitative data into quantitative data by analyzing patterns of observations. Grounded theory and analytical induction comprises of iterations between data collection and data analysis methods. The iterative process terminates with a validated theory. [52] In order to evaluate the experimental results of the research presented in this thesis, statistics are used as far as data analysis techniques are concerned. Different statistical methods like mean, standard deviation, average deviation from the best solution or known optimal solution, and some other state-of-the art methods such as Hypervolume are used.

1.4.4 Quality Assurance

Quality assurance is to validate and verify the results of the conducted research [52]. In case of quantitative research with deductive approach, validity, reliability, replicability, and ethics should be studied and applied [52, 54]. For qualitative research
with inductive approach, validity, dependability, confirmability, transferability and ethics should be studied and applied [50, 52].

Validity is to confirm that the instruments are measuring what is supposed to be measured [51, 52]. Reliability is the consistency or stability of the measurements for every testing [51, 52]. Replicability refers to the possibility of repeating the same research and getting the same results [52, 56]. Ethics is concerned with moral principles from planning to reporting results of research studies [50, 52].

Considering deductive approach in our quantitative research, validity, reliability, replicability and ethics are considered. For validation, different tests are designed to check the accuracy and performance of the algorithms to get the desired results. For reliability, tests are performed which show that the results are stable because there is a very small deviation in results obtained in multiple runs. The results are also replicable because researchers can get almost the same results by repeating the same experiments. Due to stochastic nature of the problems and algorithms, multiple replications are performed to calculate mean and standard deviation. The research presented in this thesis follow moral principles from planning to reporting results as an ethics.

1.5 Structure of the Thesis

The thesis is organized as follows:

Chapter 2 addresses the optimization concepts, categories of optimization models, assignment problems, Scheduling/Rescheduling problems, planning problems and Genetic Algorithms (GAs).

Chapter 3 presents related work.

Chapter 4 provides overall picture of the contributions, a brief summary and the contributions of papers, which are included in part II.

Chapter 5 concludes the presented research along with discussion on future work.

Part II This part includes all the papers which contribute to the presented research.

1.6 Thesis Author’s Contributions

The thesis is based on the work conducted for the following articles:

IEEE Symposium on Computational Intelligence in Scheduling, Paris, France, April 2011.

The thesis author adapted an existing GA for finding near optimal solutions to the class of task assignment problems where each task is assigned to a team of collaborating agents. The author also implemented the algorithm, designed experiments for analyzing the quality of the obtained solutions, and performed test and evaluation of the proposed method.

**Paper 2:** Efficient genetic algorithms for optimal assignment of tasks to teams of agents. Irfan Younas, Farzad Kamrani, Christian Schulte, Rassul Ayani, and Johan Schubert. submitted, 2013. Some parts of the first 5 sections of this paper were presented at the IEEE Symposium on Computational Intelligence in Scheduling, Paris, France, April 2011.

The thesis author proposed efficient GAs for solving large scale problems of optimal assignment of tasks to teams of agents. The author suggested modifications to several well-known crossover operators by adding a shuffled repair list to them and show that their efficiency is enhanced for solving the presented assignment problem. Two new crossover operators, team-based and team-based shuffled list crossover operators are also introduced to solve the large scale team assignment problems efficiently. The author implemented the algorithms and performed tests to evaluate the performance of the designed crossover operators.

**Paper 3:** Optimization of assignment of tasks to teams using multi-objective meta-heuristics. Irfan Younas, Farzad Kamrani, and Rassul Ayani. A shorter version of this paper is published in Proceedings of the fifteenth annual conference on Genetic and evolutionary computation conference companion. ACM, 2013.

The thesis author formulated a multi-objective optimization model for the optimal assignment of tasks to teams of agents. In order to solve large scale problems, the author adapted NSGA-II and SPEA2, which are two widely used multi-objective evolutionary algorithms. The author implemented the algorithms, designed experiments and performed tests for analyzing the quality of the obtained solutions.


The thesis author adapted an existing GA for investigating any part of the solution space. The suggested method helps in finding the set of assignments that are mapped onto a given subset of the solution space (region of interest). The author implemented the algorithm, designed experiments and performed tests for evaluation.


The thesis author formulated the effects-based planning problem as a multi-objective optimization problem and adapted an existing GA for the evaluation of operational plans within effects-based planning. The author suggested evaluating the quality of the results obtained by the proposed method by comparing the results with A*. The author implemented the algorithm and performed evaluation to discuss the effectiveness and efficiency of the suggested algorithm.


The thesis author participated in the formulation of a multi-objective model for rescheduling human resources in a battalion. The author adapted NSGA-II to the large scale problems to obtain a set of Pareto-optimal solutions. The algorithm was implemented by the author. The test cases for the evaluation of the experimental results are designed and implemented by the thesis author.
Chapter 2

Extended Background

In this chapter, we present the extended background to the optimization problems, categories of optimization models, assignment problems, planning problems, rescheduling problems and genetic algorithms.

2.1 Optimization Problems

An optimization problem is the problem of finding the extreme values (best or worst solutions) of one or more functions (often called objective functions or cost functions) over a set of (decision) variables subject to a set of constraints [1, 16, 18].

An optimization problem can be mathematically represented as [17]:

\[
\begin{align*}
\text{Maximize/Minimize} & \quad f_m(x) \quad m = 1, \ldots, M \\
\text{subject to} & \quad g_j(x) \leq 0, \quad j = 1, \ldots, J \\
& \quad h_k(x) = 0, \quad k = 1, \ldots, K \\
& \quad x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, \ldots, n.
\end{align*}
\]  

(2.1)

The number of objective functions are represented by \( M \) and a solution \( x \) is a vector of \( n \) decision variables such that \( x \in \mathbb{R}^n \). In case of single variable optimization \( n = 1 \), which is known as one-dimensional optimization. The constraints and variable bounds determine the feasible region of the solution space \( S \) [17].

Next subsections describe the categories of optimization models.

2.1.1 Function or Trial-and-Error

In function optimization, mathematical model is built, which describes the objective function or cost function of the problem. The optimal solution of the problem is dependent on the mathematical formula. On the other hand, trial-and-error optimization tries to find the optimal solution by adjusting the values of given input variables without knowing much about the process involved. [8, 20]
2.1.2 Single or Multi-Objective

Single Objective Optimization: If there is one objective function ($M = 1$) in equation (2.1), the problem is single objective optimization [18] problem. For example in travelling salesman problem (TSP) [21], given a list of cities and distance between pairs of cities, the aim is to find the shortest possible route such that each city is visited once and we return to the origin city. In this problem the objective is to minimize the length of the tour and constraints are that every city should be visited exactly once and return to the origin.

Multi-Objective Optimization: Multi-Objective optimization problem deals with two or more objective functions, which are often conflicting [18]. General mathematical model of multi-objective problems is shown in equation (2.1) with $M > 1$. The objective function vectors in multi-objective optimization belong to a multi-dimensional objective space $\mathbb{R}^M$ [17]. In these optimization problems, there are usually more than one optimal solutions and these solutions can be defined from a mathematical concept of partial ordering [17]. The solutions in multi-objective optimization are compared using dominance relationship. A solution $x$ dominates another solution $y$ if $x$ is no worse than $y$ in all objectives and $x$ is strictly better than $y$ in at least one of the objectives [17–19]. A solution $x$ is said to be Pareto-optimal [17, 18] if there is no other solution $y \in S$ that dominates $x$. Considering a given set of solutions in Fig. 2.1, the points which are not dominated by any other member of the solution space are non-dominated solutions. Solution space is the set of all possible outcomes with regard to the objective functions. A non-dominated solution is one in which it is impossible to improve an objective without worsening at least one other objective. In Fig. 2.1, the points 1, 3, 6 and 9 are Pareto-optimal solutions.

