



AIN SHAMS UNIVERSITY
FACULTY OF ENGINEERING
MECHANICAL ENGINEERING

Resource Constrained Project Scheduling Using Genetic Algorithm

A Thesis Submitted in partial fulfillment of the requirements of the
Degree of M.Sc. in Mechanical Engineering

By

Amr Mohamed Taha Elsalhy

B.Sc. Mechanical Engineering

Supervised By

Prof. Dr. Nahed Sobhi AbdelNour

Dr. Ahmed Mahmoud Ali

Cairo- (2013)

STATEMENT

This thesis is submitted in the partial fulfillment of Master degree in Mechanical Engineering to Ain Shams University.

The author carried out the work included in this thesis, and no part of this thesis has been submitted for a degree or qualification at any other university.

Signature

Amr Mohamed Taha Elsalhy

Researcher Data

Name	Amr Mohamed Taha Elsalhy
Academic Degree	B.Sc. in Mechanical Engineering
Field of specialization	Mechatronics
University issued the degree	Ain Shams University
Date of issued degree	2006
Current Job	Operations Manager in PepsiCo Egypt

Summary of the Master Thesis

“Resource Constrained Project Scheduling Using Genetic Algorithm”

By

Amr Mohamed Elsalhy

Resources constrained project scheduling problem is very common in industry and one of the most complicated problems to be tackled.

The problem is considered as an NP-Hard optimization problem; especially, when a set of large number of activities need to be finished as soon as possible, subject to two sets of constraints (precedence constraints, and resources constraints).

Researchers have developed numerous scheduling methods and techniques to overcome the complex nature of this problem.

Genetic Algorithms are very promising approaches to solve this problem, in terms of the computational feasibility, and the quality of the solutions. However, the most common models of the genetic algorithms are difficult to be implemented in scheduling problems. On the other hand, using specific and proper design of the genetic algorithms can make the scheduling problems tractable.

The objective of this work is:

- To introduce genetic algorithm search model, to solve the resources constrained project scheduling problem, considering the make span minimization as the objective function, subject to the activities precedence relations, and the limited resources, as the problem constraints.
- To study the effect of the genetic algorithm model operators' parameters and combinations (crossover and mutation) on the solution quality, and test a new developed (Ranged crossover) operator.

In order to achieve the work objectives, the model was constructed using the activity list permutation based chromosomes representation, different crossover

operators are incorporated in the model (Single Point Crossover, Two points Crossover, Partially Mapped Crossover, and a new developed Ranged crossover operator), additionally two types of mutation operators are parts of the model operators), with a predefined number of generations, that considered as the model run exit criterion.

In order to run this model, an interface software program is developed, to utilize Genetic algorithm library developed by Malden Janković, to test 30 different PSPLib library instances, grouped in three sets, the 30 activities set (J30), the 60 activities set (J60), and the 120 activities set (J120).

The developed program enables the user to easily tailor the genetic algorithm parameters, which opens the room of opportunities for further research on the impact of different parameters.

Three sets of experiments are carried out on each instances set, with a total of 80 experiments are taken place, comparing four different crossover operators, with the two different mutation operators.

The proposed model is applied on the (J30, J60, J120) PSPLib benchmark project instances sets. Each project instance set includes 10 different project scenarios. The results compared by the benchmark solutions.

The results showed promising performance of the proposed model. The new developed Ranged crossover operator, accompanied with the swap mutation operator, achieved the benchmark best solution, and improved the GA solution by 7% in Set-1 test runs, 1% in Set-2, and the proposed GA model with the partially mapped crossover, achieved the GA best solution in Set-3 test runs.

Using the proposed model, the average computation time is Improved by 67%, 31%, and 96% in Experiment-set1, Experiment-set2, Experiment-set3 test runs respectively, compared by the benchmark best solutions average computation time, using Intel (R) Core™i5 CPU 2.67GHz machine processor.

Finally, the model proves applicability, and opens wider opportunities to build on for further research work.

Key Words: Genetic Algorithm, Resources Constrained Project Scheduling, Crossover, Mutation.

Acknowledgment

I would like to express my gratitude to Dr. Nahed Sobhi who directed me towards this interesting point and supported me by her experience throughout the whole work. Her supervision played an essential role for this work to be completed.

Not only do I owe her a lot in my research work, but in many things in my life as well.

Special thanks to Dr. Ahmed Ali, for his valuable support.

I am also grateful to my family for their continuous support in this work and all over my whole life.

