JUNE,2012

M.Sc. in Industrial Engineering

FİLİZ ŞENYÜZLÜLER

UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF
NATURAL & APPLIED SCIENCES

# A SOFT COMPUTING BASED APPROACH TO INTEGRATED PROCESS PLANNING AND SCHEDULING WITH SETUP AND MACHINE CAPACITY CONSIDERATIONS

M. Sc. THESIS
IN
INDUSTRIAL ENGINEERING

BY
**FİLİZ ŞENYÜZLÜLER**
**JUNE 2012**

# A Soft Computing Based Approach to Integrated Process Planning and Scheduling with Setup and Machine Capacity Considerations

**A Master's Thesis**

**in**

**Industrial Engineering**

**University of Gaziantep**

**Supervisor**

**Prof.Dr. Türkay Dereli**

**Co-Supervisor**

**Prof.Dr. Adil Baykasoğlu**

**By**

**Filiz ŞENYÜZLÜLER**

**June 2012**

REPUBLIC OF TURKEY
UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES
NAME OF THE DEPARTMENT

Name of the thesis: A Soft Computing Based Approach To Integrated Process Planning And Scheduling With Setup And Machine Capacity Considerations
Name of the student: Filiz Şenyüzlüler
Exam date: 11.06.2012

Approval of the Graduate School of Natural and Applied Sciences

Prof.Dr. RAMAZAN KOÇ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science/Doctor of Philosophy.

Prof.Dr. Türkay Dereli
Head of Department

This is to certify that we have read this thesis and that in our consensus/majority opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science/Doctor of Philosophy.

Adil BAYKASOĞLU
Co-Supervisor

TÜRKAY DERELİ
Supervisor

Examining Committee Members

Signature

Prof.Dr. Adil Baykasoğlu ........................
Prof.Dr. Türkay Dereli ........................
Prof.Dr. Ömer Eyercioğlu ........................
Prof.Dr. Rızvan Erol ........................
Asst. Prof. Dr. Ömer Akgöbek ........................

# ABSTRACT

# A SOFT COMPUTING BASED APPROACH TO INTEGRATED PROCESS PLANNING AND SCHEDULING WITH SETUP AND MACHINE CAPACITY CONSIDERATIONS

ŞENYÜZLÜLER, Filiz
M.Sc. In Industrial Engineering
Supervisor: Prof.Dr. Türkay Dereli
Co-Supervisor: Prof.Dr. Adil Baykasoğlu
June 2012, 125 pages

Rapidly developing technology turned the manufacturing environments into a more and more flexible, computerized world of "complex products and requirements". In such an environment, using the traditional practice of sequential process planning and scheduling remains insufficient and generates problems. In order to overcome this situation, integrated process planning & scheduling has posed a great challenge to researchers.

The study presents a literature survey on integrated process planning and scheduling (IPPS) and investigates a new approach which makes use of Baykasoglu's grammatical representation of generic process plans and Clonal Selection Algorithm (CLONALG) in order to integrate process planning and job shop scheduling. In the proposed integration approach, Giffler & Thompson Algorithm (used to generate active schedules) and VIKOR Method (used to handle multiple objectives) are employed considering setup times and machine capacity constraints.

A computer program in C# programming language is developed for finding optimal or near-optimal schedules for IPPS problems. Proposed approach is also applied to case studies in order to analyze its performance.

**Key Words:** Integrated Process Planning and Scheduling, VIKOR, Giffler & Thompson Algorithm, CLONALG, Optimization

# ÖZ

## KURULUM SÜRESİ VE MAKİNE KAPASİTE KISITLI BÜTÜNLEŞİK PROSES PLANLAMA VE ÇİZELGELEME PROBLEMLERİNE ESNEK HESAPLAMA YÖNTEMİ İLE BİR YAKLAŞIM

ŞENYÜZLÜLER, Filiz
Yüksek Lisans Tezi, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof.Dr. Türkay Dereli
Yardımcı Tez Yöneticisi: Prof.Dr. Adil Baykasoğlu
Haziran 2012, 125 sayfa

Hızla gelişen teknoloji, üretici çevrelerini, kompleks yapıda ürünler ve karmaşık gereksinimlerle dolu, daha esnek ve bilgisayar tabanlı bir yapıda olma durumuna sürüklemiştir. Böyle bir ortamda, geleneksel ardışık süreç planlama ve çizelgeleme uygulamaları yetersiz kalmakta ve problemlere sebep olmaktadır. Bu durumun üstesinden gelebilmek için geliştirilen bütünleşik süreç planlama ve çizelgeleme metodu araştırmacılara yeni bir araştırma alanı oluşturmuştur.

Bu çalışmada, bütünleşik süreç planlama ve çizelgeleme problemleri üzerine bir literature taraması yapılmış ve Baykasoğlu'nun genel süreç planları için geliştirdiği gramatiksel gösterim yöntemi ile klonal seçim algoritması CLONALG kullanılarak bütünleşik süreç planlama ve çizelgeleme problemlerinin çözümüne yönelik yeni bir yaklaşım geliştirilmiştir. Bu yaklaşımda kurulum süresi ve makine kapasite kısıtları göz önünde bulundurularak, aktif çizelgeler üretebilmek için Giffler & Thompson algoritması, birden çok amaç fonksiyonu ile çalışabilmek için VIKOR metodu seçilmiştir.

Bütünleşik süreç planlama ve çizelgeleme problemlerinde optimum ya da optimuma yakın sonuçlar veren çizelgeler üretebilmek amacıyla C# programlama dilinde bir bilgisayar programı geliştirilmiş ve örnek olaylar üzerinde uygulamalar yapılmıştır.

**Anahtar Kelimeler:** Bütünleşik süreç planlama ve çizelgeleme, CLONALG, VIKOR Method, Giffler & Thompson, optimization.

# ACKNOWLEDGEMENT

First and foremost, I would like to express my gratitude to ALLAH, the almighty and the merciful, for giving me peace and strength to retain my hopes.

I am grateful to my supervisor Prof. Dr. Türkay Dereli and my co-supervisor Prof. Dr. Adil Baykasoğlu for giving me the opportunity to work on this project. This research project would not have been possible without their encouragement, guidance and support. I also thank to TUBITAK - BIDEB(2228) for supporting this project.

I would like to thank all people in the Department of Industrial Engineering, especially to Asst. Prof. Dr. Ömer Akgöbek for his understanding, tolerance and help during my study. Special thanks go to my dear friend Merve Emrullah for heartening and supporting me all along.

I owe many thanks to my husband Güven, for the emotional support he has given me. He has always encouraged and believed in me. His help in this study are invaluable and very precious.

Words can't express my thanks to my beloved mother Nazmiye, who trusts and helps me during this project as she does all the time in my life. She has always been a glimmer of hope for me. I also would like to thank my father Şafak and my sister Özgür for all their support.

***To my family…***

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

## LIST OF SYMBOLS/ABBREVIATIONS

$O_j$ : Operation $j$

$P_i$ : Part $i$

$M_k$ : Machine $k$

GPPc : Generic process plan for component $c$

P_Ti : Preparatory tasks for the component

F_Tj : Formative tasks for the component

TR_Tk : Transitional tasks for the component

T_Tl : Terminating tasks for the component

$P$ : Set of non-value adding tasks

$F$ : Set of value adding tasks

$\Sigma_S$ : A set of terminal symbols, representing the alternative machines for machining operations

$\Pi_S$ : A set of nonterminal symbols, representing parts, their feasible operation sequences and processing requirements

$\Phi_S$ : A finite set of production rules for determining the legal process plans and machines to be selected

S : A non-terminal start symbol. Select a process plan for each part, and a machine for each operation

$\Omega_i$ , $\Theta_i$ : Non-terminals

$\Delta$, $\Psi_{ij}$ : Sets of non-terminals

$\Lambda_{ij}$ : Terminals

$\pi_{ij}$ : The operation sequence selection controls

$\mu^k{}_{ij}$ : The machine selection controls

$PS_t$ : Partial schedule

$S_t$ : Set of schedulable operations at iteration $t$

$\sigma_i$ : Earliest time at operation $i$ can start

$\Phi_i$ : Earliest time at operation $i$ can be completed

$C_t$ : Set of conflicting operations in iteration *t*

$p_t$ : Dispatching rule chosen in iteration *t*

$Ab_i$ : Antibody *i*

$f$ : The affinity value for each antibody

C : Clone set

$L_{Pj}$ : The ranking measure for alternative j

F* : The ideal solution

$F^c$ : Compromised solution

$f_{ij}$ : Evaluation result of alternative $a_j$ according to criteria i

$f_i^*$ : $max_j\ f_{ij}$

$f_i^-$: $min_j\ f_{ij}$

$S_j$ : The average group score for $j^{th}$ alternative

$R_j$ : The worst group score

Qj : Relative closeness of a particular alternative to the ideal solution

$S^*$ : $Min_j\ S_j$

$S^-$ : $Max_j\ S_j$

$R^*$ : $Min_j\ R_j$

$R^-$: $Max_j\ R_j$

$w_i$ : Weight of criteria i

v : Weight of the strategy that ensures maximum group benefit

(1-v) : Weight of the strategy that ensures minimum individual regret

$a_k$ : Alternative *k* which is the best ranked by measure Q

$a_l$ : The second best alternative *l* ranked by measure Q

$PP_i \ldots PP_f$ : operation sequence alternatives set *(i...f)*

$O_x \ldots O_y$ : operation locations *(x...y)*

$p_1 \ldots p_t$ : dispatching rule sets *(l...t)*

n : number of candidate parts

m : number of operations

k : number of dispatching rules

Ci : completion time of part *i*

$VC_{jk}$= The variable cost function for operation *j* on machine *k*

$t_{jk}$ = The processing time for performing operation *j* on machine *k*

A : Shape factor which determines the curvature of currency function

cr : The critical ratio

$d_i$ : Due date of part *i*

T : The processing time of the part

# CHAPTER 1

# PROBLEM STATEMENT

## 1.1 Introduction

The benefits gained on manufacturing industry by integrating process planning and scheduling pushed the researchers to pay great effort on this subject for many years. Before these studies and attempts, scheduling is done separately from process planning, after the process planning is completed. This traditional approach seems to be simple to implement but it leads to some performance problems in real cases as manufacturing resources have considerable amount of overlapping capability in today's dynamic and highly capable manufacturing environments. Increasing flexibility makes it harder for companies to be able to survive and compete with others in such responsive and agile manufacturing environments. Thus the integration becomes indispensible for sustaining robustness in scheduling and for responding to dynamic changes during the process planning and scheduling.

Hence, sequential approaches cause several problems to arise such as (Gindy et al. 1999) (Zhang and Alting 1989):

- Scheduling starts right after process planning ends, so it is not possible to change fixed process plans during the scheduling phase which happens to result into generation of unrealistic process plans. Thus they cannot be truly followed in the shop floor.
- In process planning, it is assumed that the factory is %100 idle and has no capacity constraints in terms of resources.
- When the technical issues are considered, always certain machines are selected during process planning. As a result, the other machines kept idle.

- During the process planning, the machine tools are not always changed perfectly due to the foreman's inexperience and lack of knowledge. Therefore this result into inefficiencies during the process planning and scheduling phases.
- The selection of local optimum machines not always produces optimal solutions.

### 1.2 Process Planning and Computer Aided Process Planning (CAPP)

According to the definition of (Chang and Wysk, 1985) process planning is transforming an engineering design to a final part by preparing detailed operation instructions. It determines the required resources, chosen machines, route of the processes in order to manufacture the final part. There can be many alternative plans for a product as the flexibility of the system increases. Thus, generating and evaluating all the alternatives may take serious amount of time in a complex system. To overcome this problem, Computer Aided Process Planning (CAPP) is used. CAPP, which benefits the computer technology, is the main interface in CAD/CAM integration. As processes and required parameters are determined, it also converts a block into a final product. (Kayacan et al, 1996). (Wang et al, 2009c) modeled process planning and scheduling separately as CAPP module and scheduling module, each can be run in a stand-alone mode. The integration is achieved by an integrator module, which uses a set of heuristics considering tardy job reduction and cost minimization.

CAPP has two major types called variant and generative. In variant approach an existing plan of a similar part is recalled and retrieved, then that plan is adopted for the new part by making the necessary modifications on (Morad and Zalzala, 1999). This approach is usable when there exists a soft product variety. However, the effect of process planners' knowledge background on the quality of a process plan cannot be ignored (Zhang and Xie, 2007). In generative approach, the process plan is generated from scratch by advanced algorithms with the given required product data. (Kumar and Rajotia, 2005) studied on the process planning of axisymmetric components using the generative approach. Cost minimization was considered as performance criteria. Also it is possible to talk about a hybrid process planning approach developed by (Hashemipour, 2004). The study combined variant and generative process planning in order to develop a CAPP tool with the software "Process Planning and Scheduling For Garment Manufacturing Unit" (PPSGAR). The study is modeled with an object-oriented architecture to be used in apparel industry.

