CHAPTER 6
SUMMARY AND CONCLUSION

6.1 SUMMARY OF THE PRESENT WORK

6.1.1 Ant Colony Optimisation Algorithm for CTV Minimisation
In this research study, investigations are made by developing metaheuristics for solving the permutation flowshop scheduling problem (PFSP) to meet the objectives considered. In the first part of the study, the problem of scheduling in the permutation flowshop with the objective of minimising completion-time variance (CTV) is considered and solved by making use of the ant-colony optimisation (ACO) algorithm. A modified ant-colony algorithm (MACO) with two implementations namely MACO-I and MACO-II have been developed (Chapter 3) and the performance evaluation of the proposed implementations of MACO is carried out by comparing the resultant solutions with the best solutions as reported in literature (Gajpal and Rajendran, 2006), and solutions obtained by implementing the algorithms proposed by researchers (Rameshkumar et al., 2011; and Ruiz and Stützle, 2007). It is found that the proposed implementations of ant-colony algorithm, especially MACO-II, yield promising and good results in respect of many problem instances, in comparison to the existing methods.

6.1.2 Simulated Annealing Algorithm for CTV Minimisation
Existing research reveals that the two metaheuristics, namely ant colony optimisation (ACO) and simulated annealing (SA) are competitive to each other and hence are more widely used for solving flowshop scheduling problems and it will be interesting to compare these two metaheuristic approaches in solving scheduling problems with a given objective. Hence, in the second part of this study (Chapter 4), the problem of scheduling in the permutation flowshop with the objective of minimising completion-time variance is considered and solved by making use of the SA algorithm, and a comparison of the ACO-based algorithm and SA-based algorithm have been made.

A simulated annealing algorithm is developed in this part of research along with the development of two modified simulated annealing algorithms (MSA1 and MSA2) to solve the problem of scheduling in permutation flowshops with the objective of minimising the CTV of jobs. The proposed simulated annealing
algorithm is an adaptation of the conventional SA procedure whereas the proposed modified simulated annealing algorithms employ the probability of acceptance which is independent of $\Delta$ (but depends on the iteration number or counter) and differs from the conventional simulated annealing algorithm in which the probability of acceptance depends on both $\Delta$ and temperature.

The first algorithm, called SA, works as much a regular simulated annealing algorithm which follows the temperature reduction in each iteration and using the temperature value, the corresponding probability of acceptance is evaluated using $\exp(-\Delta/T)$. In the second algorithm, called MSA1, the probability of acceptance is initially fixed as 0.975 that is independent of $\Delta$ and temperature used conventionally in the SA, and this acceptance probability is subsequently reduced over the iterations of the MSA1. The third algorithm, called MSA2, is a hybridization of the SA and the MSA1 in which the effectiveness of both algorithms are utilised.

Also an attempt is made to explore the performance of the proposed algorithms with due consideration of allowing the right shifting of completion time of jobs on the last machine as has been done by some of the researchers in the past, for e.g., Mehta et al. (2012). It is also noted from literature (Mehta et al., 2012) that V-shaped sequences can improve the quality of solutions.

Taking note of these observations the proposed approach to solve the flowshop scheduling problems presented in Chapter 4 consists of four phases. In the first phase, the proposed algorithms are used to minimise the completion time variance (CTV) of jobs without any inserted idle time on the last machine. In the second phase, the CTV of jobs are evaluated with right shifting of completion times of jobs (i.e., with inserted idle time) on the last machine. In the third phase of the work, the sequences are converted so as to follow the V-shaped property with respect to processing time of jobs on the last machine, followed by right shifting of completion times of jobs on the last machine so as to take advantage of the special property associated with such sequences. The fourth phase, the variance of tardiness (VT) of jobs without and with right shifting of completion time of jobs on the last machine is computed and reported.