List of Abbreviations:

CPM	Critical Path Method
DC-RCPSP	Discounted Cash-flow Resource Constrained Project Scheduling Problem
GA	Genetic Algorithm
GPR	General Precedence Relation
LFT	Latest Finish Time
LST	Latest Start Time
MMRCPSP	Multi-Mode Resource Constrained Project Scheduling Problem
MSLK	Minimum Slack Time
MTS	Most Total Successor
NPV-RCPSP	Net Present Value Resource Constrained Project Scheduling Problem
OX	Order Crossover
PMX	Partially Mapped Crossover
PSGS	Parallel Schedule Generation Scheme
PSPlib	Project Scheduling Problems Library
RCPSP	Resource Constrained Project Scheduling Problem
RL-RCPSP	Resource Leveling Resource Constrained Project Scheduling Problem
RX	Ranged Crossover
SFT	Shift Mutation
SGS	Schedule Generation Scheme
SPX	Single Point Crossover
SSGS	Serial Schedule Generation Scheme
SWM	Swap Mutation
TPX	Two Point Crossover

List of Tables

No.	Title	Page
1.1	Single-Mode Resource Constrained Project Scheduling Problem Classification	5
1.2	Multi-Mode Resource Constrained Project Scheduling Problem Classification	7
1.3	Serial SGS of Project schedule	15
1.4	Parallel SGS of Project schedule	17
2.1	Serial SGS of the schedule shown in figure 2.4	40
2.2	Project activities processing time	52
2.3	GA Parameters	54
3.1	The GA model parameters	58
4.1	Experiment-Set1_Group1_J301_1 GA runs results	69
4.2	Experiment-Set1_Group2_J301_2 GA runs results	70
4.3	Experiment-Set1_Group3_J301_3 GA runs results	71
4.4	Experiment-Set1_Group4_J301_4 GA runs results	72
4.5	Experiment-Set1_Group5_J301_5 GA runs results	73
4.6	Experiment-Set1_Group6_J301_6 GA runs results	74
4.7	Experiment-Set1_Group7_J301_7 GA runs results	75
4.8	Experiment-Set1_Group8_J301_8 GA runs results	76
4.9	Experiment-Set1_Group9_J301_9 GA runs results	77
4.10	Experiment-Set1_Group10_J301_10 GA runs results	78
4.11	Experiment-Set2_Group1_J601_1 GA runs results	80
4.12	Experiment-Set2_Group2_J601_2 GA runs results	81
4.13	Experiment-Set2_Group3_J601_3 GA runs results	82
4.14	Experiment-Set2_Group4_J601_4 GA runs results	83
4.15	Experiment-Set2_Group5_J601_5 GA runs results	84
4.16	Experiment-Set2_Group6_J601_6 GA runs results	85
4.17	Experiment-Set2_Group7_J601_7 GA runs results	86
4.18	Experiment-Set2_Group8_J601_8 GA runs results	87
4.19	Experiment-Set2_Group9_J601_9 GA runs results	88
4.20	Experiment-Set2_Group10_J601_10 GA runs results	89
4.21	Experiment-Set3_Group1_J120_2 GA runs results	90
4.22	Experiment-Set3_Group2_J120_3 GA runs results	91
4.23	Experiment-Set3_Group3_J120_4 GA runs results	92
4.24	Experiment-Set3_Group4_J120_5 GA runs results	93
4.25	Experiment-Set3_Group5_J120_6 GA runs results	94
4.26	Experiment-Set3_Group6_J120_7 GA runs results	95
4.27	Experiment-Set3_Group7_J120_8 GA runs results	96
4.28	Experiment-Set3_Group8_J120_9 GA runs results	97

4.29	Experiment-Set3_Group9_J120_10 GA runs results	98
4.30	Experiment-Set3_Group10_J121_1 GA runs results	99
4.31	Experiment-Set1 GA Performance	100
4.32	Experiment-Set1 GA runs computation time improvement vs. the benchmark solution test runs	102
4.33	Experiment-Set2 GA Performance	103
4.34	Experiment-Set2 GA runs computation time improvement vs. the benchmark solution test runs	104
4.35	Experiment-Set3 GA Performance	105
4.36	Experiment-Set3 GA runs computation time improvement vs. the benchmark solution test runs	107