![Figure 2.1: A set of points and non-dominated front](image-url)
2.1.3 Static or Dynamic

Static optimization problems are the problems where the output does not change with time [8, 20]. On the other hand, in dynamic optimization problems, the output is a function of time [8, 20]. In other words, at different times and circumstances the output can be different. For example, if we want to travel from destination $A$ to destination $B$ and we want to follow the best route. There can be several routes from $A$ to $B$ and from distance point of view the problem is static optimization problem where the shortest route $R$ will be the optimal solution. But suppose in different times of the day the route $R$ may be crowded or not suitable and our objective is to minimize the traveling time. Considering the traveling time as an objective, the problem is dynamic and the optimal solution can be different during different times of the day. [8]

2.1.4 Continuous or Discrete

Optimization problems can also be categorized into discrete or continuous based on the variables. Continuous variables have an infinite number of possible values, while discrete variables have a finite number of possible values [8, 20]. Discrete optimization is also known as combinatorial optimization [3]. In combinatorial optimization problems, the desired optimal solution is a certain combination of variable settings from the finite pool of possibilities (variable settings). Most of the real-world optimization problems are combinatorial optimization problems. Some of the examples of combinatorial optimization problems, which can be solved using well-known polynomial-time algorithms are metroid, matching, shortest path, and spanning trees [4]. There is also a class of combinatorial optimization problems, where the search space becomes intractable even for moderate sized instances and many of them are NP-hard [27] combinatorial optimization problems. Some of the examples of combinatorial NP-hard optimization problems are generalized assignment problems (GAP) [9, 28], travelling salesman problems, nurse scheduling problems and the large scale assignment, planning and rescheduling problems discussed in this thesis. For these kinds of problems, there is no known polynomial-time algorithm. Meta-heuristics [22] play an important role in sampling such intractably large and highly complex search spaces.

2.1.5 Constrained or Unconstrained

Unconstrained optimization problems are only concerned with the objective function to be optimized without any restrictions on the problem variables [8, 20]. For example, a firm wants to outsource a project to a subcontractor who maximizes the quality of the projects. On the other hand, constrained optimization problems optimize the objective function subject to some restrictions (constraints) on the involved variables [8, 20]. For instance, the firm maximizing the quality will be usually subject to a cost constraint.
2.1.6 Linear or Non-linear

In Linear optimization problems, the objective functions and constraints are linear [8, 23]. Due to linear nature of the objective functions, the optimum is always located at the boundaries of feasible area, which is defined by constraints and local optimum is also a global optimum. Non-linear optimization problems are those problems in which one or more constraints or the objective function are non-linear [8, 23]. Non-linear means that the output of the function is not directly proportional to the input. The optimum of non-linear problems is not necessarily located at the boundaries of the feasible region, it can also be interior of the region. Further, a local optimum is not necessarily to be the global optimum. Non-linear optimization problems are harder than linear problems. Most of the real-world optimization problems are non-linear due to the nature of the physical systems [24].

2.2 Assignment Problems

The problem of optimally matching (assigning) the elements of two sets (usually called tasks and agents), where each matching may have a different weight (cost) is known as the Assignment Problem [29] and hereafter will be referred to as the AP. The problems belonging to the AP class are combinatorial optimization problems. In mathematics, an assignment can be described in different ways, for instance an assignment can be viewed as a bijective mapping between two finite sets and flows in networks. For details of these different ways of describing assignment, refer to [29].

The classic AP assigns $m$ agents to $n$ tasks, where each task is assigned to exactly one agent and the objective to minimize the cost of assignments. Basically it is one-to-one mapping between agents and tasks and the real world examples are assigning agents (workers) to machines, tasks to workers, and tasks to machines [30]. The AP has many real life applications e.g., resource scheduling, assigning developers to software projects, jobs to computers in a network, military personnel to operations, and many more. The classic AP is discussed in most of the introductory books on operations research and management sciences [30]. The mathematical model for the classic AP is as follows [9, 30]:

$$\begin{align*}
\text{Minimize} \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\text{subject to} \quad & \sum_{i=1}^{m} x_{ij} = 1, \quad j = 1, \ldots, n, \\
& \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m, \\
& x_{ij} \in \{0, 1\},
\end{align*}$$

(2.2)
where \( c_{ij} \) is the cost of assigning agent \( i \) to task \( j \), \( x_{ij} = 1 \) if agent \( i \) is assigned to task \( j \), and 0 otherwise. The constraints mean that each task is performed by a single agent and that no agent is assigned to more than one task.

The naïve approach to solve the AP by comparing all possible assignments of tasks to agents is computationally infeasible due to the combinatorial explosion [25]. To illustrate the size of the problems that arises from seemingly small problems, consider the number of combinations for assigning 60 tasks to the same number of agents, which is \( 60! \simeq 8.3 \times 10^{81} \). This number is 83 times the number of atoms in the visible universe, which is believed to be about \( 10^{80} \). Increasing the size of the problem to 102, the number of combinations will be \( 102! \simeq 9.6 \times 10^{161} \) that is 100 times the number of atoms in a hypothetical universe in which each atom is replaced by a complete universe.

Several flavors of the AP have been discussed in the literature [30]. For instance, models with multiple tasks per agent, multiple agents per task, and many multi-dimensional assignment problems with the matching of members of more than two sets [30]. Different variations of the AP have varying degrees of complexity. Unlike the classic AP, some of them like Generalized Assignment Problem (GAP) [9, 30] are NP-hard combinatorial optimization problem and they do not have a polynomial time solution. The problems, which belong to NP-hard family, require heuristic approaches [26]. In related work chapter in section 3.1, we describe different variations of the AP in detail.

### 2.3 Scheduling/Rescheduling and Planning Problems

**Scheduling** is “the allocation of resources over time to perform a collection of tasks” [33]. The objective of scheduling is to optimize one or more performance criteria while allocating scarce resources over time to a sequence of activities [34]. Sequencing of activities and timing on resources are two important concerns in scheduling [35]. Schedules are divided into two categories based on the availability of the jobs before or after the creation of schedules [37]. These two categories are static, and dynamic schedules [36, 37]. In static scheduling, specifications of all jobs (sequencing, number of activities, time) are identified prior to the creation of the schedule [37]. Once the schedule has been defined, these jobs do not change during the process and no new jobs are added to the schedule. One the other hand in dynamic scheduling, arrival of the jobs is dynamic and unpredictable activities can emerge at any time, which need to be scheduled [37]. Due to emergence of unpredictable activities, the original schedule need to be modified and this is also known as rescheduling [37].

Rescheduling plays an important role in many individual, organizational and economical decision making fields. There are many real-life applications of rescheduling e.g., rescheduling on break down of some vehicles on previously scheduled trips, nurses shifts rescheduling in case of emergency or other kind of disruptions, rescheduling of human resources in a battalion on the emergence of new activities,
operating room rescheduling on arrival of emergency surgeries or surgery time variability, job shop and flow shop rescheduling. Rescheduling decisions can have direct impact on different criteria like cost, quality, time, and security.