### 1.3 Scheduling

In 1967, Conway made the definition of scheduling as; " sequencing the operations of all jobs (products) based on precedence constraints, considering time aspects". (Conway et al, 1967) classified the scheduling problem into different categories considering four parameters, which are machine numbers in the shop, job arrival patterns, flow patterns in the shop and the schedule evaluation criteria. Uncertainty in the existence of resources, finite capacity of resources, complex constraints in the system and multiple resource requirements of various operations are the key factors for that the production scheduling is a hard and complex problem (Sadeh et al, 1998).

### 1.4 Integrated Process Planning and Scheduling (IPPS)

From the traditional point of view, production design, process planning and manufacturing control functions are considered as separate phases in production environment. However, decisions taken in one stage affect the decisions in the up-coming stages (Gu et al, 1997). The traditional vision has justifiable reasons for its time period, but with the developments in technology today these reasons attenuated and a trail for integration is blazed. Idea of integrating these two important functions of manufacturing system has made a tremendous impression in the literature and till now there has been a considerable amount of studies have been done. (Tan and Khoshnevis, 2000) defined the IPPS problem as: "In a system with *n parts* and *m machines*, determine a feasible process plan, manufacturing resources and sequence of operations to determine a schedule considering the alternative process plans, resources and constraints with the aim of achieving stated objectives."

Some of the integration objectives are as follows:

- To use manufacturing resources in a more effective way.
- To be adaptive to dynamic changes (break-downs, order changes…etc.) in the shop floor.

- To avoid the problems caused by conflicting objectives of process planning and scheduling. (Kumar and Rajotia, 2003)
- To reduce scheduling conflicts, flow-time and work-in-process, (Li et al, 2010b)
- To avoid unrealistic process plans which cannot be executed on the shop floor.

This thesis presents a literature survey on IPPS. Ninety-six papers are reviewed and a differently from the other reviews in literature, a statistical study of them is proposed. The papers are classified according to the integration approaches, methodologies and algorithms used. The results of the statistical study are showed using graphs and charts.

A new approach is proposed which uses Baykasoglu’s grammatical representation of generic process plans to solve IPPS problem with setup and machine capacity considerations. CLONALG optimization algorithm, Giffler & Thompson algorithm and VIKOR Method are used during the implementation of the approach. In the literature, the problem proposed in this study has not been studied with the capacity constraints and setup times. These constraints on this topic are firstly considered in this thesis.

A software program in C# programming language is developed. MYSQL Database is used to store and retrieve data. Three cases with small, medium and large data sets are developed. The benchmark problems are solved in order to analyze the performance of the proposed method.

## 1.5 Conclusion

Due to the large number of interacting decisions, the ill defined and often conflicting nature of objectives and the inherent unpredictability of production systems, IPPS becomes a complex problem (Sadeh et al, 1998). (Shobrys and White, 2000) handled this situation from a different point of view and commented that companies struggled in the pursuit of integration because of two main challenges. The first one is the change in human behavior in the way that they are getting adopted to use more complicated and complex tools. The other is that, the organizations give more emphasis on integrating decision making by getting different parts of the organization together.

In this study, a literature survey on IPPS is presented in Chapter 2. The methodologies (grammatical representation of generic process plans, CLONALG, Giffler & Thompson Algorithm, VIKOR Method), which are benefited during the implementation of the proposed method, are presented in Chapter 3.

A computer program in C# programming language is developed for finding optimal or near-optimal schedules for IPPS problems. Proposed approach is also applied on some case studies in order to analyze its performance.

# CHAPTER 2

# LITERATURE SURVEY

## 2.1 Introduction

Rapidly developing technology turned the manufacturing environments in to a more and more flexible, computerized world of "complex products and requirements". In such an environment, using the traditional practice of sequential process planning and scheduling remains insufficient and generates problems. In order to overcome this situation, integrated process planning scheduling has posed a great challenge to researchers. As the need for integration of process planning and scheduling functions grows, the number of studies done on this issue has grown too.

This chapter presents a literature survey on integrated process planning and scheduling (IPPS). The reviewed papers are discussed in detail and a statistical study on them is presented. The methods used, architectures, integration approaches - Non-linear Process Planning (NLPP), Closed-Loop Process Planning (CLPP) and Distributed Process Planning (DPP); implementation models - Multi-agent architecture, The algorithm-based approach, Holonic Architecture, Object - Oriented Based Approach and other; and objectives are taken into consideration. Many databases including, "*ScienceDirect, Informaworld, IEE, JStor and Springerlink* " are used to conduct the research.

## 2.2 Background of The Previous Reviews

Tan and Khoshnevis review the studies up to year 2000. In 2010 (Li et al, 2010) presented their review, which contains the new studies done on the topic.

(Tan and Khoshnevis, 2000) reviewed the background of integration problem, discussed the advantages and difficulties of integration. Authors presented a critique to the current approaches and presented some potential future research directions. Categorization of the reviewed papers are done via matrices where the columns are major categories of approaches (Concurrent assignment, AI method, Decision matrix, Nonlinear process planning, Math Programming, Search based heuristic), and the value in each cell indicates the extent of concentration of the papers on the category, with 5 being the maximum concentration and 0 the minimum.

(Li et al, 2010) reviewed the process planning, scheduling and IPPS. The authors examine the IPPS papers according to their integration model: NLPP, CLPP and DPP. The advantages and disadvantages of each model are discussed. The review is classified under three main implementation approaches: Agent-based approach, the petri-net and the algorithm-based approach. The critique of the approaches is done and future research trends are discussed.

(Phanden et al, 2011) reviewed the IPPS according to integration approaches. 77 papers are examined and the features of non-linear approach (NA), closed-loop approach (CA) and distributed approach (DA) are presented. The authors also presented their comments on IPPS according to the findings in their study.

Differently from the previous review studies like (Tan and Khoshnevis, 2000), (Phanden et al, 2011) and (Li et al, 2010), a statistical study on the reviewed papers is also performed. The distribution of the studies using different integration approaches according to the years and the portion of studies considering the implementation models /architectures used are shown in graphics. The algorithm-based approaches are also subdivided into sections like “ GA, SA, ACO, AIS, Hybrid and Neural Network” and the percentage of each section is also shown. Each studied paper is elaborately presented and detailed information about each of them is presented to the reader.

## 2.3 The Reviewed Studies

### 2.3.1 Integration Approaches

The traditional way of process planning and scheduling, which is done sequentially, generates immense problems as it is mentioned before in section 1.1. To overcome these problems, researchers have focused on developing new approaches, which can achieve optimal manufacturing performance. According to the integration approaches IPPS problem can be classified into three groups: NLPP, CLPP and DPP.

#### 2.3.1.1 NLPP

NLPP got its name because of its inherent property of being non-linear, for instance, the generated process plans don't form a linear structure, instead; they are branch at every node (Zhang and Alting, 1989). This approach is also called alternative process planning or flexible process planning. In NLPP, all the possible process plans for a given part are generated and then based on their feasibility values the plans are ranked (Tönshoff et al., 1989), (Carton and Ray, 1991). The one with the highest feasibility is chosen if it is suitable for the current status otherwise the next plan with highest feasibility is checked for suitability. Production scheduling still strictly follows process planning. The integration is mainly achieved by providing alternative process and/or operation routes to the scheduling function (Zhang et al, 1998). This approach is usable when the part number is not very large, because the number of plans expands exponentially as the part number increases. This results in that high amount of time and storage is consumed.

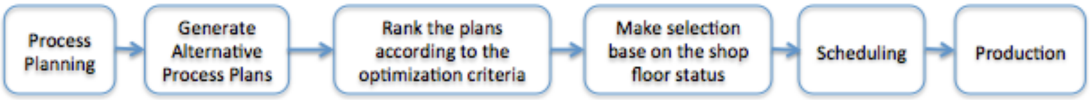


**Figure 1** NLPP approach (Li et al, 2010)

The first approach in IPPS is proposed by (Chryssolouris and Chan, 1985) which they called MADEMA (Manufacturing Decision Making Approach). MADEMA has five steps which are; determining alternatives, determining attributes, determining consequences with respect to attributes for each alternative, applying decision rules for choosing the best alternative and lastly selecting the best alternative.

(Jablonski et al, 1990) developed a system called FIPS (Flexibly Integrated Production Planning and Scheduling) that consists of three main parts: a feature recognizing system (FREDOS), a dynamic resource allocation system (DRS) AND static process planning system (SSM). First of all, the geometric features of the parts are recognized by FREDOS, and the needed manufacturing operations for execution are determined. Then all the possible resource combinations to be able to execute the determined manufacturing operations are extracted by SSM. Lastly for each manufacturing order, DRS selects on-line the suitable resource combination found by SSM considering the features of the parts to be manufactured. A specification language SORC (Strategic Ordering of Resource Combinations) is used to hold and order the combinations supplied by FIPS, according to requirements like a database. (Palmer, 1996) also used NLPP approach in his study, which is told in section 2.3.2.2.3 detail.

(Cho et al, 1997) studied on NP-complete problem of determining the rule combinations for robust process planning and scheduling. An intelligent workstation controller (IWC) is developed for automated flexible manufacturing systems. Planning and scheduling strategies are considered as controllable factors. Machine breakdown rates, number of machines, part mix, processing time distribution and number of buffers are considered as uncontrollable factors. The aim of the study was to determine the set of controllable factors, which consistently provide the IWC with effectiveness and robustness over various uncontrollable factors.

(Kim and Egbelu, 1998), also studied IPPS in flexible manufacturing environments. The detailed information about the study is given in section 2.3.2.5.4.

(Kim and Egbelu, 1999) proposed a framework for integrated process selection and scheduling in a job shop environment. There were multiple jobs and multiple process plans of each job. The framework of the study comprises two subsystems. The first one is process plan selection subsystem and the second one is shop scheduling subsystem. When a new part is arrived to the system process plan selection subsystem generates the alternative process plans for that part and saves them in process plan database. From shop capacity database, process capability information and the number of machines in the shop are obtained. In order to select the right process plans to optimize the production makespan, finally the following information is obtained from the shop status database: workload on each machine, in-process jobs currently assigned to each machine and the functional status of each machine. The problem is modeled and solved using the mathematical programming language MPSX (Mathematical Programming System eXtended) and two algorithms are developed which are preprocessing algorithm and iterative algorithm in order to decrease computational effort and to be able to make comparison.

(Aldakhilallah and Ramesh, 1999) developed a Computer Integrated Process Planning and Scheduling (CIPPS) system. The proposed system has a dynamic support mechanism for design decisions. The system is composed of four modules. The first one is the super relation graph, which recognize the features automatically. The second one is the cover set model, which determines the minimal cover sets of product features. The third one is the cover set planning and scheduling algorithm. This module determines an efficient and feasible process plan. The last one is the cover set cyclic scheduling algorithm, which generates an efficient and feasible cyclic production schedule.

(Saygin and Kilic, 1999) integrated flexible process plans with off-line scheduling in FMS using a framework, which has four integrated stages. *Machine tool selection*: linear alternative process plans was input to this module, which included alternative machine tools. For each operation one machine tool was selected. *Process plan selection:* The best alternative process plan for each part was selected using Dissimilarity Maximization Method (DMM). The performance criterion was the number of dissimilarities among the candidate process plans to be maximized. *Scheduling:* An integer linear program (ILP) formulation was adopted in this stage. *Re-scheduling modules*: the operation was re-allocated considering the idle time on the alternative machine tools. Also during this stage, the flow time of the part, which had the re-allocated operation, was reduced, if it is possible. The overall objective of the study was to reduce makespan.

(Lee and Kim, 2001) studied the IPPS using simulation based genetic algorithms. For each part the alternative process plans are generated by a CAPP system and one alternative process plan is randomly selected for simulation module. The detailed information about the study is given in section 2.3.2.2.

(Yang et al, 2001) developed a prototype feature-based multiple alternative process planning system. The system generates process plans based on recognized manufacturing features and production rules. The prototype system is composed of four basic parts which are; relational manufacturing database, form feature recognition system, alternative process plan generator and scheduling state evaluation system. The objective of the study is cost minimization.

(Gan and Lee, 2002) used flexible process planning approach in their study using branch and bound algorithm. The performance criterion was the weighted earliness of jobs in a mould-manufacturing environment. The purpose of preferring mould-manufacturing shops as studying environment is to be able to focus on the demands of less integrated factories. The proposed system acts as an intermediate stage from a present semi-automated factory to a fully automated FMS. (Moon et al, 2003) used NLPP approach in their study, which is told in section 2.3.2.2.1 in detail.

(Kim et al, 2003) developed the symbiotic evolutionary algorithm for IPPS problem in job shop flexible manufacturing systems. Localized interactions, steady-state reproduction, and random symbiotic partner selection strategies are used to enhance population diversity and search efficiency in the algorithm. In this study, the objective of process planning problem is minimizing the absolute deviation of machine loads while the objective of scheduling problem is minimizing the makespan, minimizing the mean flow time and maximizing the utilization of the machines.