List of Figures

No.	Title	Page
1.1	Project schedule	15
1.2	Project Network SGS	18
1.3	Project schedule SGS	19
1.4	Simple Single point Crossover	22
1.5	Modified Single point crossover	23
1.6	Simple two points crossover	24
1.7	Eligible chromosome produced after a two points crossover	25
1.8	Activities Precedence relationships	25
1.9	Partially Mapped Crossover PMX	26
1.10	Order crossover OX	26
1.11	Shift Mutation	27
1.12	Adjacent Swap Mutation	28
1.13	Random swap mutation	28
2.1	Activity On Node (AON) model diagram	36
2.2	Feasible schedule with makespan of 12 time-periods	37
2.3	Chromosome representation	38
2.4	Project schedule model	40
2.5	Single point crossover	42
2.6	Eligible chromosome produced after a two point crossover	44
2.7	Eligible chromosome produced after a partially mapped crossover	46
2.8	The Ranged Crossover (RX) algorithm flowchart	48
2.9	Shift Mutation	50
2.10	Random swap mutation	50
2.11	Project Network	51
2.12	Project Network CPM	52
2.13	Project Network configuration text file format	53
3.1	My Solver Flow Diagram	60
3.2	Diagram - Structure of the Genetic Algorithm Library	61
3.3	Diagram of the library Namespaces	63
3.4	My solver interface program instance upload form	64
3.5	The PSPLib RCPSP instance configuration	65
3.6	The PSPLib RCPSP instance configuration .txt file illustration	65
3.7	The solver analyzer form	66
3.8	The solver analyzer output forms	66
3.9	Proposed GA model	67
4.1	Experiment-Set1_Group1_J301_1 GA Makespan solution quality	69

4.2	Experiment-Set1_Group2_J301_2 GA Makespan solution quality	71
4.3	Experiment-Set1_Group3_J301_3 GA Makespan solution quality	72
4.4	Experiment-Set1_Group4_J301_4 GA Makespan solution quality	73
4.5	Experiment-Set1_Group5_J301_5 GA Makespan solution quality	74
4.6	Experiment-Set1_Group6_J301_6 GA Makespan solution quality	75
4.7	Experiment-Set1_Group7_J301_7 GA Makespan solution quality	76
4.8	Experiment-Set1_Group8_J301_8 GA Makespan solution quality	77
4.9	Experiment-Set1_Group9_J301_9 GA Makespan solution quality	78
4.10	Experiment-Set1_Group10_J301_10 GA Makespan solution quality	79
4.11	Experiment-Set2_Group1_J601_1 GA Makespan solution quality	80
4.12	Experiment-Set2_Group2_J601_2 GA Makespan solution quality	81
4.13	Experiment-Set2_Group3_J601_3 GA Makespan solution quality	82
4.14	Experiment-Set2_Group4_J601_4 GA Makespan solution quality	83
4.15	Experiment-Set2_Group5_J601_5 GA Makespan solution quality	84
4.16	Experiment-Set2_Group6_J601_6 GA Makespan solution quality	85
4.17	Experiment-Set2_Group7_J601_7 GA Makespan solution quality	86
4.18	Experiment-Set2_Group8_J601_8 GA Makespan solution quality	87
4.19	Experiment-Set2_Group9_J601_9 GA Makespan solution quality	88
4.20	Experiment-Set2_Group10_J601_10 GA Makespan solution quality	89
4.21	Experiment-Set3_Group1_J120_2 GA Makespan solution quality	91
4.22	Experiment-Set3_Group2_J120_3 GA Makespan solution quality	92
4.23	Experiment-Set3_Group3_J120_4 GA Makespan solution quality	93
4.24	Experiment-Set3_Group4_J120_5 GA Makespan solution quality	94
4.25	Experiment-Set3_Group5_J120_6 GA Makespan solution quality	95
4.26	Experiment-Set3_Group6_J120_7 GA Makespan solution quality	96
4.27	Experiment-Set3_Group7_J120_8 GA Makespan solution quality	97
4.28	Experiment-Set3_Group8_J120_9 GA Makespan solution quality	98
4.29	Experiment-Set3_Group9_J120_10 GA Makespan solution quality	99
4.30	Experiment-Set3_Group10_J121_1 GA Makespan solution quality	100
4.31	Experiment-Set1 GA performance	101
4.32	Experiment-Set1 GA runs computation time improvement vs. the bench mark best solution test runs	102
4.33	Experiment-Set2 GA performance	103
4.34	Experiment-Set2 GA runs computation time improvement vs. the bench mark best solution test runs	105
4.35	Experiment-Set2 GA performance	106
4.36	Experiment-Set2 GA runs computation time improvement vs. the bench mark best solution test runs	107