Planning problems in general are concerned in determining which activities need to be performed [36]. The difference between scheduling and planning is that scheduling is concerned with the allocation of scarce resources over time to a sequence of activities and planning is to determine which subset of activities need to be performed to reach the goal [36]. The problems concerned with the selection of subset of activities, which are chosen from a set of given alternatives, are planning problems. Most of the real-life planning problems are combinatorial optimization problems. Some of the planning problems such as shortest path, and spanning tree problems can be solved using well-known polynomial-time algorithms but many of them are NP-hard e.g., traveling salesman problem, and the effects-based planning problem.

Different variations of the rescheduling and planning problems have varying degrees of complexity and most of them are NP-hard combinatorial optimization problems. For many of them, there are no known algorithms with polynomial time complexity and hence heuristic approaches are used to solve these intractable combinatorial problems within a reasonable time.

2.4 Genetic Algorithms

Genetic Algorithms (GAs) [6–8] are widely used meta-heuristics for optimization, which are based on evolutionary ideas of genetics and natural selection. GAs were first suggested by John Holland [38] during 1960s to 1970s and got popularity by their application to solve a complicated problem of gas-pipeline transmission by Goldberg [6]. “Genetic Algorithms are good at taking large, potentially huge search spaces and navigating them, looking for optimal combinations of things, solutions you might not otherwise find in a lifetime” [39]. The GAs work on the Darwin’s principle of “survival of the fittest” [40]. GAs are widely used in many engineering, scientific and business areas for sampling intractably large and highly complex search spaces [41]. They work with a population of individuals where each individual is represented by of a set of parameters known as genes. The set of all genes in an individual is called chromosome, which is basically a candidate solution.

There are two types of well-known Genetic Algorithm (GA) models: the steady state (also known as incremental), and the generational [44]. In steady state GA, normally, a single offspring is generated in each iteration and inserted into the existing population [42, 44]. Throughout the process to keep the population size fixed, the replacement strategy can be either to remove the oldest individual or the worst individual of the population. The generational GAs create new offsprings using members of old population and all new offsprings will be the new population and the old population is discarded [43, 44]. This new population of the generated offsprings will become old population in the next generation and so on. Vavak and
Fogarty [44] performed different set of experiments and showed that steady state GAs outperform generational GAs.

The basic steps of a steady state GA are as follows [9]:

1. First step in any GA is to devise a suitable representation scheme to represent chromosomes which are basically candidate solutions for the considered problem. There are different types of representation encoding schemes like binary, permutation, and value encoding. Depending on the nature of the problem, the suitable representation scheme can be selected.

2. Generate a random population $P$ consisting of $N$ candidate solutions where $N$ is the decided size of the population.

3. Evaluate fitness for each solution $x \in P$ using the designed fitness function $f(x)$.

4. Select two parent solutions for reproduction. There are a number of different selection schemes and most commonly used are tournament selection, rank-based selection, and proportionate reproduction [45].

5. An offspring is generated by applying a crossover operator to the two selected parent solutions. Depending on the nature of the problem, there exist different kinds of crossover operators in the literature. The crossover operation makes the new offspring to inherit partial characteristics from its parent solutions and it is useful in evolution.

6. Mutation with some probability $p_m$ is applied to the newly generated offspring. Mutation prevents the loss of diversity [38]. It is helpful to traverse different regions of search space and thus escaping local minima/maxima.

7. The generated offspring should be inserted in the population if it is not already found. In order to keep the size of the population fixed, we need to either remove the oldest individual or the worst individual of the population.

8. The steps 2 to 6 are performed repeatedly until some termination condition is fulfilled.

GAs are also widely used for multi-objective optimization problems using different approaches such as combining multiple objectives into a single objective (aggregation approach), Pareto-based, and population-based non-Pareto approach. In the aggregation-based approach multiple objectives are combined to form a single objective function using aggregation operators such as weighted sum. The advantage of this approach is that there exists single optimal solution at the end and there is no need to interact with the decision maker. The problem with this approach is that decision makers do not always know how to combine the objective functions and different objective function can have different importance. In the
population-based non-Pareto approach the main population is divided into mul-
tiple sub-populations and for each sub-population, a separate objective function
is optimized. The third and the most widely used approach is the Pareto-based
approach, which is also known as dominance-based approach. In Pareto-based ap-
proach the evolved candidate solutions are evaluated using dominance relationship.
The solutions which are non-dominated are considered to be the fittest one and they
are called Pareto-optimal solutions. Non-dominated Sorting Genetic Algorithm-II
(NSGA-II) and Strength Pareto Evolutionary Algorithm 2 (SPEA2), which are
two widely used multi-objective GAs, use the Pareto-based approach for solving
multi-objective optimization problems. [18, 46]

GAs are widely used in different domains such as robotics, machine learning,
control, scheduling, planning, engineering design, economics, and combinatorial
optimization [47, 48]. GAs have been applied to a variety of function optimization
problems and are effective in searching large and complex search spaces even with
multi-modality, discontinuity and multi-dimensionality [49].
Chapter 3

Related Work

This chapter presents literature review of assignment problems, planning problems, rescheduling problems and genetic algorithms.

3.1 Literature Review of Assignment Problems

Harold Kuhn [31] proposed the Hungarian method, which is the first algorithm to solve the classical AP in polynomial time and had a vital impact on combinatorial optimization [32].

In the classical AP, each task is performed by a single agent and each agent is assigned to exactly one task [30]. The slight modification of the original AP can assume that the number of tasks and agents are not equal. The modified version of the problem can be easily converted to the basic AP by adding dummy agents or tasks. In case of problems where the objective is to minimize the cost, the cost of dummy assignment $c_{ij}$ is set such that $c_{ij} > c_{\text{max}}$, where $c_{\text{max}}$ is the maximum cost of assignments. For a maximization problem, $c_{\text{max}} - c_{ij}$ is substituted for each $c_{ij}$ to reduce it to basic AP.

Besides these slight modifications, there are many variations of the AP with varying complexity. Pentico [30] presents an extensive survey on the variations of the assignment problems. The golden anniversary survey recognizes three main categories of AP, where each category discusses several variations [30]:

1. Models with at most one task per agent.

Some of the variations in this category are:

- The classical AP.
- Classical AP where not every agent is qualified to do every task.
- The $k$-cardinality AP, in which only $k$ tasks and agents are to be assigned, where $k$ is less than both number of tasks and agents.
• The semi-assignment problem, in which each task is assigned to several agents while each agent may only be involved in one task.
• Multi-criteria AP, in which the single objective is substituted by multiple decision criteria.

2. Models with multiple tasks per agent.

Some of the variations are:

• The generalized assignment problem (GAP), in which each task is assigned to one agent, but agents have different capacities that may be used to perform more than one task.
• The multiple resource GAP, where instead of having a capacity, each agent may have several limited resources.

3. Multi-dimensional assignment problems, focusing on matching the members of three or more sets.

Some types of multi-dimensional AP are:

• The planar three-dimensional AP, in which tasks are assigned to agents over a set of time periods, where during each time period each task is assigned to exactly one agent and each agent performs at most one task.
• The three-dimensional bottleneck AP, which is a straightforward generalization of the standard two-dimensional bottleneck AP, where the numbers of tasks, agents and machines are equal and the goal is to minimize the maximum of costs of assigning a job to an agent and a machine.
• Multi-period AP, which is a three-dimensional AP that discusses assigning agents (often staff members) to changing tasks over a time horizon and exists in different versions.