The performance evaluation of the proposed algorithms (SA, MSA1 and MSA2) in comparison to the existing procedures (MACO-I & MACO-II (proposed in
the earlier part of this research)), ACO-PSOA (Rameshkumar, 2011), IG_RS (Ruiz and Stützle, 2007) was carried out on the ninety benchmark problem instances of Taillard (1993). In the first phase of evaluation it is observed that the newly developed algorithms SA, MSA1 and MSA2 yield, respectively, the best CTV for 25, 31 and 31 problem instances out of 90 problems under consideration, and they are competitive with respect to each other and in all they have produced a total of 53 new improved solutions. In the second phase, these three algorithms produced a total of 44 new improved solutions with respect to the existing ADJ-Reduced algorithm (Mehta et al., 2012) out of the 90 problems under consideration. This finding indicates that the employment of right shifting of jobs on the last machine will indeed enhance the quality of solution and also shows the superiority of the proposed algorithms over the existing one under consideration. In the third phase, the proposed three algorithms perform very well in that they have produced a total of 63 new improved solutions with respect to the existing ADJ-Reduced algorithm out of ninety benchmark problems of Taillard (1993). This result leads to the conclusion that, as stated by Eilon and Chowdhury (1977), the V-shaped sequences produce better solutions as is evident from the better performance shown by the proposed algorithms with due consideration for this property of a sequence.

In the fourth phase, the minimisation of the variance of tardiness (VT) of jobs without and with allowing the inserted idle time (i.e., right shifting of job completion times) on the last machine is carried out to solve flowshop scheduling problem. In addition, the combination of V-shaped sequence and right shifting of completion times of jobs on the last machine is also attempted. It is observed observe that no single approach emerges to the best in all possible instances. The performance or outcome appears to depend on the common due date in relative to job completion times. Nevertheless, it appears worthwhile to implement all these three approaches (without and with right shifting) coupled with V-shaped sequences and choose the best outcome.

To summarise the findings of this part of research it can be said that the extensive experimentation shows that the SA performs better than ACO in solving the problem of scheduling in permutation flowshops with the objective of minimising the completion time variance of jobs.
6.1.3 Heuristic Algorithm for Minimisation of Total Flowtime

The third and concluding part of the research work focuses on the development of a heuristic algorithm and new search schemes to solve the flowshop scheduling problem with the objective of minimising the total flowtime (TFT) of jobs, which is one of the most widely researched performance measure because of its practical significance. The performance analysis of the ant-colony optimisation (ACO) algorithm and simulated annealing (SA) algorithm developed in the earlier part of this work (Chapter 4) shows that the SA performs better than ACO in solving the problem of scheduling in permutation flowshops with the objective of minimising the completion time variance of jobs. Based on this observation, an attempt is made in this part of research (Chapter 5) to employ a simulated annealing based Heuristic Algorithm (HA) to solve the permutation flowshop scheduling problems with the objective of minimising the TFT of jobs.

With the proposal of the HA, this research work adds to the existing literature on flowtime minimisation in flowshop scheduling. The extensive performance analysis carried out in this part of work shows that the HA implementations altogether provide better solutions for 50 problem instances out of 90 problem instances considered in comparison to the TFT of jobs obtained by the particle swarm optimisation (PSO) algorithm proposed by Tasgetiren et al. (2007) as reported in Dong et al. (2009). Also, the HA implementations emerge competitive to the ILS by Dong et al. (2009), and are able to provide the improved TFT of jobs for 28 problem instances out of 90 problem instances in comparison to the ILS proposed by Dong et al. (2009).

Further in an attempt to compare the performance of the proposed HA with that of SA algorithm additional experimentations were carried out with MSA2 (developed in Chapter 4) as applied to TFT criterion. The results demonstrate the better performance of the HA implementations. Nevertheless it appears that the developed HA implementations do not perform quite well for the larger-sized problems (i.e., 100 × 20). This observation calls for further investigation.

6.2 CONTRIBUTIONS OF THE PRESENT WORK

The present work focuses on development of heuristic algorithms to solve permutation flowshop scheduling problems (PFSP) with the consideration of
performance measures such as completion time variance (CTV) of jobs and variance of tardiness (VT) of jobs that are not so widely considered by researchers. Algorithms based on the concepts of established metaheuristics, namely ant colony optimisation (ACO) and simulated annealing (SA), have been developed.