(Jain et al, 2006) developed a framework, which comprises two modules. The first module was process plans selection module (PPSM) and the other was scheduling module (SM). PPSM computes total production time of multiple process plans for each part type. Four of the best process plans are selected according to the minimum total production time as criterion for each part. Then the selected plans are ranked in a descending order according to their implemented criterion. The data of the plans are saved in database. SM retrieves the best process plan from database and selects a part using a dispatching rule. Based on the selected process plan and selected part, SM determines the machine to be used for the selected part. If the machine is not available it switches to the next best plan. Makespan and mean flow time are used as performance measures for this study.

(Leung et al, 2009) presented an ACO algorithm benefiting the advantages of flexible system architectures and responsive fault tolerance in a multi-agent system (MAS) platform. The detailed information about the study is given in section 2.3.2.2.

(Haddadzade et al, 2009) developed an integrated model, which is formulated mathematically. The model is consist of two modules with the objective of optimizing cutting parameters for milling operations as minimizing the cost considering overtime .The process planning module generates all possible process plans with help of CAPP; then, scheduling module ranks these process plans based on minimum cost.

(Wang et al, 2009a) aims to reduce tardy jobs in IPPS problem with a dynamic approach in a batch-manufacturing environment. This approach follows a flexible process planning strategy with the objective of minimizing manufacturing cost. Set of heuristic based algorithms have been used such as EDD, SPT, FH-tardy and QH tardy.

(Raj Kumar et al, 2010) proposed GRASP (multi-objective greedy randomized adaptive search procedures) for IPPS problem considering minimum makespan, maximum workload, total workload, tardiness and total flow time as performance criteria. The proposed procedure is tested with four benchmarking problems and the results showed that GRASP is a promising for solving IPPS problem.

(Li et al, 2009a) and (Zattar et al, 2010) also studied IPPS in flexible manufacturing environments. The detailed information about the study is given in section 2.3.2.1. (Li and McMahon, 2007)' s study will be told in section 2.3.2.2.3 detail. (Weintraub et al, 1998) 's study will be told in section 2.3.2.2.7 in detail. (Shao et al, 2009) and (Lihong and Shengping, 2012) also proposed a study with NLPP approach, which is told in section 2.3.2.2.1 in detail.

#### 2.3.1.2 CLPP

In CLPP, based on the current shop-floor status, information about availability of the resources and machines are sent to the process planning function. CLPP (*or dynamic process planning*) is so called because it forms closed loops in which the dynamic feedback from the scheduling function is taken. Because of it structure based on real time data, in each loop only suitable process plans are generated. However obtaining and updating the real-time data can be hard, when it has to regenerate process plans in every scheduling phase (Li et al, 2010a).

(Dong et al, 1992) used dynamic process planning approach for IPPS problem in their study, where the performance criterion was the smallest slack time for scheduling of a batch size manufacturing shop. The rough process plans were prepared and the priority for each operation is determined via geometric constraints, then these process plans were sent to scheduling module. (Cho et al., 1998) developed a prototype Block Assembly Process Planning and Scheduling system in shipbuilding. In process planning phase, the optimal assembly units and sequences are determined by a rule based technology. In scheduling phase, the blocks are reallocated to alternative assembly shops using a schedule revision heuristic. The workload is balanced in the study. (Baker and Maropoulos, 2000) used CLPP approach in their study, which is told in section 2.3.2.4 in detail. (Sugimura et al, 2001) developed an IPPS System in a holonic environment to realize a flexible production control, which is told in section 2.3.2.3 in detail. (Wang et al., 2002) also used CLPP approach in their study to solve IPPS problem. They developed a mechanism, which facilitates dynamically in a batch-manufacturing environment.

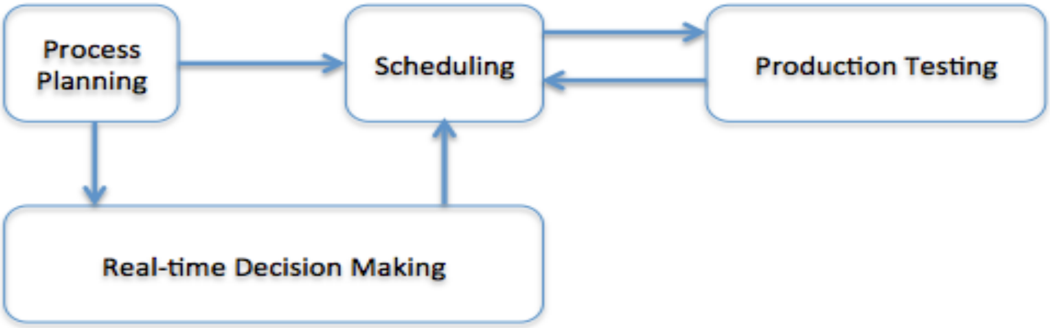


**Figure 2** CLPP approach (Li et al, 2010)

(Kumar and Rajotia, 2003) proposed a method for integrating CAPP and scheduling. The system considered the shop floor's real time status and got feedback dynamically. The performance criteria were the number of tardy jobs and the mean flow time to be reduced. The method which can be called as online process planning starts with checking the machines in case of a breakdown, then the availability of each machine is checked, this time the available machines are checked for capability so that the suitable machines for process plans are determined. The ideal process plan is transformed into an actual process plan based on the scheduling factor. The machines are assigned to the operations by the actual scheduling factors, which are computed after considering the availability, capability and machines breakdown condition. According to the results, operations are assigned to the machines with highest value of actual scheduling factor.

(Zhang et al, 2003a) used CLPP approach in their study, which is told in detail in section 2.3.2.2.1. (Wang et al, 2008) also studied on IPPS using the same approach and broadened the study of (Zhang et al, 2003a) using the algorithms they developed: Fine-Tuning (FH-Tardy) and Quick Tuning (QH- Tardy). The results showed that the method reduced the job tardiness by exploring the process plan solution space.

(Lim and Zhang, 2004) developed a system, which integrates dynamic process planning and dynamic production scheduling using a multi-agent architecture. The main goal of the proposed study was increasing the responsiveness of adaptive manufacturing systems against dynamic market changes. The objective of the study is to optimize the utilization of the resources. (Lim and Zhang, 2003), (Wong et al, 2005) and (Wong et al, 2006) also used CLPP approach in their studies which are told in detail in section 2.3.2.1.

(Shresta et al, 2008) studied on IPPS problem for holonic manufacturing systems using dynamic process planning approach, which is told in detail in section 2.3.2.3.

(Guo et al, 2009a) and (Guo et al, 2009b) developed a Particle Swarm Optimization (PSO) algorithm for the IPPS problem. A re-planning method has been developed in case of machine breakdowns and new order arrivals to improve the system flexibility and agility. The detailed information about the studies is given in section 2.3.2.2.

### 2.3.1.3 DPP

In Distributed Process Planning (or *concurrent process planning*), process planning and scheduling are two distributed functions, which start, continue, and end concurrently on the same timeline interacting with each other through the phases they passed. In the first phase the inputs of the system are recognized such as parts, machines, resources and the relationships among them. In the next phase a rough process plan and scheduling plan are obtained. In the last phase detailed plans are determined based on the current shop floor status.

In 1993 (Zhang and Merchant, 1993) proposed a prototype for IPPS functions in three levels. The first level is the integration level, in which pre-planning module determines the possible setups, machining operations and associate times while the available equipment is provided by scheduling module for the next time period. In decision-making level, pairing planning module selects the machines, tools and fixtures. Then the available equipment is matched with the requirements. In the final planning level; operational sequencing, operational tolerance analysis and cost calculations are done. (Huang et al, 1995) used the same levels in their study with a progressive approach. These levels all executed in different time periods but Process planning module and scheduling module interacts with each other in all these three levels. Mathematical models and optimization algorithms are used to solve IPPS problem. The results showed that, using the proposed approach, the computational complexity was reduced.

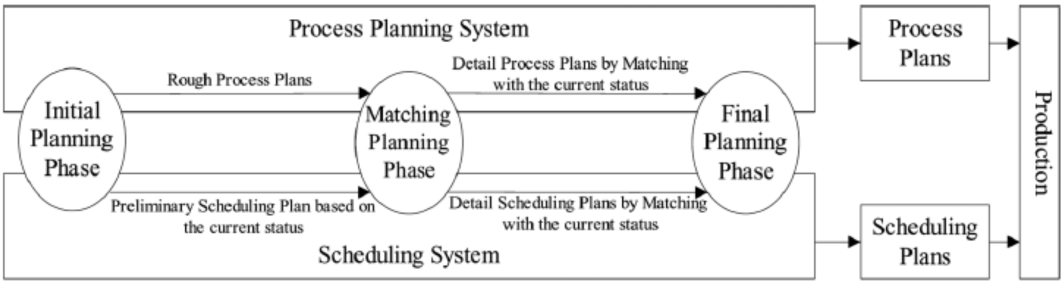


**Figure 3** DPP approach **(Li et al, 2010)**

(Mamalis et al, 1996) developed an online IPPS system with two phases. The first phase was an offline process planning and generation where an information flow is shared between CAPP and scheduling system. In the next phase, the disruption at shop floor is considered by an online process planning and scheduling system. The operation time during the process planning and the total delay of parts during the scheduling were minimized.

(Hildum et al, 1996) and (Sadeh et al, 1998) proposed an IP3S (Integrated Process Planning/Production Scheduling) system for agile manufacturing. The blackboard architecture is adopted. The proposed system supports concurrent development and dynamic revision of IP3S solutions. The detailed information about the study is given in section 2.3.2.5.1.

(Gu et al, 1997) studied on the integration of CAD, process planning and real time scheduling using a bidding-based approach. (Chan et al, 2001) developed an agent-based framework for an integrated, distributed and cooperative process planning system (IDCPPS). (Wu et al, 2002) proposed a multi-agent based computerized model for IPPS problem and aims to resolve critical problems in distributed virtual manufacturing environment. (Shukla et al, 2008) used a bidding based approach for IPPS problem using multi agent structure.  (Wang et al, 2009b) also developed a distributed process planning and scheduling system using an agent-based approach. (Zattar et al, 2007) developed a hierarchical multi-agent model using operation-based time-extended negotiation protocol in a flexible manufacturing environment. (Li et al, 2009c) and (Li et al, 2010c) tried to minimize the production time and makespan for IPPS problem in a job shop environment, using multi-agent based approach. In 'Multi-agent architecture' part, these studies are told in detail.

(Morad and Zalzala, 1999) presented a study consists of two parts, using genetic algorithm for integration. These studies are told in detail in section 2.3.2.2.1. (Gindy et al, 1999) also used concurrent processing planning approach in their study, which contains knowledge base facility modeling functions, feature base process planning system and simulation based scheduling model. The performance criteria are machine utilization and tardiness.

(Zhang et al, 2003b) used a holonic architecture in which the holons are able to cooperate with each other to perform an appointed task flexibly using DPP approach. The detail information is given in section 2.3.2.3.

(Wang et al, 2003) proposed a new methodology for dynamic and distributed process planning using architecture consist of function blocks. Also other supporting technologies such as machining features and agents are used in the model in order to improve responsiveness, flexibility and productivity on machining shop floors.

(Hashemipour, 2004) developed a CAPP tool for the apparel industry. The software PPSGAR is built in an object-oriented architecture. When a new order is received, the required data is obtained from an operation table in the database and the process planner object generates an activity network diagram of the tasks in order to determine the tasks, which can be done simultaneously. Based on the network diagram, a Gantt Chart is drawn and with the processing times data, these are sent as an input to Scheduling object as well as the resource and probability distribution data, in order to get the output data of workstations, efficiency and balanced delay. The possible changes are done and the process planner object is restarted till the best assembly line is selected from the complete result.

In their study, (Sugimura et al, 2006), proposed new manufacturing system architectures with more flexible control structures which can overcome the dynamic changes during manufacturing such as unscheduled disruptions. They used an iterative method, in which the individual job holons modify their process plans in a distributed procedure. The manufacturing processing time and machining cost are considered as performance criteria.

(Ueda et al, 2007) used evolutionary artificial neural networks (EANN) based on an emergent synthesis for simultaneously applying process planning and scheduling. The detailed information about the study is given in section 2.3.2.2.5. (Zhanjie and Ju, 2008) also used DPP approach in their study. They applied GA on their research about IPPS considering load balancing, lowest processing costs and the shortest processing time.

(Cai et al, 2009) touched upon the issue of setup planning and its impact on the integration of process planning and scheduling. Genetic algorithms and a cross-machine setup planning approach is use in the study for machines with different configurations. In the setup-planning phase, the integration of process planning and scheduling is considered. The authors remarked that the most important feature of their study is the distributed process planning (DPP) system, in which the process planning tasks are progressively performed. The objective was to maximize machine utilization and minimize total number of final setups, machining cost and makespan.

(Pawlewski et al, 2009) presented a multi-agent based distributed system and a production planning algorithm is developed that adopts the concept of centralized planning for distributed plans with the aim of improving the control of synchronized production and material flow in supply chain. Issues related to DPP and needs for distributed decision making in multi-entities environment are discussed in their study. Supply chain throughput was optimized.