Many variations of the AP have more than one objective and can be categorized as multi-objective optimization problems. Chicano et al. [62] solve a multi-objective model of the Software Project Scheduling (SPS) [63]. Given a number of tasks with certain requirements, resources with a set of skills need to be assigned such that cost and time of the project are minimized. A group of resources can be assigned to each task but there is no collaborations within groups. In solving the proposed SPS model, the authors compare performance of different multi-objective evolutionary algorithms. Fan et al. [64] suggest a multi-objective model for selecting R&D teams and the considered objectives are based on the individual and collaborative information of the members. In their paper, collaboration between pairs is considered and bi-objective programming model is solved by slightly modifying NSGA-II. Feng et al. [65] extend the problem of member selection of R&D teams to three objectives: (i) individual performance (ii) interior organizational collaborative performance (iii) exterior organizational collaborative performance. The collaboration is again between pairs only. The proposed tri-objective 0-1 programming
model is solved using an Improved Non-dominated Sorting Genetic Algorithm-II (INSGA-II). Gueorguiev et al. [66] focus on robustness of software projects using a search-based method. Robustness and completion time of the projects are two conflicting criteria which are considered while assigning work packages to teams of developers. The bi-objective programming model of work packages assignment is solved using SPEA2, which is a well-known multi-objective evolutionary algorithm.

The AP, as defined in this thesis (see Younas et al. [59] for more details), is different from all the presented APs in the literature. The groups of agents are assigned to the given tasks such that the agents collaborate within teams. However, the problem belongs to the first category of APs described in [30] because each agent can be assigned to one task only. Among various types of APs under category one, it is more close to semi-assignment problems, where a task is performed by a group of agents. Nevertheless, due to collaboration between agents, the presented problem in this thesis is much more complex. A GA is suggested for finding a near optimal solution to this class of task assignment problems. To the best of our knowledge, the defined problem has not been discussed in the literature due to the unavailability of methods for evaluating the performance of teams. To formulate the objective function of the suggested GA, a recently developed formal method is used to evaluate the performance of a team of agents.

In order to solve large instances of the problem efficiently, in [60] we design efficient GAs by focusing on the construction of several different crossover operators. We modify some of the existing crossover operators by adding a shuffled repair list to them and show that these modified operators outperform existing crossover operators for near-optimal assignment of agents to teams. Further, we also design two new crossover operators, team-based and team-based shuffled list crossover operators which are efficient in solving large scale instances of this class of assignment problems.

In the paper [61], we address the team assignment problem as a multi-objective optimization problem, in which two objectives: (i) cost and (ii) quality are considered. A multi-objective optimization model is proposed, which considers cost and quality as two conflicting objectives. The quality is calculated by aggregating the capabilities of assigned agents. The agents in teams collaborate with each other and help to improve skills of weaker agents. The non-linearity in the solution space, which is caused by collaboration of agents, makes it difficult to find the optimal solution. To solve the model, NSGA-II and SPEA2, which are two widely used multi-objective evolutionary algorithms, are suggested. To the best of our knowledge, the literature does not target multi-objective optimization problem of assignment of collaborating agents to tasks. Though, the existing work addresses some of the multi-objective optimization assignment problems, but they focus on different flavors of the problem. The flavor of the problem considered in this thesis deals with multi-objective optimal assignment of tasks to collaborating teams of agents. The problem is intractable due to the non-linearity in the solution space, which is introduced by intra-team collaboration of agents.
3.2 Literature Review of Scheduling/Rescheduling and Planning Problems

To the best of our knowledge, the literature does not address multi-objective optimization rescheduling problems similar to our battalion rescheduling problem presented in [68]. The existing work on scheduling and rescheduling problems is different in several aspects. In the literature, we find related work on a number of rescheduling problems e.g., operating rooms rescheduling, shift rescheduling of nurses, flight crew rescheduling, flow shop rescheduling, job shop rescheduling, and human resources rescheduling in a number of real-life scenarios.

Moz and Pato have proposed a number of techniques for solving the nurse rescheduling problem. Moz and Pato [69] propose an integer multicommodity flow model for nurses rescheduling and solve it using an integer linear program and a constructive heuristic. The service rendered by each nurse is represented as a commodity and there are 3 sets of nodes for shifts, source and destination nodes. Each node represents a nurse and the goal is to minimize the cost while flowing the rendered services through the network subject to requirements of nodes as far as demand and supply are concerned. The integer linear program finds the optimal solution but it takes more time than the constructive heuristic approach. Though heuristic approach is efficient as far as time is concerned but the solution found is not to be necessarily optimal one. Moz and Pato perform experiments and discuss optimality but they do not focus on number of disruptions to the initial schedule. In order to improve the algorithm, Moz and Pato [12] use a genetic algorithm to solve the problem. Along with the aim to minimize the changes in the original schedule, they also introduce a second objective into the fitness function. The second objective is to minimize the overtime, which is minimizing the difference between the number of scheduled duties and the number of performed duties. The proposed genetic algorithm improves the quality of the obtained solutions and is considered to be good for these kind of problems. Moz and Pato focus on methodology and algorithms to solve the nurse rescheduling problem without considering how the changes to the shifts affect the nurses. The preferences of the nurses are not considered in their models. Clark and Walker [70] present models and perform experiments for nurse scheduling and rescheduling considering nurses’ preferences. The initial schedule is created considering the preferences of the available nurses. The rescheduling is needed when the nurses are unavailable to perform their assigned shifts or where there are any kinds of changes in nursing cover requirements. The aim is to reschedule nurses while considering their preferences in a manner that changes as little as possible the initial schedule. The purpose is to minimize changes to the existing schedule as well as minimizing the total cost. Chicano et al. [62] apply evolutionary algorithms to solve multi-objective optimization problem of the Software Project Scheduling [63]. The goal is to assign suitable and skilled resources to the given number of tasks such that the cost and time are minimized. A number of multi-objective evolutionary algorithms are applied and compared.
Maenhout and Vanhoucke [14] solve nurse rostering problem, where the personal roster determines the work for each person. Rescheduling is necessary to cope with dynamic and unexpected events.

The job shop scheduling is also a well-known combinatorial optimization problem, which is NP-hard. The job shop scheduling considers a set of jobs to be performed on a given set of machines optimally subject to a set of constraints. Different methods like dispatching rules, artificial intelligence, mathematical techniques have been applied to solve job shop scheduling problems but most of them focus on static job shop scheduling and they do not consider random breakdown of machines or arrival of new jobs [14]. The flavor of the problem which deals with these dynamic events is known as dynamic job shop scheduling problem or job shop rescheduling. The job shop rescheduling is more complex than static model. In the literature, we find a large number of papers which focus on models and algorithms to solve job shop rescheduling problems. Vieira et al. [71] present a framework for understanding rescheduling research. The authors give definitions and a detailed survey of most applications of rescheduling manufacturing systems. Dominic et al. [72] combine different dispatching rules to solve dynamic job shop scheduling efficiently. Basically the mechanism to select next job to be performed from a set of available jobs follows the dispatching rules. Sabuncuoglu and Kizilisik [73] propose simulation-based scheduling system for dynamic and stochastic manufacturing environments. The work analyzes major scheduling issues and present several reactive scheduling policies, which are basically how and when to schedule. Sha and Liu [74] propose a model for due date assignments of jobs in a dynamic job shop environment. The model incorporates a data mining tool for mining the knowledge about due date assignments in job shop scheduling. Moratori et al. [75] propose match-up strategies for solving a real world rescheduling problem of integrating new rush orders into the current schedule. The rescheduling problem belongs to a printing company which is related to accommodate new orders by using idle times on machines optimally. Gholami and Zandieh [76] propose a method of integrating simulations into genetic algorithms to solve flexible job shop rescheduling problems in case of disruptions due to random breakdowns of machines. The model considers two objectives which are minimization of expected mean tardiness and minimization of expected makespan. Makespan is the total time needed to complete all jobs and tardiness (also referred as disruption cost) is the difference between due date and completion date of the job. Ouelhadj and Petrovic [77] defines dynamic scheduling and provides a survey of the state-of-the-art. The survey paper discusses and compares several heuristics, meta-heuristics, artificial intelligence-based techniques, multi-agent systems, and many other algorithms to solve dynamic scheduling problems.