Table of Contents

Title	Page
Summary	i
Acknowledgment	iii
List of Abbreviations	iv
List of Tables	v
List of figures	vii
Table of Contents	ix
Introduction	1
Chapter1: Literature Review	3
1.1 Introduction	3
1.2 Project Scheduling Problems	3
1.3 Resource Constrained Project Scheduling Problem (RCPSP)	3
Definition	
1.3.1 Basic Single-Mode Resource Constrained Project Scheduling Problem	5
1.3.2 Multi-Mode Resource Constrained Project Scheduling Problem	6
1.3.3 Resource Constrained Project Scheduling Problem with non-regular objective functions	7
1.3.4 Stochastic Resource Constrained Project Scheduling Problem	8
1.3.5 Bin-packing-related Resource Constrained Project Scheduling Problem	9
1.4 Resource Constrained Project Scheduling Problem complexity	10
1.5 Introduction to Genetic Algorithm (GA)	10
1.6 Application of Genetic Algorithm (GA) in scheduling	11
1.7 Representation encoding of scheduling problem	12
1.7.1 Random key representation	13
1.7.2 Activity list permutation representation	18
1.7.3 Priority rule-based representation	21
1.8 Crossover Operators	21
1.8.1 Single Point Crossover	22
1.8.2 Two Points Crossover	23
1.8.3 Partially Mapped Crossover (PMX)	25
1.8.4 Order Crossover	26
1.9. Mutation Operator	27
1.9.1. Shift Mutation	27
1.9.2. Swap Mutation	28
1.10. How does GA works?	29

1.11. GA performance solving RCPSPs	29
1.12. Conclusion of the literature review	33
Chapter2. The problem definition and the proposed model	34
2.1. Introduction	34
2.2. Problem definition	34
2.3. Mathematical formulation of the problem	34
2.3.1. Model assumptions	35
2.3.2. Mathematical model	36
2.4. Genetic Algorithm outline	37
2.4.1. Chromosome representation	37
2.4.2. Serial Schedule Generation Scheme (SSGS) Decoding	38
2.4.3. Fitness function	40
2.4.4. Selection scheme	41
2.4.5. Crossover	41
2.4.6. Replacement Scheme	49
2.4.7. Mutation	49
2.4.8. Elitism	51
2.4.9. Termination criteria	51
2.4.10. Model verification and validation	51
Chapter3. Experimental Work	57
3.1. Objectives	57
3.2. Key GA Parameters	57
3.3. Experiments Set-1	58
3.4. Experiments Set-2	58
3.5. Experiments Set-3	59
3.6. Software	59
3.6.1. Genetic Algorithm library	60
3.6.3. My Solver	64
Chapter4. Results and Discussion	68
4.1. Introduction	68
4.2. Performance of the GA on Experiments Set-1	68
4.3. Performance of the GA on Experiments Set-2	79
4.4. Performance of the GA on Experiments Set-3	90
4.5. Experiments Set-1 results discussion	100
4.6. Experiments Set-2 results discussion	102
4.7. Experiments Set-3 results discussion	105
4.8. Discussions summary	108
Chapter5. Conclusions and For Future Work Recommendations	109
5.1. Conclusions	109
5.2. Future Work Recommendations	111
References	112
Appendix A: Experimental Work Data	117
Appendix B: Software Program Structure	138

Introduction

The research is using the genetic algorithm as one of the powerful optimization tools. The usage of this tool has tremendously increased in the last decade, as observed by the increase in the number of published papers, conferences, and workshops.

Based on the biological evolutionary nature of the Genetic Algorithm (GA), in contrast to the local search techniques, GA is working on a set of populations of solutions, instead of only one individual, which guarantees better search in the solution space, with an effective computation time.

Researchers utilized the genetic algorithm as a searching technique in solving the combinatorial optimization problems.

The resources constrained project scheduling problem (RCPSP) is considered as one of the most complex optimization problems, and as one of the great challenges to different solution heuristics.

By the increase in the number of the project activities, and the number of constraints, the exact solutions, and the exhaustive search enumeration techniques, take a very high computational time to solve the problem, and sometimes fail to find the optimum solution. That is why researchers paid a lot of attention trying to solve the RCPSP using the GA.

The objective of this work is:

To investigate the capabilities of the GA in solving the Resources constrained project scheduling problem by:

- Testing different combinations of the GA operators to solve global project instances sets, using a Genetic library through a developed software interface testing environment.
- Developing a new crossover operator, and testing its capability in improving the solution quality.

The thesis is divided into the following chapters and appendices:

Chapter 1: A literature review is presented, concerning the application of the GA in solving the RCPSP. Objectives of the research are then set.

Chapter 2: Problem definition, modeling and assumptions are defined; outlines of the GA model and the developed crossover operator are discussed.

Chapter 3: The experiments' sets that are carried out to meet the research objectives are discussed.

Chapter 4: Results obtained for solving the problems (presented in appendix A) are introduced, discussed and compared.

Chapter 5: The conclusions are summarized, and the recommendations for future work are suggested.

Appendix A: Demonstration of the experimental work data.

Appendix B: Demonstration of the software program main structure.