When the integration approaches of the reviewed papers are considered, it is shown that NLPP and DPP approaches are the most studied approaches with a percentage of 40 and 38. Then CLPP comes next with %22 as it is shown in Figure4.

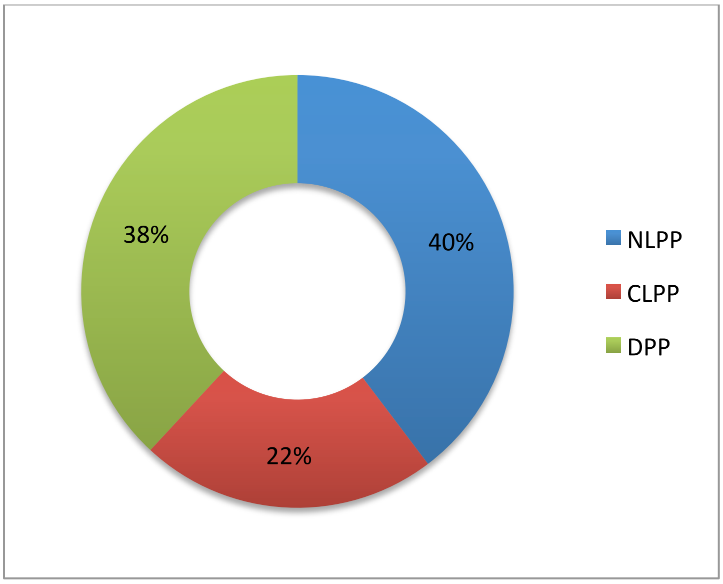


**Figure 4** Percentage of the reviewed papers according to integration approaches

NLLP has an advantage of providing all possible process plans and enhancing the flexibility and availability of process plans (Li et al, 2010a). The NLPP method was preferred this much in the reviewed studies, the reason may be the small size problems are chosen as case studies, because in a large size complex problems providing all possible process plans is hard and time consuming. The reason of CLPP is the least preferred method might be the inefficiencies in assuring the real time data of the current shop floor status (Li et al, 2010a). DPP has a considerable percentage in the statistical results and the future studies are seem to be developed by this method, because process planning and scheduling is concurrently start and end continuously interacting with each other, this method seems to be more efficient in integration logic.

### 2.3.2 Implementation Models / Architectures

The implementation models/ architectures of IPPS can be classified as:

1. Multi-agent architecture
2. The algorithm-based approach
3. Holonic Architecture
4. Object - Oriented Based Approach
5. Other Models

#### 2.3.2.1 Multi-Agent Architecture

An Agent-based model is a computational model where actions and interactions of autonomous individuals are simulated and the effect they have on the system is assessed.

(Nwana and Ndumu, 1997) described the term “agent” as “a component of software and/or hardware which is able to act exactly in order to fulfill tasks in the name of its user” (Nwana and Ndumu, 1997). Agents are intelligent, persistent, social and reactive entities, which have the ability to observe and act upon an environment and direct its activity towards achieving goals. In an agent-based model that is built up for IPPS problem, each agent is responsible for a given task that is coherent with the main goal of integration. The agents get their ability of deciding autonomously from the decision-making heuristics developed for the model. Also each agent-based model needs a communication topology and learning rules in order to maintain integrity and consistency between agents. Number of agents should be determined carefully in a multi-agent based model. Too many agents can cause a high amount of time spent on communicating rather than doing actual work (Zhang and Xie, 2007).

Agent technology is one of the most studied topics in IPPS problem. The researchers may be seen a light at the end of the tunnel that, the number of studies done have grown up especially in recent years.

(Hildum et al, 1997) studied on Blackboard-based agents for supporting mixed-initiative decision-making and integration functionalities. Two approaches are used to obtain results in the experiments done. The first one is the traditional decoupled approach, where process plans were generated without considering the loading issues. The second one is the integrated approach where process plans were optimized by taking into account the presence of bottlenecks. The objective of the study is to make improvements in due date performance. In same year, (Gu et al, 1997) described a bidding-based approach to integrate CAD, process planning and real time scheduling. Detailed information of each part such as design specifications, batch size, and due dates are represented by a STEP Model. The study is done in a multi-agent environment. Defining a coordinating agent and the contract net ensures the necessary hierarchy. Four types of agents are used in the system: part agent, machine agent, shop manager agent and tool agent. The part agent registers it with the shop manager when they first arrive the system. The shop manager (a coordinating agent) keeps track of the system state and inform the machines in the first level of shop hierarchy about the new coming part requirements, the machine agents in this level proves bids for the task in order to respond to the request. The bid selection (scheduling) is based on a cost model which considers machine schedule, machining time, setup time, cost of tooling, tool changing time and the part due date. The agent adds the best bid to its task plan and returns it as the bid of current level, to the agent one level up. As a result the network contraction is provided and the final task plan is determined by the overall best bid.

Up to 2001 not much progress had been done, but after 2001 the attention of the researchers on Multi-agent systems and as a result the number of studies have risen rapidly. (Tönshoff et al, 2001) proposed system architecture based on the application of cooperative agents in order to improve information logistics in the area of process planning and production control. Three different types of agents are used in the proposed system: resource, order, and service agents. Performance measures are chosen as capacity information and due dates to be able to eliminate the problems that result from time delayed return of manufacturing knowledge and capacity data and/or from other lacks of information flows. (Chan et al, 2001) developed an agent-based framework for an integrated, distributed and cooperative process planning system (IDCPPS). All activities of process planning are integrated into a distributed intelligent open environment using CORBA technology with this system. Objective of the study is to minimizing manufacturing cost or lead-time. The characteristics of multi-agent systems are also described in the study.

In 2002, (Wu et al, 2002) presented an integration model based on multi-agent approach in a distributed virtual manufacturing environment. The agents used in the system are as follows: communication facilitator, process planning agent, scheduling agent and common database service agent. The aim of the study is to select the most suitable partner in the virtual enterprise using a cost function which considers the partner manufacturing capability, process requirements, partner's location, processing times and due dates of the products. (Gao et al, 2002) implemented an agent-based system using an Asynchronous Team architecture in which the agents work asynchronously and embody their own methods while making decisions considering number of setups, quantity of late orders and tardiness. Agents in this architecture do not explicitly cooperate with each other, instead they modify a solution in the population created by other agents, and so the cooperation is achieved. Mathematical algorithms, domain dependent heuristics and search algorithms are used for dynamic planning and scheduling of integrated steel processes. There exist construction agents, improvement agents and destruction agents in the system. The objectives were reducing production cost, improving product quality and decreasing lead-time.

(Lim and Zhang, 2003) developed a strategy for responsive manufacturing where multi-agents are used for manufacturing control in order to enable the utilization of manufacturing resources to be dynamically optimized. Resource manager agent, product manager agent, task manager agent, and execution manager agent are used in the framework. Manager agents also have sub-agents under their supervision. Performance criteria of the study are total production cost, machine reliability, total production time and number of bottleneck. The authors later presented an integrated agent-based approach for responsive control of manufacturing resources (Lim and Zhang, 2004).

(Wong et al, 2005) presented an agent-based negotiation approach to IPPS in a job shop kind of flexible manufacturing environment. Makespan, mean flow time, average machine utilization and CPU time are considered as performance criteria. A bidding mechanism is used as in the previous study of " Bidding-based process planning and scheduling in a multi-agent system" (Gu et al, 1997). Machine agents and part agents engage in bidding in order to allocate tasks. Later the bids are evaluated according to a currency function, which considers an agent's multi-objectives and IPPS parameters. Multi-agent negotiation (MAN) approach is compared to Symbiotic evolutionary algorithm (SEA) and results showed that MAN is better in mean flow time and makespan. Machine utilization couldn't be compared because it is not mentioned in SEA in the study of (Kim et al, 2003).

After this study the authors also presented a hybrid-based multi-agent system that integrates process planning with scheduling/rescheduling in job shops or similar kinds of flexible manufacturing environments considering machine breakdowns and new part arrivals with the objective of minimizing the deviations in cost, part flow time and maximizing machine utilization (Wong et al, 2006).

(Shen et al, 2006) reviewed the literature on manufacturing process planning scheduling, and their integration focusing on the ones used agent based approaches. The papers are classified and examined according to approaches they used: Centralized Optimization Algorithms, Close Loop Optimization, Distributed Process-Planning (DPP) Approaches as traditional approaches and Agent-Based Approaches. Advantages of agent-based approaches, their major design and implementation issues such as agent encapsulation, agent modeling, system structures, coordination and negotiation are told. Future research opportunities, as well as challenges, are identified.

(Zattar et al, 2007) developed a hierarchical multi-agent model using operation-based time-extended negotiation protocol in a flexible manufacturing environment using dynamic process planning and scheduling strategy. There are part agents and resource agents, which renegotiate with each other by a mechanism so that the dynamic scheduling environment is achieved. The objective of the study is to reduce makespan and flow time by reducing the setup times between the jobs.

(Shukla et al, 2008) used a bidding based approach for IPPS problem using multi agent structure. The performance criterion was tool cost (tool using cost and tool repairing cost). The agents were: data mining agent, machine agents, component agents and optimization agent. Data mining agent predicts the tool cost. Component agent gives a bid for each feature of the arrived job for each machine agent in order to assign each feature to an appropriate machine. According to this assignment, the optimization agent tries to find the optimal process plan and a schedule, using hybrid TS-SA algorithm.

(Nejad et al, 2008a) proposed a multi-agent based, integrated and dynamic system that consists multi jobs. In order to generate the process plans and schedules of the resources and jobs dynamically, a negotiation protocol is used and combined with heuristic search algorithms. The agents in the system are classified as: physical agents (job agents, machine tool agents and machining process agents) and information agents (production engineering agents and job order agents). The objective of the study is to minimize the completion time. (Nejad et al, 2008b) and (Nejad et al, 2011) later presented a study on agent-based dynamic process planning and scheduling in flexible manufacturing system with the aim of minimizing average flow time.

(Fujii et al, 2008) proposed a system with a multi-agent learning based integration method, in which each machine makes decisions about process planning and scheduling simultaneously using evolutionary artificial neural networks. The objective of the system is to minimize makespan.

(Li et al, 2009b) classified the agents used in the proposed multi-agent based system as job agent, machine agent and an optimization agent. The optimization agent based on an evolutionary algorithm is used to manage the interactions and communications between agents in order to make proper decisions and optimize makespan. (Li et al, 2009c) and (Li et al, 2010c) used job agents, machine agents and an optimization agent in a job shop environment and proposed a mathematical model. The proposed system had operational, sequential and processing flexibilities in process planning part. The performance criteria were production time in process planning part and the makespan in the scheduling part.

(Wang et al, 2009b) developed a distributed process planning and scheduling system using an agent-based approach to allow geographically dispersed entities to work cooperatively towards the global goals. The agents are composed of resource agent, process identification agent, production monitor agent, process planning agent, scheduling agent and facilitator agent. Particle Swarm Optimization algorithm is used to optimize the performance criteria total manufacturing cost.

(Leung et al, 2009) presented an ACO algorithm for the integration of process planning and scheduling in an agent-based system with the objective of minimizing makespan. A disjunctive graph is implemented as a path for intelligent ants to walk on. For each part the graph presents the precedence relations between operations, alternative process routes and alternative machines. The proposed ACO algorithm, which is under the responsibility of a supervisor agent, has mainly four stages: Initialization, Iteration, Control, and Termination and the most computational exhaustive stage “Iteration”, will be delegated to multiple local agents. As the result of the algorithm the ants will gather on the path with minimum makespan.

(Pawlewski et al, 2009) introduced a multi-agent approach in supply chain that involves hybrid solutions considering the advantages of MRP simple logic and theory of constrains (TOC) ability to synchronize all production and material flow in supply chain. The system consisted of four main types of agents distributed in hierarchical levels. GUI- Agent and MP-Agent are in Supply Chain layer, M-Agents in Company layer and CI-Agents in Intra-Company layer. GUI-Agent sends behavior parameters and messages to Master Planning Agent (MP-Agent). (MP-Agent) gets all the data from managing agents (M-agents) and generate re-planning schedules for the production, distributors or supplier. (M-agents) are responsible for translation of the global plan into detail schedules. CI-Agents are responsible for control of plans execution within one company based on given performance indicators. The aim of the paper is to overcome the limitation of standard MRP/ERP systems in changing environment.

(Zattar et al, 2010) developed a multi-agent system model, using operation-based time-extended negotiation protocols, in which the operations of manufacturing parts are grouped by a tracing mechanism in order to reduce set up times. The objective of this reduction is to minimize makespan and flowtime. Four types of flexibility are studied to determine with which one the system performs better. By the time-extended negotiation protocols, all times while manufacturing a part can be visualized and the deadline becomes a fixed percentage of the expected time that will be required to set-up and process the job on the current resource. The system performance is compared with SEA developed by (Kim et al, 2003). Results showed that the proposed model provides the best makespan performance among the compared algorithms.