The battalion rescheduling problem proposed in this thesis in [68] has some similarity with the nurse rescheduling problem in one of the objectives, which is to minimize the differences from the initial schedule. However, there are several differences in the proposed battalion rescheduling problem and the nurse rescheduling problem. Firstly, the proposed multi-objective battalion rescheduling problem...
considers tasks that need different types of personnel with different types and levels of skills. Secondly, new task with different set of requirements can arrive during any time of the year which needs to be performed optimally by the given number of personnel. Thirdly, the skills of personnel are not static and they are enhanced as the personnel attend military exercises, trainings and other courses. All these aspects make the multi-objective battalion rescheduling problem more challenging and difficult as compared to the nurse rescheduling problem.

The battalion rescheduling problem discussed in [68] is also similar to the job shop rescheduling problem in some aspects but the dynamic behavior of personnel skills in the battalion rescheduling problem makes it different from the job shop rescheduling problem. The battalion personnel attend military exercises, trainings and educational courses to enhance their competencies, which make the problem harder.

To the best of our knowledge, existing work does not address multi-objective optimization model of the planning problem similar to our effects-based planning model discussed in [79]. The effects-based planning (EBP) is a method used in military operational plans for developing effects and objectives by a series of activities. Operational plans are a set of actions or effects and each action may have several alternative ways of performing that action. All these alternatives together make up possible plans which need to be evaluated to find effective, efficient and robust plans. For the evaluation of operational plans within effects-based planning, Schubert et al. [80] propose simulation-based techniques. The distance function $f$, which is the sum of overall consequence of all performed actions as a distance from the initial state to the current state, denoted by $g$ and the distance of the current state to the desired end state, denoted by $h$, is minimized by A*. As the functions $g$ and $h$ are conflicting, and even if possible, it may not be appropriate to combine these two criteria into a single objective. Firstly, it is not clear how they should be compared and weighed against each other. Secondly, different decision-makers may have different preferences and prioritize $g$ and $h$ differently. To overcome this situation, we formulate the planning problem as a multi-objective optimization problem and solve it using multi-objective genetic algorithms. Many researchers [82–85] have applied GAs to solve different types of planning problems. Kanoh [82] propose a GA-based approach for dynamic route planning of car navigation systems. The path planning system uses real road maps which contain information about traffic signals, number of lanes and classification of roads. The experimental results show that the proposed GA-based approach for dynamic path planning for cars navigation outperform the Dijkstra algorithm and can help to improve car navigation devices. AL-Tahrarwa et al. [83] adapt a GA to solve mobile robots path planning problem in a static environment. The GA helps the controllable robots to find the optimal path between a starting and ending point in a grid environment. The optimal path is the shortest distance between source and the destination and the objective of the GA is to minimize the number of steps needed to move from the start to end point. Saadatseresht et al. [84] present a method for emergency evacuation planning using multi-objective evolutionary algorithms and geographical information system. The
suggested approach helps to displace evacuees from a dangerous place to a safer
place during emergency situations. Sasaki [85] use GAs to optimize current and
future health planning problems. The example considered in the paper is related
to the availability and response time of ambulances in case of emergency situation.
The method suggests optimal settings for locations of ambulances and the number
of ambulances needed in the future for a particular region. The experiments show
that the algorithm can help in survival of the patients by optimized fast response
time of the ambulances.

3.3 Literature Review of Genetic Algorithms

GAs have been successfully applied to a variety of real-life search and optimization
problems. They have been used in many engineering, scientific and business areas
for sampling intractably large and highly complex search spaces. After the birth of
the GA, Goldberg [6] brought fame to GAs by applying them to solve a complicated
problem of gas-pipeline transmission in 1989. Since then the GAs have been widely
used in many areas like robotics, engineering design, evolvable hardware, computer
gaming, encryption, finance, and many other real-life problems. A considerable
amount of literature e.g., [9–14, 76, 78, 82–85] focuses on solving several different
assignment, planning, scheduling and rescheduling problems using GAs.

The objective of the problem is to minimize the makespan subject to a set of con-
straints. The results indicate that GA-based techniques are suitable for solving
flow shop scheduling problems. Chu and Beasley [9] apply GA-based heuristic to
solve the generalized assignment problem. The proposed GA heuristic consists of a
local procedure for improvement and a pair of fitness and unfitness functions eval-
uations. The performance of the algorithm is tested on 84 standard test problems
and the results show that the GA can find optimal or near-optimal solution for all
the test problems in a reasonable time. Drezner [10] proposes various variants of
GAs for solving the quadratic assignment problem. In the creation of offspring, he
proposes a merging technique which is helpful in exploiting the properties of the
problem. The performance of the algorithm is tested on 29 different test problems
with different sizes and the results show that the algorithm performs very well in
finding the optimal solution.

Moz and Pato [12] propose a constructive heuristic and several versions of gen-
etic algorithms for solving the nurse rerostering problem. The suggested genetic
algorithms are based on specific operators and encoding schemes of sequence prob-
lems. The nurse rerostering is needed in those situations when the nurses are not
able to perform their assigned shifts and there are no more staff available in the
pool for assignment. The rerostering is done in such a way that the current schedule
is disturbed as minimum as possible. An individual in the population of the GA
is represented by two chromosomes which are basically permutations of nurses and
tasks. The proposed genetic algorithms improve the quality of the obtained solu-
tions and are considered to be good for these kind of problems. Gonçalves et al. [13] present a genetic algorithm for resource-constrained multi-project scheduling problem. The representation of chromosome is based on random keys and considering release dates, delay times and priorities, the schedule is generated using a heuristic. The experiments are performed on several test problems of different sizes and the computational results show that GA-based approach is suitable for the optimization of multi-project scheduling problem. Maenhout and Vanhoucke [14] solve the nurse rostering problem using a genetic algorithm. Due to the dynamic nature of the working environment, a set of unexpected events can emerge which can cause disturbances and infeasibilities in the initial schedule. In order to cope with this situation rescheduling is necessary. Experiments are performed on a well designed data set and the results of the proposed method are compared with already existing methods. Hao and Lin [78] present a random key-based genetic algorithm to solve multi-objective model of the job shop rescheduling problem.

The genetic algorithms have also been used to solve many planning problems [82–85]. A short description about these problems is given at the end of Section 3.2.

In order to apply GAs effectively, there is a need to design appropriate problem representation scheme, and intelligent selection of genetic operators (parents selection, crossover, mutation) and associated parameters settings. In the literature, we find some papers which focus on how to select these genetic operators and parameter settings which can affect the performance of the algorithm [86, 87]. There is no single best operator which is good for all different types of applications. The choice of these operators depends on the nature of the application and structure of the search space of the problem which is basically the search space properties, size and distribution of solutions. One type of operator can be good for one problem or application type and it may be not appropriate for other types of applications. For instance, the position-based and order-based crossover operators [88] show good performance for the one-machine total weighted tardiness problem. However, these two operators are not as good as the partially mapped crossover operator for the traveling salesman problem [89]. Kellegöz [88] applies a GA to one machine total weighted tardiness problem and compares the performance of several crossover operators. The studied crossover operators are one-point crossover, three flavors of two-point crossover [90], order-based crossover [91], position-based crossover, order crossover [92], cycle crossover [93], edge recombination crossover [94], partially mapped crossover [89], and linear order crossover. All these operators have been widely used in several combinatorial optimization problems.