Though past research has resulted in the development of metaheuristics such as ACO or SA in solving scheduling problems, variations in these generic structures are being attempted and in this direction, the present work has extended the research on flowshop scheduling by developing variations in ACO-based and SA-based algorithms.

Researchers in the past have developed many local search techniques which helped to create neighbourhood sequences in the process of improving the solution. Though these processes have been found to be effective in yielding better results in this direction no attempt has been made to study the influence of using these local search techniques in a combined way. This research work makes an attempt in this direction by incorporating the combination or concatenation of proven local search processes such as JIBIS and JIBSS, which is perhaps the first of its kind work in scheduling research. The study shows the effective exploitation of such concatenations in improving the solution.

Further two new local search techniques have also been developed in the process of developing algorithms.

The research work also focused on the study on the effect of right shifting of jobs on the last machines as well as the consideration of V-shaped sequences on the chosen performance measures which only a very few researchers such as Mehta et al. (2012) have attempted. Through extensive computational experimentation it has been shown that these properties can be exploited in improving the solution to the scheduling problems.

The study has been extended towards minimisation of variance in tardiness (VT) which is, again, perhaps the first kind of its work and hence the values for VT reported in the present work can set benchmark values for VT for the problem instances of Taillard (1993).

To conclude, the development of ACO-based and SA-based algorithms along with the incorporation of the concept of the concatenations of proven local search
schemes, which have been shown to yield effective improvements in the solution of PFSP problems, is deemed to contribute to the existing scheduling literature. Further the variance in tardiness values obtained in this work can be set as benchmark values for the problems of Taillard (1993) which can also add to the related literature. Therefore the present study is a humble contribution to the existing body of knowledge in the field of flowshop scheduling.

6.3 SCOPE FOR FURTHER RESEARCH

From this study, one see that combining good features, fine tuning of parameters involved and modifications with respect to the existing algorithms lead to improve the new algorithm which in turn produces high quality solutions. The following future extensions are possible to the research work discussed in this thesis.

i) Future research may address the problem of permutation flowshop scheduling with multi-objectives and the possibility of applying the algorithms developed in the present work may be explored.

ii) Future research may be targeted by employing the proposed algorithms on other shop environments with various non regular performance measures like completion time variance and variance of tardiness under consideration.

iii) The proposed algorithms in this research study may be hybridized with other local search methods to explore the possibility of getting better solutions for the problem under consideration.
APPENDIX – A1

Description of the NEH procedure

The NEH heuristic, as adapted in the current study with the CTV objective, works as follows.

Phase 1:

Arrange the job in descending order of \( \sum_{j=1}^{m} t_{ij} \), where \( t_{ij} \) denotes the process time of job \( i \) on machine \( j \).

Phase 2:

Improve the sequence obtained from Phase 1 using the insertion scheme explained below.

Suppose we have \{5-4-3-2-1\} as the seed sequence obtained from Phase 1. Call this sequence the overall seed sequence.

Do the following insertion procedure once:

Step 1: Choose the job found in the first and second positions, i.e., jobs 5 and 4. Form the two possible sequences, i.e., \{5-4\} and \{4-5\}. Choose the better partial sequence with respect to the objective under consideration, i.e. CTV. Assume that \{4-5\} is chosen as the current partial sequence.

Step 2: Consider the job found in the third position of the overall seed sequence, i.e., job 3. Insert the job in all positions of the current partial sequence, and hence obtain \{3-4-5\}, \{4-3-5\} and \{4-5-3\}. Choose the best partial sequence, say \{4-3-5\}, and update the current partial sequence as \{4-3-5\}.

Step 3: Likewise, proceed until the job found in the last position of the overall seed sequence is inserted in all possible positions of the current partial sequence, and the best complete sequence is obtained.

Step 4: Compare the sequence obtained from Step 3 and the overall seed sequence (i.e., \{5-4-3-2-1\} in this case) with respect to the CTV. Return the better solution.