The most selected performance criterion is minimizing makespan and flow time in the above studies. The other performance criteria are due date performance, manufacturing cost, machine reliability, total production time and number of bottlenecks.

#### 2.3.2.2 The Algorithm-Based Approach

Algorithm is a step-by-step solution procedure for a problem, consist of well defined instructions that describe a computation that starts from the initial state, and goes on the successive states till it reaches the termination state. The transition between states can be deterministic as well as it can be randomized.

Many researchers used this approach for integration process adopting different algorithms. The approach works like that: The scheduling function uses the developed algorithm in order to simulate the generated process plans. According the results of the simulation, a predefined number of plans are selected (Li et al, 2010a).

Some of the algorithms used in recent studies are:

##### 2.3.2.2.1 Genetic Algorithm (GA)

(Morad and Zalzala, 1999) presented a study consists of two parts, using genetic algorithm for integration. In the first part authors consider only the time aspect of alternative machines. In the second part, processing costs and the processing capabilities of alternative machines, with different tolerance limits are also considered. Makespan, total cost and the number of total rejects produces are the performance criteria of the study to be minimized. They adopted the plan resource crossover of (Uckun et al, 1993); Order-based operators, Position-based operators and Plan-resource operators are used for crossovers. The results are compared by traditional approach (SPT). GA outperformed the traditional sequential method.

(Lee and Kim, 2001) studied the IPPS using simulation based genetic algorithms. For each part the alternative process plans are generated by a CAPP system and one alternative process plans is randomly selected for simulation module. From this module, a schedule for the process plan combination and the corresponding performance measure is obtained. Process plan combination is converted in to a string, which keeps all the necessary data represented by the position of the bits. Genetic Algorithm based mechanism takes the performance measure as fitness function and evolves it until no change can be made that leads to a cost (fitness) reduction.

(Moon et al, 2003) formulated the IPPS problem as a 0 – 1 integer programming method using a genetic algorithm based heuristic approach. The authors considered alternative machines and sequences, sequence dependent setup, and distinct due dates in their study. The performance criterion was minimizing the total tardiness.

(Kim et al, 2003) proposed a symbiotic algorithm for IPPS problem. The authors used efficient genetic representations and operator schemes in their study. The strategies of steady-state reproduction, localized interactions and random symbiotic partner selection are used in the proposed method.

(Zhao et al, 2004) used a fuzzy inference system FIS for IPPS in a job shop manufacturing system. The fuzzy inference mechanism gets the MTF values as input and returns the reliability values of machines. The machine with highest reliability has the higher probability to be chosen. The machine loads is balanced using genetic algorithms. As a result, the number of un-utilized machines is reduced. Two crossover operators used in the formulation are order-based operator and plan-resource operator like (Morad and Zalzala, 1999). The study has multiple objectives: minimize makespan, total number of rejects and total processing cost. The authors revised the study, in year 2006, with PSO algorithm that is used to balance the load for all the machines and holonic architecture is adopted (Zhao et al, 2006). In 2010 the study is improved by (Zhao et al, 2010), a hybrid particle swarm optimization algorithm is presented with differential evolution (DE) has been applied to balance the load for all the machines, minimize number of rejects, processing cost manufacturing lead time and total completion time.

(Moon and Seo, 2005) formulated the IPPS problem in a flexible multi plant environment as a mathematical model and used genetic algorithm to solve the model and minimize makespan. They treated the operations sequencing problem (OSP) as multiple travelling salesman problems (TSPs), each of which determines the machine operations sequence for each part type.

(Park and Choi, 2006a) studied on IPPS in a Job Shop using a GA based method. Each part of a product is considered as job, which is composed of operations with/without precedence constraints in the study. The information about of operations belong to jobs are encoded in a gene consist of three rows, in the first row the order of operations are represented, in the second row the selected alternative sequence is represented and in the last row alternative machine for the operations of the specified jobs in the first row is mentioned if there exists. Objective of the study is to minimize makespan so that responding the due date demands of the customers more effectively. (Park and Choi, 2006b) also presented a conference about their study in 2006.

(Li et al, 2008b) developed an approach for IPPS functions using genetic algorithms. Scheduling is encoded in an operation-based representation, in which all offspring formed by crossover are feasible solutions. Makespan is the optimization criterion of the proposed study and job-shop scheduling type is adopted. Two-point swapping mutation, changing one job's alternative process plan and the mutation of alternative machines are used as different mutation operators. The proposed integration model is compared with no-integration model and two experimental studies are done for illustration. By (Shao et al, 2009), a new integration model and a modified genetic algorithm-based approach have been developed considering makespan minimization. In this model, the alternative process plans are taken from CAPP system to be optimized by GA and the near optimal process plans are found. "S" number of near optimal process plans are selected and then, the integration of process plan and scheduling is optimized by GA. (Zhanjie and Ju, 2008) also applied GA on their research about IPPS considering load balancing, lowest processing costs and the shortest processing time. In year 2010, (Li et al, 2010b) made improvements on this topic and integrate Tabu Search to their study as a local search algorithm. The Hybrid algorithm applied in the study combined the advantages of GA and TS in order to solve the IPPS problem more efficiently. According to this model, every individual evolves by the genetic operators firstly, and then it focuses on the local search.

(Cai et al, 2009) A cross-machine adaptive setup planning (ASP) approach is presented in the study using genetic algorithms for machines with different configurations. ASP implants the adaptive process plans into 3-axis-based generic setups using GA. Then, dynamic scheduling makes request to ASP for generating machine specific setups. The performance criteria in the proposed study are cost, makespan and machine utilization. Also, machine capability and configurations are considered by during the setup-planning phase. The fitness evaluation is accomplished by a primary locating surface (PLS) based post-processing after each GA operation. Iteratively, an optimal or near-optimal setup plan can be obtained.

GA has ability to find generally good global solutions, but not very good at finding the absolute optimum so combining GA with other optimization methods is seem to be quite effective by the researchers. Also for comparing the results, studies containing more than one method have been done. (Zhang et al, 2003a) studied the IPPS problem in a batch-manufacturing environment considering the performance criteria of machine utilization and number of tardy jobs. The proposed system is composed of three modules, which are “the process planning module, scheduling module and the facilitator module”. The CAD model of a given part is used to generate the solution space using GA or SA in the process-planning module. The schedule is formed in the scheduling module, considering shortest processing time, the earliest due date and weights. The other optimization values; machine utilization and job tardiness are generated by the facilitator module and displayed on the screen with graphics.

(Lihong and Shengping, 2012) developed a mathematical model and an improved genetic algorithm with new initial selection methods and new genetic representations. The study achieved improvement in minimizing makespan and mean flow time. The authors wanted to extend the study with additional performance criteria and want to consider the possible interruptions caused by added orders, shortage of material and machine breakdowns in their future studies.

The IPPS problem is formulated as a mathematical model and a hybrid GA is developed in the study of (Amin-Naseri and Afsari, 2011). Authors presented a novel neighborhood function, which considers the constraints of a flexible job shop environment and nonlinear precedence relations among operations. The objective is to minimize the total processing time.

The studies showed that the issue most dealt with is minimizing makespan. Total rejects produced, total cost of production and total tardiness are some of the other performance criteria that are studied.

#### 2.3.2.2.2 Particle Swarm Optimization Algorithm (PSO)

(Zhao et al, 2006) proposed an improved PSO algorithm-based approach, which has a fuzzy inference system for IPPS in holonic manufacturing system. The authors revised their previous study (Zhao et al, 2004), this time with PSO algorithm is used to balance the load for all the machines and holonic architecture is adopted. This architecture is discussed in detail in section 2.3.2.3. The objectives are same with the previous study.

(Guo et al, 2009a) presented a combinatorial optimization model of IPPS problem and used PSO algorithm for solving it considering the performance criteria: makespan, balanced level of machine utilization, total job tardiness and fixed penalty time. Each particle in PSO algorithm represents a process plan and by flying in the search space they tried to achieve best sequence. The algorithm is then modified by the adjustment of new operators such as “mutation, crossover and shift” in order to obtain better results and to avoid being stuck in a local optima. Authors used PSO algorithm in their other studies to solve IPPS problems (Guo et al, 2009b).

Based on their previous studies on IPPS problem (Zhao et al, 2010c) developed a hybrid PSO algorithm and a fuzzy inference system in holonic manufacturing systems. Differently from the previous studies, hybrid PSO algorithm with differential evolution (DE) is used to balance the load for all the machines. DE is used because of its capability of memorizing the best solution and sharing the group information in the candidate.

### 2.3.2.2.3 Simulated Annealing (SA)

(Palmer, 1996) proposed a simulated annealing approach for IPPS and compared the performance of the proposed method with dispatching rules. The performance criteria are tardiness, mean flowtime and makespan. The results showed that SA is remaining high across varying situations and outperformed the use of dispatching rules.

(Li and McMahon, 2007) used SA algorithm for IPPS problem and its optimization considering processing flexibility, operation sequencing flexibility and scheduling flexibility. Manufacturing cost, makespan, balanced level of machine utilization and part tardiness are chosen as performance criteria to be optimized. The SA algorithm applied in this study is with two different options: in the first one, a single criterion is chosen to guide the searching process based on the feasible process plan for each part; in the second option two or more criteria are incorporated as a simultaneous consideration where the criteria are added up with weights as a single criterion.

(Shukla et al, 2008) used a bidding based approach for IPPS problem using multi agent structure. The optimal process plan and a schedule is tried to be found using hybrid TS-SA algorithm. The detail information is given in section 2.3.2.1.

(Wang et al, 2008) developed a heuristic for tardiness minimization and also to maintain cost of process plan involved in modification of process plan in a batch-manufacturing environment. SA algorithm is used to find optimal process plans for prismatic parts.

### 2.3.2.2.4 Ant Colony Optimization Algorithm (ACO)

(Leung et al, 2009) presented an ACO algorithm for the IPPS problem in an agent-based system. A disjunctive graph is implemented as a path for intelligent ants to walk on. For each part the graph presents the precedence relations between operations, alternative process routes and alternative machines. As the result of the algorithm the ants will gather on the path with minimum makespan.

#### 2.3.2.2.5 Artificial Immune System (AIS)

(Chan et al, 2006) used AIS algorithm embedded with a fuzzy logic controller in order to solve the proposed model of IPPS problem, which encapsulates the outstanding characteristics of outsourcing strategy. The objective in the proposed study is minimizing the makespan as well as considering the due dates of the customer orders. The computational time and convergence rate reduction are considered as the important issues in the study.

#### 2.3.2.2.6 Neural Network

(Ueda et al, 2007) used evolutionary artificial neural networks (EANN) based on an emergent synthesis for simultaneously applying process planning and scheduling. Process plans and production schedules are generated through interaction among local decisions of machine agents. Machines agents have two type of decisions: which product to select from machine buffer and then decides which machine to select for the next operation of the selected product. A three-layered feed forward ANN is employed. Input information to the ANN includes: the product types in each buffer, the occupation rate of other machines' buffers, whether other machines are processing products or not. Outputs from the ANN are the decisions of the machine agents. The weight and threshold values of the ANN are encoded into a gene and update during GA. The simulation results showed that the proposed method achieves an optimal solution.

(Fujii et al, 2008) proposed a system with a multi-agent learning based integration method, in which each machine has been modeled as a learning agent using evolutionary artificial neural networks (EANN) to make decisions. The objective of the system is to minimize makespan.

#### 2.3.2.2.7 Tabu Search Algorithm

(Weintraub et al, 1998) developed a scheduling algorithm that incorporates a Tabu Search procedure in which alternative process plans and routing for jobs are identified. The performance criteria were maximum lateness, manufacturing costs and due date satisfaction.

(Shukla et al, 2008) used a bidding based approach for IPPS problem using multi agent structure. The optimal process plan and a schedule is tried to be found using hybrid TS-SA algorithm. The detail information is given in section in section 2.3.2.1.

#### 2.3.2.2.8 Imperialist Competitive Algorithm (ICA)

ICA is a new population based evolutionary algorithm proposed by (Atashpaz-Gargari and Lucas, 2007). Each solution in ICA is named as "country" and each country has a "power" for evaluating its fitness. The best countries are selected as 'imperialists' and rest is shared among the imperialists as "colonies". An imperialist with its colonies constructs an "empire". ICA terminates when there is only one empire left after iteration many times proceeding through the steps: Assimilation, Imperialistic Competition, Revolution and Elimination.

(Lian et al, 2011) applied this algorithm to IPPS problem with the objective of minimizing makespan considering the flexibility of operation, sequencing and processing. Results showed that ICA could achieve promising results in a reasonable computational time.

#### 2.3.2.2.9 Game Theory Based Algorithm

(Li et al, 2008a) presented a game theory-based cooperation of process planning and scheduling using three game theory strategies: Pareto strategy, Nash strategy and Stackelberg strategy. The results of the applied algorithms SA, PSO and GA are compared considering the performance criteria of makespan, manufacturing cost and balanced utilization of machines. Results show that in different situations, different strategies outperforms. SA is fast but vigilant to its parameters. GA and PSO are slow but robust for optimization problems.