In this thesis, we adapt existing and design new GAs for optimization of large scale assignment, planning and rescheduling problems. There are a lot of solutions in the literature for solving different assignment, planning and rescheduling problems, but to the best of our knowledge, the flavors of the problems discussed in this thesis have not been discussed in the literature. We also modify some of the existing genetic crossover operators and design two new crossover operators to solve the specific class of APs efficiently. In the considered APs, each task is assigned to
a team of agents and each task requires collaboration of all team members.
Chapter 4

Thesis Contributions

The research in this thesis uses GAs for optimization of large scale assignment, planning and rescheduling problems. The considered problems are combinatorial optimization problems with huge and intractable search spaces and cannot be solved using analytical methods. For large scale assignment problems, existing GAs are adapted to find optimal assignments of tasks to teams of collaborating agents for single and multi-objective flavors of the problem. Moreover, new GAs are developed to solve larger problem instances more efficiently. First four papers cover the assignment problems. Three papers provide solutions for optimal task assignment to collaborating teams of agents and the fourth paper shows how to use GAs to investigate predetermined solution space by searching for the combinations of input values that correspond to the region of interest. For the large scale planning and rescheduling problems, GAs are adapted and applied to effects-based planning, and multi-criteria resource rescheduling optimization problems. These problems are covered in the last two papers, paper five and six. The fifth paper shows that the adapted GA is more efficient than A* for the optimization of effects-based planning problem. The last paper concludes that GAs can be used for rescheduling of activities and assigning them to human resources.

4.1 Optimization of task assignment to collaborating agents


Summary and Results

This paper focuses on a specific class of task assignment problems where each task is assigned to a team of collaborating agents. It is assumed that each task has a set of requirements and each agent has a set of skills (capabilities) corresponding to those requirements. In the proposed model in this paper, the performance (gain)
of a team is a possibly non-linear function of its members, where the gain of the team may have major impact due to change of its members. The objective is to assign teams of agents to the tasks such that the gain is maximized. We adapt and apply a GA for finding a near optimal solution to this class of task assignment problems. The use of GAs is inspired by the fact that there are no known polynomial time algorithms to solve this class of assignment problems and GAs have been acknowledged to solve a range of combinatorial optimization problems. We believe that due to the difficulty of evaluating the performance of a team of agents, this class of assignment problems has not been considered in the literature. A recently developed formal method for measuring the performance of a team is used in this paper to formulate the objective function of the GA. The accuracy of the algorithm is verified by comparing the results for the cases, where agents work independently, with the Hungarian algorithm. A test is also performed to check the accuracy of the algorithm to solve the flavor of problem with agents collaboration within teams. In order to analyze the quality of the obtained solutions, we discuss and evaluate the experimental results with respect to scalability, efficiency, stability, and robustness.

Contributions

The contribution of this paper is to adapt an existing GA for finding a near optimal solution to the class of large scale assignment problems, where each tasks is assigned to a team of collaborating agents. The algorithm is implemented and experiments are designed to evaluate the quality of the results with respect to accuracy, stability, robustness, scalability and efficiency.

4.2 Efficient genetic algorithms for optimal assignment of tasks to teams of agents


Some parts of the first 5 sections of this paper were presented at the 2011 IEEE Symposium on Computational Intelligence in Scheduling, Paris, France, April 2011.

Summary and Results

In this paper, we focus on the development of efficient GAs for solving large scale problems of optimal assignment of tasks to teams of agents. We present a mathematical formulation of the problem and propose GAs to solve the problem, since there are no known algorithms which have polynomial time complexity for the considered assignment problems. Our investigation shows that GAs with one-point crossover are not efficient to solve large scale team assignment problems. The choice of genetic operators (selection, crossover, and mutation) and settings of associated parameters affect the performance of a GA. In order to design an efficient GA for
solving large scale problems of near-optimal assignment of tasks to collaborative teams of agents, we focus on the construction of crossover operators.

Considering synthetically generated data, experiments are performed and the results are compared and analyzed to verify the accuracy and efficiency of the GA and crossover operators. Several well-known crossover operators (one-point, two-point, three-point, position-based, order-based, and uniform) are tested to analyze their efficiency in finding near-optimal assignment of teams to tasks. We suggest modifications to these well-known crossover operators by adding a shuffled repair list to them, which enhances the performance of these operators in solving large scale instances of the presented assignment problem. We also propose two new efficient crossover operators, team-based and team-based shuffled list crossover operators. Our investigation shows that the position-based crossover with shuffled list is the best as far as performance is concerned. However, this crossover operator is sensitive to a large initial population size and it requires that the initial population size is chosen carefully. Experiments are also performed to fine tune the parameter settings of the best crossover operator.

**Contributions**

This paper contributes by proposing efficient GAs to solve large scale problems of optimal assignment of tasks to teams of agents. Several well-known crossover operators are modified by adding a shuffled repair list to them to enhance the performance of the GAs. Two new crossover operators, team-based and team-based shuffled list crossover operators are also introduced to solve large scale instances of the problem efficiently. The algorithms are implemented and the performance of the designed crossover operators is evaluated.

### 4.3 Optimization of assignment of tasks to teams using multi-objective metaheuristics


**Summary and Results**

In this paper, we propose a multi-objective model for the specific type of assignment problem, where teams of agents are assigned to the given number of tasks and agents performing a task collaborate with each other in a team. The objectives of the multi-objective optimization are to maximize the quality and minimize the cost. The quality of a task performed by a group of agents is the function of the capabilities of the participating agents and requirements of the task performed. The cost is a conflicting objective and the salary of assigning an agent to a task
is calculated as a function of the agent’s capabilities and the importance of these capabilities. The overall quality and the cost are calculated as the sum of the qualities and the costs of assigning all tasks to teams, respectively.

To solve the multi-objective model, we adapt two widely used multi-objective evolutionary algorithms, Non-dominated Sorting Genetic Algorithm-II (NSGA-II) and Strength Pareto Evolutionary Algorithm 2 (SPEA2). The accuracy of NSGA-II and SPEA2 in this context is verified by comparing the results for a small-sized problem, which can be solved using an enumerative algorithm. The experimental results show that both algorithms are accurate in finding all Pareto-optimal solutions for the small-sized problem. In order to compare the NSGA-II and SPEA2 algorithms, several experiments are performed on problems with varying sizes. In order to assess the performance of the algorithms, we adopt Hypervolume (HV) as a quality indicator. The experimental results show that for larger problem instances, NSGA-II generally outperforms SPEA2 with respect to the quality of the obtained solutions. NSGA-II has a higher HV value, that is NSGA-II provides Pareto-optimal solutions with better convergence and distribution of the obtained non-dominated solutions.

Considering NSGA-II as the superior algorithm, we further investigate and discuss the quality of the solutions obtained by NSGA-II on two large scale problem instances. The results show that with the increase of the size of the problem, we require a higher number of iterations to achieve the results with high HV values. However, the algorithm solves the larger problem instances in a reasonable time.