(Li et al, 2012) focused on multiple objective IPPS problem and developed a game theory based algorithm using the Nash equilibrium. Three criteria are optimized simultaneously; work efficiency, utilization of existing resources and the total workload of machines. This new idea of applying Game Theory on IPPS gave promising contributions.

#### 2.3.2.3 Holonic architecture

Literally, holon gets its name from the Greek word “holos” which means, “complete, whole and entire”.

A holon is, autonomous, collaborative and has communication abilities like agents but the main difference between agents and holons is that holons can be composed of other holons, which reflects an object-oriented structure. Holons are able to broadcast a message rather than using function calls and each holon is free to accept or ignore the message coming. Number of holons in a Holonic system is also important, too many holons make the configuration of the system hard and on the other hand too few holons cause more interface messages, which means integration difficulty (Zhang et al, 2003).

(Sugimura et al, 2001) developed an IPPS System in a holonic environment to realize a flexible production control. The proposed system consists of two parts: physical processing part and the information processing part. These parts are composed of holons. GA is used to generate machining sequences and then DP is used to select optimum of the generated sequences. The objective is to minimize total machining time.

(Zhang et al, 2003b) also used a holonic architecture in which the holons are able to cooperate with each other to perform an appointed task flexibly. System is composed of three main levels: Initial planning level, in which alternative processing plans are generated; the decision making level, in which one suitable plan is chosen based on current status; and detailed planning level in which a detailed PP of the selected plan is generated. All those steps are done via holons communicating and interacting with each other. Also there exist a task holon, C/P coordination holon, P/P coordination holon, resource holon in the system which work cooperatively. The aim of the study was to integrate CAD, PP, and production scheduling system in order to optimize due-date performance and minimize manufacturing cost or lead-time.

(Sugimura et al, 2003) improved their study in 2001 by formulating an objective function based on shop time and machining cost of products. As in the previous study of the authors, GA is used to generate machining sequences and then DP is used to select optimum of the generated sequences. Holonic Manufacturing Systems

Architecture used for IPPS is discussed. Later (Sugimura et al, 2006) continue their studies and presented a book chapter.

(Zhao et al, 2006) presented holonic architecture for the dynamic scheduling of Manufacturing Systems. The alternative machine selection is done by a fuzzy interference system instead of random selection. The system includes scheduling holon, task manager holon, task holons and resource holons. The scheduling holon has two parts: one concerning resources and another concerning tasks. Task Manager holon receives orders for new tasks for manufacturing system. Task holons are launched by task manager holon when a new task order is received. Resource holons represent the current situation of one resource. The information about resources such as the sequence of operations, expected durations of each operation, free time intervals are kept in an agenda. In 2010 the study is improved by (Zhao et al, 2010). The load of all the machines is balanced using a hybrid particle swarm optimization algorithm with differential evolution (DE). DE is a population-based globally evolutionary algorithm, which has the ability to memorize the best solution and share the group information in the candidate. DE creates new candidate solutions using a simple operator and select new candidates greedily using one-to-one competition scheme.

(Shresta et al, 2008) studied on IPPS problem for holonic manufacturing systems considering makespan, total machining cost and weighted tardiness cost as performance criteria. Holonic architecture is used where the job holons generate process plans and scheduling holons make the schedules. GA is used in process planning part and a set of dispatching rules is used in scheduling part.

#### 2.3.2.4 Object Oriented Architecture

An Object-oriented approach is composed of objects, which are data structures each has distinct tasks, communicating with each other using function calls. Objects are passive in contrast to agents and holons. By the inheritance property of object-oriented architecture, creating new objects from existing objects is possible, so that the subclasses inherit the properties of its super class.

(Zhang and Zhang, 1999) presented a simulation study using an Object Oriented Integration Test-bed for PPS. Separated planning method and integrated planning methods are examined. Simulation is done using fixed and variable processing times. In fixed processing time simulation, the preference of selecting machines will depend solely on technical considerations. Separated planning method, Static loading feedback method and Dynamic loading feedback method are used as planning algorithms. Simulation with variable processing times has two different planning algorithms: Simulation with separate selection rules, Simulation switching selection rules. The object-oriented models for the integration test-bed include an object class model, a dynamic model, and a functional model. In this simplified object class design, the objects include part, machine, process plan, and dynamic decision table. First, the operational requirements of a part are determined; then, the machine selection is done using the dynamic decision table. When parts in the buffer reach the buffer size, they are loaded onto machines. Dynamic loading index for the machines is computed and decision table is modified using the dynamic loading index.

(Baker and Maropoulos, 2000) developed a new application VITool, for the vertical integration of tooling considerations, using an object-oriented architecture. Five-level tool selection procedure is applied, which is mapped to a time-phased aggregate, management and detailed process planning framework. Concurrent engineering principles are adopted. Capability, power, tool availability, spindle speed, feed rate, axis suitability, cost and utilization are considered as performance criteria.

(Kis et al, 2000) presented a new Petri net (Chameleon systems) based model for modeling and analyzing joined process planning and machine scheduling using two level timed object Petri nets. The state of the two-level-object-system is given by the arrangement of the token nets on the places of the system net and the arrangement of the black tokens on the places of the token nets. The makespan was minimized. Through the agency of this approach, modular representation of the process plan alternatives and the job-shop become possible, classical known Petri net analysis methods become applicable and the analysis can be done at different levels.

(Hashemipour, 2004) developed a CAPP tool for the apparel industry. The software PPSGAR is built in an object-oriented architecture. When a new order is received, the required data is obtained from an operation table in the database and the process planner object generates an activity network diagram of the tasks in order to determine the tasks, which can be done simultaneously. Based on the network diagram, a Gantt Chart is drawn and with the processing times data, these are sent as an input to Scheduling object as well as the resource and probability distribution data, in order to get the output data of workstations, efficiency and balanced delay. The possible changes are done and the process planner object is restarted till the best assembly line is selected from the complete result.

### 2.3.2.5 Other Models

#### 2.3.2.5.1 Blackboard Architecture

As it is mentioned before in section 2.3.2.1, (Hildum et al, 1997) used the idea of applying blackboard architecture with agent technology.

(Sadeh et al, 1998) described the IP3S system, an Integrated Process Planning/Production Scheduling shell for agile manufacturing considering the due dates. The problem solving knowledge is encapsulated in independent knowledge sources (KSs) using blackboard architecture. The KS modules uses a data structure called "blackboard" in order to make required communications while generating solutions to the problems. The blackboard operates as a server in the system. The "controller" and GUI operate together as a client. Each KSs in the system operates separately and changes the roles of server and client. The blackboard, the controller/GUI and KSs are all run independently and communicate with each other using a CORBA (Common Object Request Broker Architecture)-based environment. The resulting shell provides a customizable framework, which is able to support a wide range of IPPS decision flows.

#### 2.3.2.5.2 Grammatical Optimization Approach

In order to solve IPPS problem (Baykasoglu and Ozbakir, 2009) proposed an approach in which the generic process plans are represented using special grammar rules. The proposed model is composed of two integrated and concurrently acting parts: In the first part scheduling instances are generated using generic process plan grammar and in the second part feasible schedules are generated using Giffler and Thompson's dispatching rule based heuristic. The optimization algorithm is chosen as Multiple Objective Tabu Search algorithm and the performance measures are total flow time and the total cost of process plans.

### 2.3.2.5.3 Web-Based Approach

(Wang et al, 1998) presented a web-based approach for IPPS using heuristics. The objective was to reduce tardy jobs exploring the process plan solution space of the selected jobs. The heuristics used in the proposed study were: the FH-tardy (fine tunning) and the QH-tardy (quick tunning) to generate constraints. This web-based approach provides solutions faster with a reduced cost and fewer resources. Also geographically dispersed departments are enabled for cooperation with each other in a distributed, transactional and portable environment. Job tardiness and cost were considered as the performance criteria.

### 2.3.2.5.4 Mathematical Modeling

(Kim and Egbelu, 1998), developed a mixed integer programming model for scheduling a set of jobs through a shop when each job is supplied or provided with multiple process plans or process routings. For each job a process plan is selected from the set of multiple possible process plans and the scheduling is done according to the selected plan simultaneously with the objection to minimize makespan. Initially, all process plan combinations are given the system as input. After a bounding procedure is applied; a process plan “$S_n$” with the lowest lower bound on makespan is selected. The chosen process plan is solved by mathematical method to obtain best solution “T”. For any process plan combination “$Sj$”, whose lower bound is smaller than T , obtain the best solution $Tj$ using the mathematical method. If $Tj < T$ , let $T = Tj$ and $Y = Sn$ . Continue this step until no process plan combinations whose $LB < T$ exist. Stop and $Y$ is the best solution of the R total process plan combinations. The corresponding best makespan is $T$. (Kim and Egbelu, 1999) also presented a study using mathematical modeling on Scheduling in a production environment with multiple process plans per job.

A linearized polynomial mixed integer programming model (PMIPM) for IPPS problem is presented by (Tan and Khoshnevis, 2004). The terms of the model are both binary and continuous in nature. Using certain linearization techniques the polynomial model is linearized so that the equivalent model provides the opportunity to obtain global optimal solutions, which is difficult to achieve with a polynomial model. Minimizing total cost, total completion time of all features, makespan, work in process, number of tardy jobs and mean tardiness are determined as objectives of the study.

(Moon and Seo, 2005) proposed an advanced process planning and scheduling model for the multi-plant. The problem is formulated as a mathematical model, and an evolutionary algorithm (genetic algorithm) is developed to solve the model. The operation sequencing problem is formulated as a traveling salesman problem (TSP) since the transition time is sequence dependent. The aim of the study is to minimize makespan by determining the optimal schedule of machine assignments and operation sequences. With the same objective in his study, (Moon et al, 2008) presented an evolutionary search approach based on a topological sort (TS) to solve the problem in a supply chain. TS is used for determining the operation sequence for an order and genetic algorithm is chosen as evolutionary search approach. Problem is represented as a mixed integer-programming model and solved by the proposed approach.

(Omar and Teo, 2007) proposed a mixed-integer linear programming formulation for aggregate production planning (APP) model. Job scheduling and day-to-day activities are controlled at the shop floor and explicitly integrated with APP. Also a job-sequencing model (JSM) is developed using a 0-1 integer-programming model. Tardiness/earliness minimization and set-up time among product families are considered as performance criteria in the proposed study.

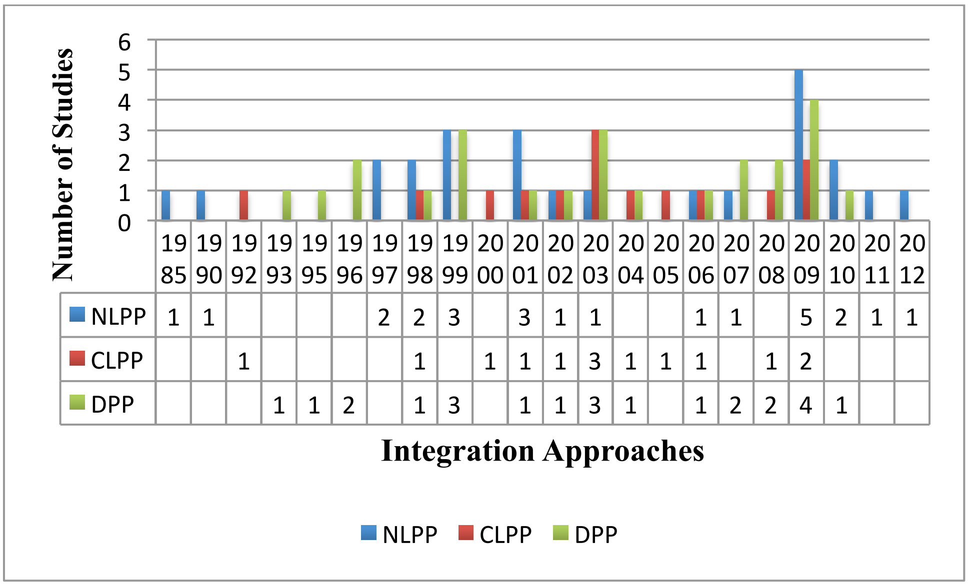
(Nejad et al, 2008b) proposed a system using mathematical modeling in an agent-based structure. The system coordination is performed by coordination agents. All received proposals from the machine tool agents are scanned by the coordination agents. Then the appropriate machine tool agents are assigned to the job agents. This coordination problem is represented as an integer-programming model with the objective of minimizing the average flow time of all the job agents.

(Li et al, 2009a) formulated a mathematical model of IPPS and an evolutionary algorithm-based approach is used for solving it. To improve the optimized performance of the approach, efficient genetic representation and operator schemes also have been developed. The objective of the study was to minimize makespan and cost.