Contributions

Instead of treating the team assignment problem as a single objective, this paper focuses on a multi-objective optimization model for the optimal assignment of tasks to teams of agents. The contribution of the paper is to formulate a multi-objective optimization model of teams assignment, and adapt NSGA-II and SPEA2 to solve the larger instances of the proposed problem. The algorithm is implemented, and experiments and test cases are designed for analyzing the quality of the obtained Pareto-optimal solutions.

4.4 Using Genetic Algorithms for Investigating Specific Regions of the Solution Space

Summary and Results

This paper proposes a GA-based method for finding the combinations of the input parameter values that can lead to a specific region of the solution space. In many combinatorial optimization problems, understanding of the whole solution space is a very complex task. The Data Farming community has done a lot of work in this direction and has developed several design of experiments methods such as factorial designs, and Latin Hypercube designs for better understanding of the solution space. The Latin Hypercube designs are widely used methods for sampling and selecting a representative subset of the input space.

This paper looks at an outsourcing example from a different perspective such that it considers the case where the decision makers are interested in a specific region of the solution space and it investigates how to identify those assignments that are mapped into the region of interest. An outsourcing scenario is considered, where a company $A$ wants to outsource its $m$ subprojects such that each subproject has $n$ potential subcontractors. Project managers can be interested in finding those assignments that complete all the given subprojects within a cost and time frame. With $n$ potential subcontractors for each of the $m$ subprojects, the total number of possible assignments are $n^m$. If $n$ and $m$ are big numbers, it can lead to combinatorial explosion and in that case an exhaustive examination of all assignments is not feasible. This paper proposes an objective-based genetic algorithm (GA) for finding the set of assignments that are mapped onto a given subset of the solution space (region of interest). In contrast to design of experiments, the proposed algorithm starts from the solution space and tries to find the combinations of the input parameter values that can lead to a specific region of the solution space. The experimental results show that by redefining the region of interest, the GAs can be used to investigate any part of the solution space.

Contributions

Due to huge search spaces in large scale combinatorial optimization problems, decision makers may be interested in a set of assignments that are mapped onto a given subset of the solution space (region of interest). This paper contributes by adapting an existing GA to help decision makers in finding those assignments that correspond to a part of the solution space. The algorithm is implemented and experiments are designed to evaluate the suitability of the proposed algorithm in finding those combinations of the input space which lead to a specific region of the solution space.

4.5 Using genetic algorithms in effects-based planning

Summary and Results

In this paper, we formulate the effects-based planning problem as a multi-objective optimization problem and adapt an existing GA for the evaluation of operational plans within effects-based planning. There are two objectives of the problem: (i) minimizing the overall consequences of all performed actions as a distance from the initial state to the current state \( g \), and (ii) minimizing the distance from the current state to the desired end state \( h \). We adapt and apply Non-dominated Sorting Genetic Algorithm-II (NSGA-II) to solve the large scale multi-objective optimization problem.

This paper considers a large scale expeditionary operational scenario, where exhaustive evaluation of all plans is not feasible. The suggested algorithm simulates a subset of possible plans and presents the decision maker with a set of promising plans, which are capable of approaching the end state efficiently.

The experimental results of NSGA-II are compared with the results obtained by A* and the efficiency and effectiveness of the suggested algorithm are discussed. The experimental results show that NSGA-II is much more efficient than A* in minimizing the overall consequences of all performed actions (getting good \( g \)). On the contrary, A* is a little more effective in approaching the goal state (getting good \( h \)). Furthermore, Hypervolume (HV) is used to compare the two sets of non-dominated solutions obtained by the both algorithms. The HV of the non-dominated solutions obtained by NSGA-II and A* are 0.57 and 0.50 respectively, which shows that NSGA-II is better than A*.

Contributions

In this paper, the contribution is to adapt an existing GA for the evaluation of a large number of operational plans within effects-based planning. An effects-based planning problem is formulated as a multi-objective optimization problem and a GA is adapted for the evaluation of operational plans in large scale effects-based planning. To evaluate the quality of the obtained Pareto-optimal solutions obtained by the GA, the results are compared with A*. The algorithm is implemented and tests are performed to evaluate the effectiveness and efficiency of the suggested algorithm.

4.6 Solving battalion rescheduling problem using multi-objective genetic algorithms

Summary and Results

This paper proposes a novel model for rescheduling of human resources in a battalion to handle an essentially larger number of tasks with the same manpower resources. Due to unpredictable emergence of new activities, there is a need to utilize the available resources efficiently and effectively and that can lead to modification of the original schedule. We formulate the rescheduling problem as a multi-criteria optimization problem with three objectives: (i) maximizing the total number of performed tasks, (ii) minimizing the number of high-priority tasks that are missed, and (iii) minimizing the differences between the initial schedule and the updated schedule.

A multi-objective mathematical model is built. In order to solve the large scale optimization rescheduling problem, we apply Non-dominated Sorting Genetic Algorithm-II (NSGA-II). The experimental results, which are obtained using NSGA-II, are presented and discussed. To verify the accuracy of the algorithm, we consider a small problem with easy to verify solutions. Considering a realistic large problem instance for a battalion with 400 agents and 66 tasks, the computational results of rescheduling human resources on arrival of new activities are presented and discussed. The experimental results show that NSGA-II efficiently provides Pareto-optimal solutions for the proposed battalion rescheduling problem.

Contributions

The contribution of this paper is to adapt an existing GA to solve a large scale rescheduling problem. A multi-objective optimization model for rescheduling human resources in a battalion is proposed. A multi-objective evolutionary algorithm NSGA-II is adapted to the large scale problem to obtain a set of Pareto-optimal solutions. The algorithm is implemented and the test cases for the evaluation are designed.
Chapter 5

Conclusions and Future Work

The subject in this thesis is using GAs for different large scale combinatorial optimization, in particular: assignment, planning and rescheduling problems. These large scale optimization problems are intractable with large search spaces, and GAs are used to explore search spaces efficiently to search for near-optimal solutions. The conducted research shows that GAs can be either adapted or newly developed to solve these complex large scale optimization problems.

5.1 Result of the research

Using GAs for different large scale optimization problems is highly recommended. The GAs are shown to be suitable for these kinds of problems, which have been demonstrated in the different papers, presented in the research contribution. Specifically the results of the papers are:

- GAs are used for finding a near optimal solution to solve a task assignment problem with a maximized gain by assigning tasks to teams of collaborating agents.
- GAs are used to efficiently and effectively solve large instances of assignment problems by modifying and constructing new crossover operators for assigning tasks to groups of collaborating agents.
- Non-dominated Sorting Genetic Algorithm-II (NSGA-II) and Strength Pareto Evolutionary Algorithm 2 (SPEA2) are adopted to provide near Pareto-optimal solutions for larger multi-objective team assignment problems, with a minimized cost and maximized quality.
- An objective-based GA is applied for finding a set of assignments mapped onto a given subset of the solution space by randomly selecting an initial set of design points and identifying those candidates that are of interest by cross-over and mutation operators.
NSGA-II is adapted to solve a multi-objective (bi-objective) optimization planning problem, by proposing a genetic algorithm-based method for evaluating operational plans within effects-based planning, showing that NSGA-II is more efficient than A*.

NSGA-II is adapted to solve multi-criteria rescheduling problem, showing that this algorithm efficiently provides near Pareto-optimal solutions for the proposed rescheduling problem.