## 2.4 Conclusion

This chapter presents a literature survey on integrated process planning and scheduling (IPPS). Ninety-five articles are reviewed in this study. The reviewed papers are discussed in detail and a statistical study of them is presented. The methods used, architectures, integration approaches, implementation models and objectives are taken into consideration.

When the integration approaches of the reviewed papers are considered, it is shown in Figure4 that 40% of the studies used NLPP approach and 38%of them used DPP approach and the rest used CLPP with 22%. In Figure5 the distribution of the studies over years according to the integration approaches is shown. As it is derived from the figure, the researchers have used CLPP approach more in the last few years when it is compared to the 90s. NLPP and DPP have always been popular but in the latest years, the number of studies on these approaches both showed an increase.



| | 1985 | 1990 | 1992 | 1993 | 1995 | 1996 | 1997 | 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 | 2010 | 2011 | 2012 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NLPP | 1 | 1 | | | | | 2 | 2 | 3 | | 3 | 1 | 1 | | | 1 | 1 | | 5 | 2 | 1 | 1 |
| CLPP | | | 1 | | | | | 1 | | 1 | 1 | 1 | 3 | 1 | 1 | 1 | | 1 | 2 | | | |
| DPP | | | | 1 | 1 | 2 | | 1 | 3 | | 1 | 1 | 3 | 1 | | 1 | 2 | 2 | 4 | 1 | | |

**Figure 5** Distribution of the studies over years according to the integration approaches

Figure6 shows the portion of studies in bars considering the implementation models /architectures used. Multi-Agent-Based and Algorithm-Based approaches have been the most studied issues.

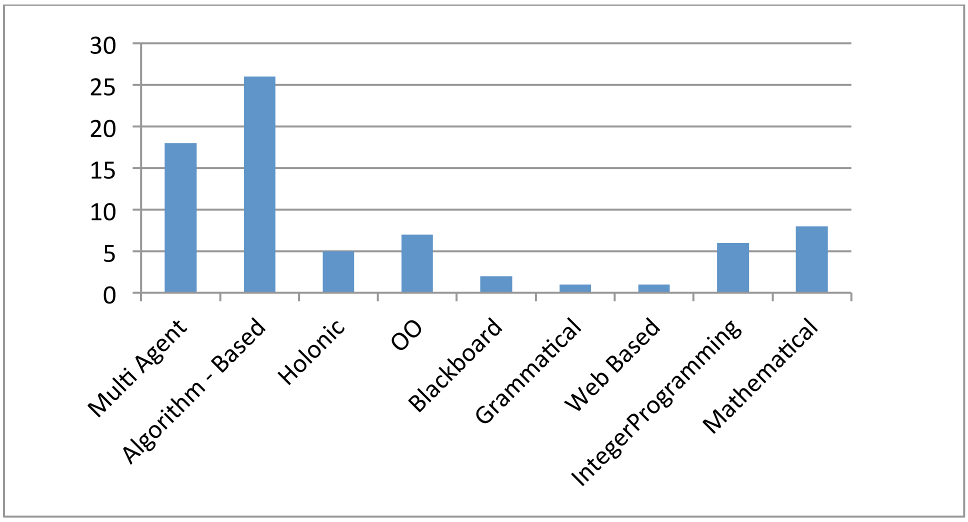


**Figure 6** Number of studies considering implementation models / architectures

The algorithm-based approaches are also subdivided to sections in itself as GA, SA, ACO, AIS, Hybrid and Neural Network. The percentage of each section is shown in Figure7.

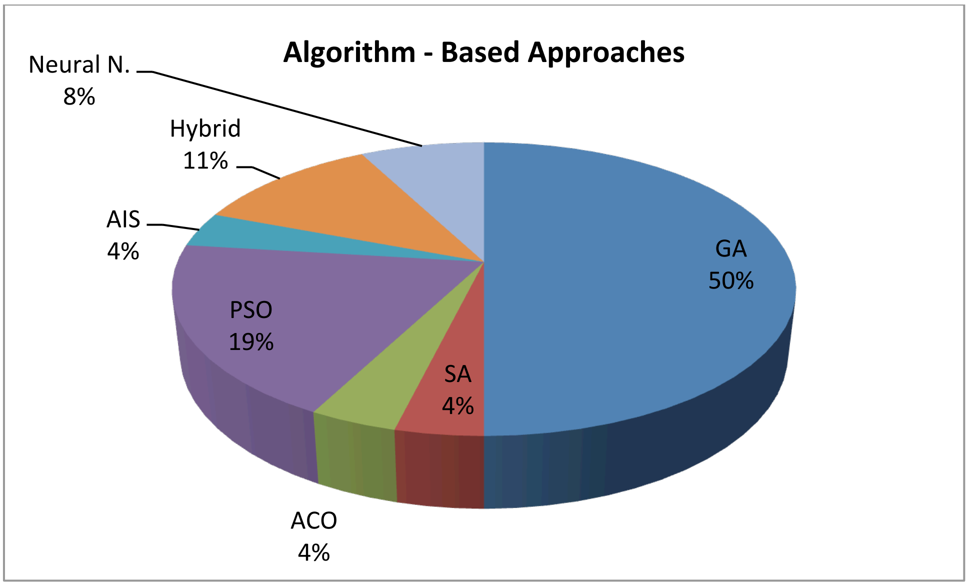


**Figure 7** Percentage of different algorithms in algorithm-based approaches

Even the years between the last review and this study is very near, there is an inevitable need of this review on the subject. Figure8 shows that, the number of studies during the latest years, have increased. As a result of growing number of studies on IPPS problem in recent years, some papers haven't been considered by the previous reviews due to fast adjustments to the literature. This shows the importance of this review in terms of catching up the latest developments on the issue.

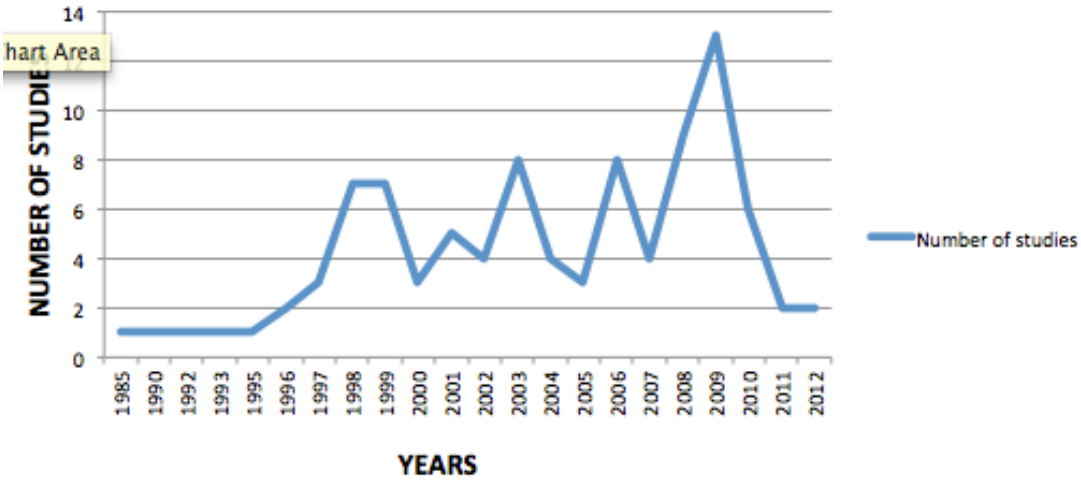


**Figure 8** Number of studies over years

The literature survey showed that, the most considered performance criterion is *makespan* and the next popular criterion is *cost.* Forty-three of the studies on IPPS selected *makespan* and twenty-nine of the studies selected *cost* as one of their performance criteria. Forty-two of the studies used only one performance criterion to be optimized and the rest used multi-criteria. The constraints such as machine capacity and setup times have not been considered on the IPPS problem, which is proposed in this thesis.

# CHAPTER 3

# METHODOLOGIES USED IN THE PROPOSED METHOD

## 3.1 Introduction

In this chapter, the methodologies, which are benefited during the implementation of the proposed method, are briefly explained.

Baykasoglu's grammatical representation method and Clonal Selection Algorithm (CLONALG) are used in order to integrate process planning and job shop scheduling. In the proposed integration approach, Giffler & Thompson Algorithm (used to generate active schedules) and VIKOR Method (used to handle multiple objectives) are employed considering setup times and machine capacity constraints.

## 3.2 Baykasoglu's Grammatical Representation Of Generic Process Plans

In order to solve IPPS problem (Baykasoglu, 2002) proposed an approach in which the generic process plans are represented using special grammar rules. The proposed model comprises two integrated and concurrently acting parts: In the first part, scheduling instances are generated using generic process plan grammar. In the second part, feasible schedules are generated using Giffler and Thompson's dispatching rule based heuristic. The optimization algorithm is chosen as simulated annealing and the performance measures are total flow time and the total cost of process plans.

### 3.2.2 Generic Process Plans

In the proposed approach generic process plans are used as the central medium to be able to solve IPPS problems.

The generic process plan represents only the task requirements; the operations-resources assignments are not included in the plan. An operation can be performed by any actual resource, which has the necessary capability. The final process plan represents all planning and scheduling activities (Baykasoglu and Ozbakir, 2009).

(Baykasoglu and Ozbakir, 2009) considered only formative tasks such as such as milling, painting, assembling, in their study. The information for GPP can also be represented using context free grammars (Fu 1974). The syntax of the data is defined by a grammar, which has four tuples:

$G_s=\{\Sigma_S , \Pi_S, Ps, S\}$

Productions (PS) are of the form:

$$\Phi \rightarrow \Delta \;, \quad \Omega_i \xrightarrow{\pi_{ij}} \Psi_{ij} \;, \quad \Theta_i k \xrightarrow{\mu^{k}_{ij}} \Lambda_{ij}$$

$\Phi$, $\Omega_i$ , $\Theta_i$ are non-terminals; $\Delta$, $\Psi_{ij}$ , are sets of non-terminals, $\Lambda_{ij}$ are terminals; $\pi_{ij}$ and $\mu^{k}{}_{ij}$ are controls (Baykasoglu, 2002).

### 3.2.3 The Case Problem of (Baykasoglu, 2002)

In this thesis, the same problem of (Baykasoglu, 2002) in which there are two parts with their corresponding feasible operation sequences defined in terms of three operations, which can be performed on three machines, is considered as an example. The number of parts and their alternative operation sequences are given in Table 1.

**Table 1** Part and operation sequence information

| PARTS | Alternatives | Feasible Operation Sequences |
|---|---|---|
| P1 | 1 | O1 → O2 → O3 |
| | 2 | O3 → O1 → O2 |
| P2 | 1 | O1 → O3 → O2 |
| | 2 | O2 → O3 → O1 |

The data of which operation can be processed on which machine is given in Table 2.

**Table 2** Machine-operation suitability information

| Machine /Operation | O1 | O2 | O3 |
|---|---|---|---|
| M1 | * | | |
| M2 | * | * | * |
| M3 | | * | |

Each operation that belongs to a particular part has a different processing time on the different machines. The data of processing times are given in Table 3.

**Table 3** Processing time information

| **Operations: Machines** | **O1** | | **O2** | | **O3** | |
|---|---|---|---|---|---|---|
| | **P1** | **P2** | **P1** | **P2** | **P1** | **P2** |
| **M1** | 2 | 4 | | | | |
| **M2** | 4 | 3 | 2 | 1 | 2 | 2 |
| **M3** | | | 3 | 2 | | |

The grammar (Generic Process Plan) generated for the problem above is:

$$\Sigma_s = \{M1,M2,M3\}$$

$$\Pi_s = \{P1,P2,O1,O2,O3\}$$

$P1 \xrightarrow{\pi 11} O1_{P1},\ O2_{P1},\ O3_{P1}$

$P1 \xrightarrow{\pi 12} O3_{P1},\ O1_{P1},\ O2_{P1}$

$P2 \xrightarrow{\pi 21} O1_{P2},\ O3_{P2},\ O2_{P2}$

$P2 \xrightarrow{\pi 22} O2_{P2},\ O3_{P2},\ O1_{P2}$

$O1_{P1} \xrightarrow{\mu 11^{1}} M1(2)$

$O1_{P1} \xrightarrow{\mu 12^{1}} M2(4)$

$O1_{P2} \xrightarrow{\mu 11^{2}} M1(4)$

$O1_{P2}\xrightarrow{\mu12^{2}}M2(3)$

$O2_{P1}\xrightarrow{\mu21^{1}}M2(2)$

$O2_{P1}\xrightarrow{\mu22^{1}}M3(3)$

$O2_{P2}\xrightarrow{\mu21^{2}}M2(1)$

$O2_{P2}\xrightarrow{\mu22^{2}}M3(2)$

$O3_{P1}\xrightarrow{\mu31^{1}}M2(2)$

$O3_{P1}\xrightarrow{\mu32^{1}}M2(2)$

$O3_{P2}\xrightarrow{\mu31^{2}}M2(2)$

$O3_{P2}\xrightarrow{\mu32^{2}}M2(2)$

The final process plan can be obtained by selecting random set of controls for each part and their corresponding operations:

$$\{ [ (\boldsymbol{\pi}_{11}), \boldsymbol{\mu}_{11}{}^{1}, \boldsymbol{\mu}_{22}{}^{1}, \boldsymbol{\mu}_{32}{}^{1} ] , [ (\boldsymbol{\pi}_{21}), \boldsymbol{\mu}_{11}{}^{2}, \boldsymbol{\mu}_{31}{}^{2}, \boldsymbol{\mu}_{21}{}^{2} ] \}$$

P1→M1(2), M3(3), M2(2)
P2→M1(4), M2(2), M2(1)

Using different set of controls, different final process plans can be generated. If a scheduling algorithm with suitable dispatching rules is used while selecting the controls, this method can be a solution for IPPS problem.