5.2 Conclusions

From successfully applying adapted GAs to large scale team assignment problems, we conclude that GAs can be used to solve assignment problems with accuracy, stability, robustness, and scalability. The considered team assignment deals with optimal assignment of tasks to cooperative agent teams such that the overall gain is maximized. One of the findings of the research, carried out in this thesis, is that one point crossover is not efficient enough to solve intractable large-sized problem instances of the team assignment with intra-teams interactions. Nonetheless, the performance of a GA in solving large scale team assignment problem can be enhanced by modifying existing and developing new crossover operators with an added shuffled list. This can be argued by the fact that the modified and developed crossover operators with shuffled list outperform the existing crossover operators without shuffled list. Another proof for showing that GAs can be used to solve large scale assignment problems, is effectively using GAs for multi-objective team assignment. The conclusion is that the algorithms can find all Pareto-optimal solutions for small problem instances. It is inferred that the multi-objective GA can solve the large-sized multi-objective team assignment instances in a reasonable time. Besides applying existing GAs, developing new efficient GAs to team assignment problems with intra-teams interactions, it is found that GAs cannot only solve large scale assignment problems but can also do it efficiently.

By satisfyingly adapting and applying existing GAs to investigate desired region of the solution space, it is concluded that GAs are suitable for searching those combinations of input parameter settings that are mapped on to the region of interest. By redefining the region of interest, GAs can be used to investigate any part of the solution space.

Moreover, from successfully applying the modified GAs to large scale effects-based planning and multi-objective rescheduling optimization problems, the conclusion is that GAs can be used to efficiently solve large scale planning and rescheduling problems. The conclusion for effects-based planning is drawn from the fact that the multi-objective GA has proven to be more efficient than commonly used search algorithms as far as the cost of finding optimal plans is concerned; the conclusion for multi-criteria rescheduling is made by the fact that the multi-objective GA efficiently provides the near Pareto-optimal solutions for the proposed rescheduling problem.
By solving optimal assignment of tasks to teams of collaborating agents, effects-based planning, and multi-criteria resource rescheduling, which belong to the intractable combinatorial optimization class, it is possible to claim the contribution in this thesis as can be used to solve some other similar combinatorial optimizations problems, which are NP-hard. By the research in this thesis, it is shown that GAs can be used for efficient optimal allocation of limited resources, and other complex and conflicting situations, which are related to combinatorial optimization genre, and hence, it is possible to solve real-life large scale problems. For example, GAs applied in the military case shows that GAs can support military and other defence organizations for optimal assignments of soldiers to operations, rescheduling resources, and optimal planning of military or other defence activities.

5.3 Evaluation

In order to verify that GAs can be used for large scale optimization of assignment, planning and rescheduling problems, we have evaluated the different GAs separately: (i) GAs are adapted, developed and applied to large scale assignment problems, and (ii) GAs are adapted and applied to large scale planning and rescheduling problems.

First, we evaluate the adapted and new efficient GAs applied to large scale assignment problems. By considering synthetic input data, the computational results of single and multi-objective flavors of a team assignment problem are evaluated using statistical methods. The accuracy of the algorithms is verified by comparing the computational results with manually calculated optimal solutions for smaller problems. By evaluating the quality of the obtained solutions as an average percentage deviation from the best solution found and by using standard deviation, it is found that the stability, robustness and scalability of the adapted algorithm for single objective optimal assignment of tasks to teams of synergetic agents is valid. Moreover, by evaluating the quality of non-dominated solutions obtained by NSGA-II and SPEA2, using average, standard deviation, and Hypervolume, the results show that, for larger instances of multi-objective team assignment problem, the quality of the solutions obtained by NSGA-II is generally better than SPEA2.

Further, considering two large-sized problem instances, the experiments search for the most efficient GA crossover operator, which can search optimal solution to the cooperative team assignment problem within limited time. The efficiency of the modified existing and newly developed crossover operators is compared with well-known existing crossover operators and results show that the modified and new crossover operators with a shuffled list perform better than all other operators without shuffled list, except for the position-based crossover with shuffled list that shows poor performance for larger initial population sizes. In order to show how GAs can be used to investigate a particular region of the solution space, an outsourcing example of projects is considered and computational results of experiments are evaluated. A few numerical examples are provided to demonstrate how
the adapted GA searches assignments that correspond to the region of interest. The results show that GA can be used to investigate any part of the solution space.

Secondly, to verify that GAs are suitable to solve large scale planning and rescheduling problems, the computational experiments for large scale effects-based planning and multi-criteria resource rescheduling are evaluated. For the effects-based planning, computational results of the NSGA-II for a large scale expeditionary operation scenario are compared with the results of A* using Hypervolume and mean values of the given objectives. The comparison results show that GAs are more efficient than common search algorithms, such as A*, in approaching the goal state with minimum cost. From evaluating the GA for the multi-criteria optimization problems of rescheduling human resources, the adapted algorithm is accurate. Considering a realistic large problem instance for a battalion, the evaluation shows that GAs efficiently provide the near Pareto-optimal solutions for the problem.

By examining the results of the two different evaluations of using GAs for different large scale combinatorial optimization problems, it shows that GAs can be used to solve many large scale optimization problems.

5.4 Discussion

Though the research, presented in this thesis, show that GAs are suitable in solving large scale assignment, planning and rescheduling problems, there are some limitations of the work. First of all, the result for each of the multi-objective optimization problem considered in this thesis is a set of Pareto-optimal solutions. This set may contain many solutions and the decision maker may need to identify some of them, which is not possible unless we rank the solutions. Secondly, the presented model for optimal assignment of tasks to teams of collaborating agents assumes that the agents collaborating within a team always enhance the performance of the team. There can be cases where agents' collaboration within a team can create negative synergy, which is not considered in the thesis.

Moreover, though for small problem instances, we verify that the algorithms used in this thesis find optimal solutions, still we can never be 100% sure that the obtained solutions for the large scale problems with intractable search spaces are truly optimal because the results of GA like other search heuristics are just approximation.

Another limitation is that, in order to solve large scale problems of optimal assignment of tasks to teams of agents, efficient crossover operators are developed. Position-based shuffled list crossover operators is found to be best as far as performance is concerned but it is sensitive to large initial population size. However, we have not investigated how to find an appropriate initial population size.

Further, an existing GA is adapted in the presented research to find the assignments that are mapped on to the specific region of the solution space. It is possible that some of the feasible assignments are not detected by the GA, which is certainly one of the limitations of our research.
5.5 Future Work

Due to broad scope and multiple aspects of large scale optimization, it is not possible to answer each and every aspect of the considered problems in this thesis. These are important problems to investigate, and this section provides possible future directions for the research.

In all multi-objective optimization problems, we have treated all the objectives equally. In future, we can incorporate decision maker’s preferences into multi-objective GAs, which can help in narrowing the search.

For each of the multi-objective optimization problems, considered in this thesis, the result is a set of Pareto-optimal solutions, which is just one aspect of multi-objective optimization. The obtained Pareto-optimal sets of the large scale assignment, planning and rescheduling problems usually consist of many solutions, which make it harder for decision maker to identify the best solution for implementation. Ranking the Pareto-optimal set is another interesting aspect of the problem.

There can be situations when GAs can take longer to solve large-sized instances of the considered problems effectively. In order to find optimal solutions quickly, parallel GAs can be developed and applied.

In order to suggest best algorithm for the problems, the results of GAs can be compared with other search techniques like local search, constraint programming, simulated annealing, tabu search, particle swarm optimization and combinations of these search heuristics.
Bibliography