### 3.3 Giffler And Thompson Algorithm

In 1967, Conway made the definition of scheduling as; “ sequencing the operations of all jobs (products) based on precedence constraints, considering time aspects” (Conway et al, 1967). Many algorithms have been developed so far by the researchers in order to minimize the length of the production schedules. Giffler and Thompson proposed an algorithm, which is called by their name ‘Giffler & Thompson Algorithm’ in 1960.

### 3.3.1 The Notation For the Algorithm Proposed by Giffler & Thompson

Given a problem with *i* operations,

$PS_t$ = Partial schedule
$S_t$ = The set of operations which can be scheduled at iteration *t*
$\sigma_i$ = the Earliest starting time at operation *i* can start
$\Phi_i$ = Earliest time at operation *i* can be completed
$C_t$ = Set of conflicting operations in iteration *t*
$p_t$ = Dispatching rule chosen in iteration *t*

### 3.3.2 Steps of Giffler & Thompson Algorithm

**Step 1:** Let $t = 1$. $PS_t$ is null in the first iteration. St includes all schedulable operations which means the operations with no predecessors.

**Step 2:** Determine the operation in $S_t$ which has the earliest completion time, $\Phi_t^* = \min_{i \in St}\{\Phi_i\}$. Determine the machine $m^*$ which can process that operation with the earliest completion time. If the operation can be processed on another machine, choose one of them randomly.

**Step 3:** Form conflicting set $C_t$ which includes all operations $i \in St$ with $\sigma_i \leq \Phi_i^*$ that requires machine $m^*$. Select one operation from Ct with respect to the dispatching rule $p_t$ and add this operation $PS_t$ as early as possible, thus creating new partial schedule $PS_{t+1}$. If more than one operation exists according to the dispatching rule $p_t$, choose one of them randomly.

**Step 4:** Remove the chosen operation from St and add the direct successor of the operation to St. Increment *t* by one.

**Step 5:** Return to step 2 until a complete schedule is generated.

The algorithm is modified in order to consider the machine capacities and setup times. The considerations are embedded in the third step of the algorithm, where an operation is chosen from the $C_t$ and placed in the $PS_t$. The flowchart in Figure9 presents the modified step.

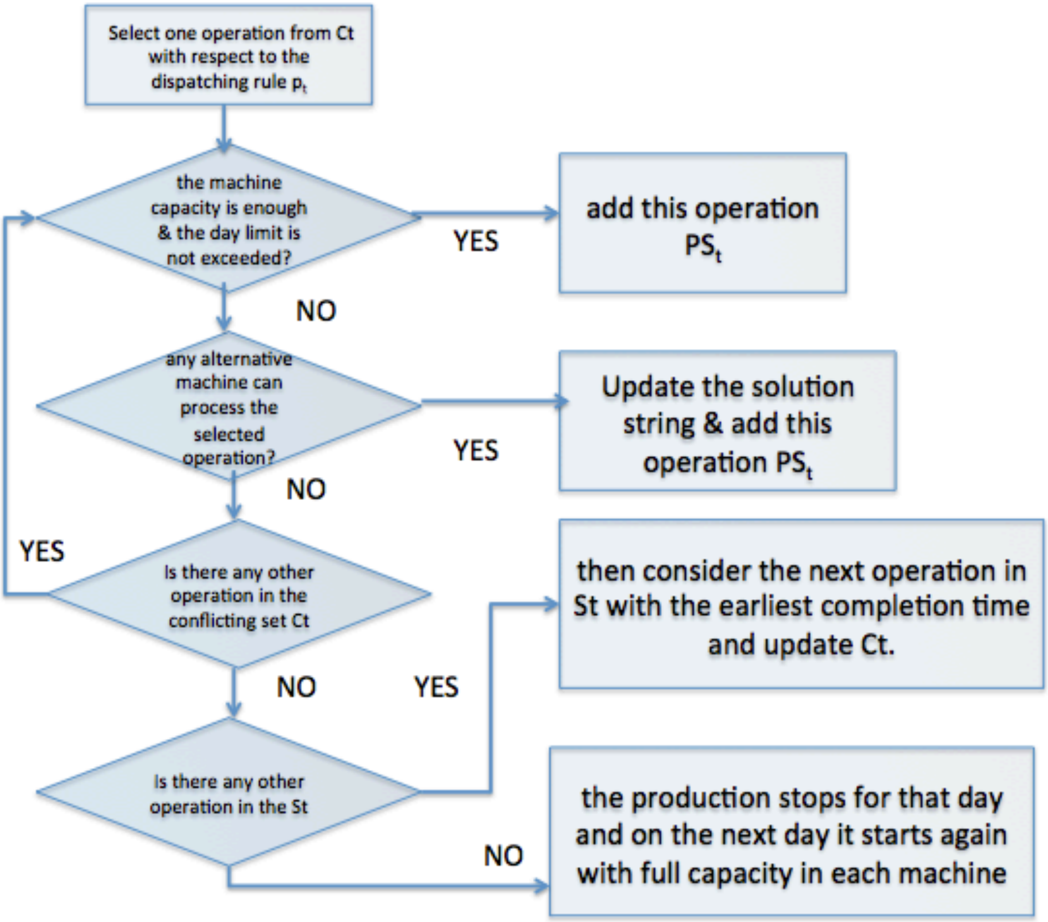


**Figure 9** The setup time and machine capacity considerations on the third step of Giffler & Thompson Algorithm

### 3.4 Artificial Immune System (AIS) Algorithm: CLONALG

Artificial Immune System Algorithm is inspired by the complex and effective vertebrate immune system, which has an ability to adapt the process of recognizing some pathogens more effectively over time. This adaptation requires a memory feature. The algorithms that belong to this group use this property of memory in order to reach the optimal or near optimal solutions from a faster way in searching the complete solution set.

### 3.4.1 The Vertebrate Immune System

An immune system is a system consists of biological structures and processes, which detects and destroys the pathogens (such as bacteria, viruses, incompatible blood cells and man-made molecules…) so that it protects the organism against diseases. Immune system has immune cells (lymphocytes: B-cells/antibody and T-cells) whose mission is to detect and distinguish the pathogens from the organism's own healthy cells.

Artificial Immune System Algorithm is inspired by the complex and effective vertebrate immune system. The vertebrate immune system has an ability to adapt the process of recognizing some pathogens more effectively over time. This adaptation requires a memory feature. Each immune cell has receptor molecules on their surface, which are for recognizing the pathogens. Pathogens also have protein-bodied antigens on their surface. Like a key and lock adjustment, when antigens and receptor molecules have complementary surfaces they can bind together. As a result of the binding action, antigen is recognized and the immune system starts to response (Engin and Döyen, 2007).

There are two important mechanisms of immune systems: clonal selection and affinity maturation. After the antigen and the receptor molecules bind together, the antigen stimulates the B-cell to divide into many terminal plasma cells (De Castro and Von Zuben, 2000). This process is named as proliferation. The proliferation degree is directly proportional to its affinity. The best-matched cells with antigens are selected for survival by the clonal selection mechanism. This selection can be viewed as a type of Darwinian microcosm (Brownlee, 2005).

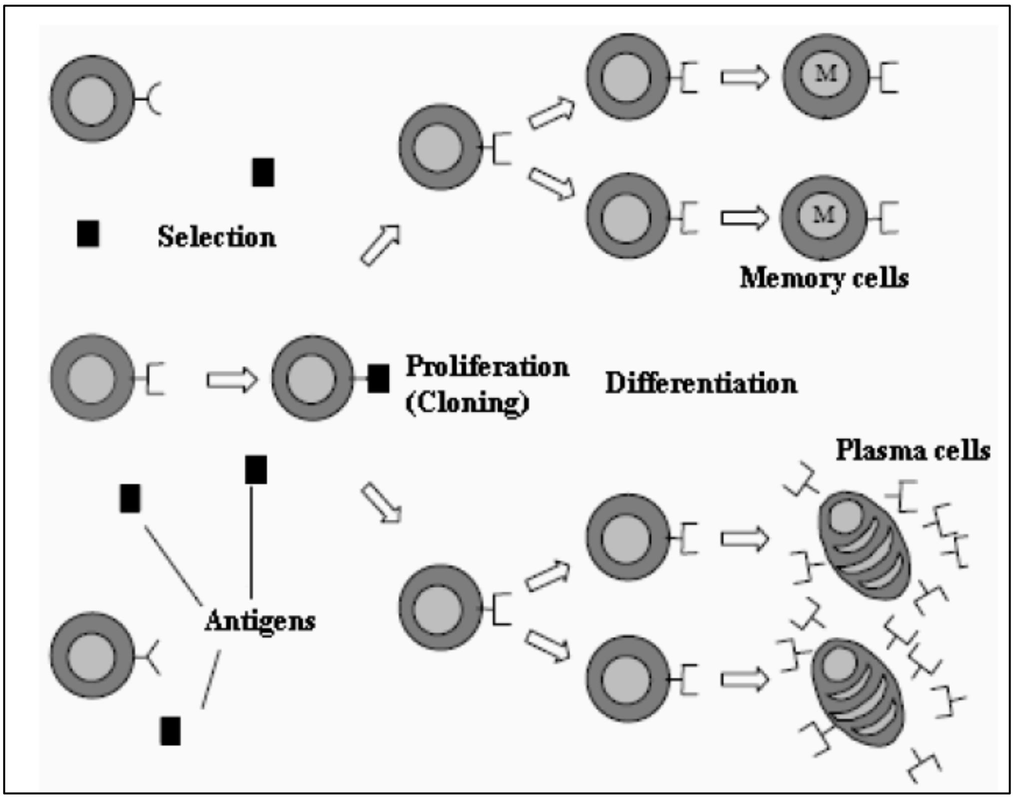


**Figure 10** The Clonal Selection Principle (De Castro and Von Zuben, 2000)

Affinity is the degree affinity of the cell recognition with the antigen. Affinity maturation mechanism provides a mutation process to cloned cells so that they match the antigens in a better way. The maturation degree is inversely proportional to their parent cell's affinity.

### 3.4.2 CLONALG (Clonal Selection Algorithm)

The algorithm used in this study is CLONALG, which is an artificial immune system algorithm inspired by the vertebrate immune system. The algorithm is proposed by (De Castro and Von Zuben, 2000). In Figure11 the steps of the algorithm are shown.

The steps of CLONALG:

1. The objective function should be optimized. Each member of the population is an antibody ($Ab_i$) that has an affinity value, which corresponds to the objective function.

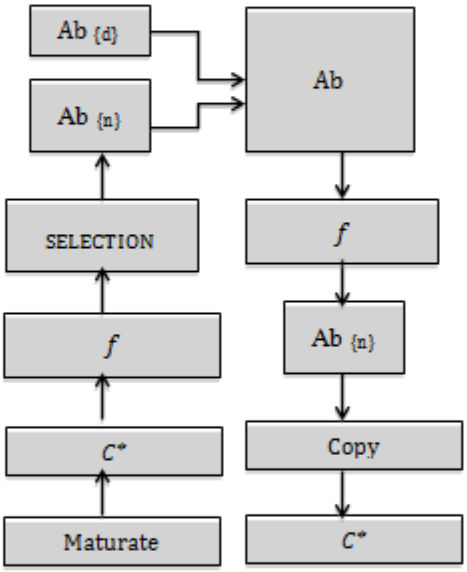


**Figure 11** Steps of the CLONALG algorithm (De Castro and Von Zuben, 2002)

2. The affinity value *f* is calculated for each antibody.
3. The n amount of antibody with the highest affinity values, are selected from the population.
4. The selected n antibodies are cloned proportional to their affinity values and formed a C clone set. The amount of each clone formed for each selected antibody is proportional to the affinity value *f*.
5. The C clone set is maturated (mutated) reversely proportional to their affinity values.
6. The affinity values of the clones are calculated.
7. The n numbers of clones are selected which have the highest affinity values and added to the population set.
8. Finally, d amount of antibodies are selected from the population with the lowest affinity values and replaced by newly formed antibodies.

### 3.5 Multi-Criteria Decision Making (MCDM) – VIKOR Method

Decision analysis deals with how an individual decision maker or a decision group make a rational choice among many alternatives in an uncertain environment.

MCDM is one of the most studied areas of decision analysis, because it is a problem that is faced in every aspect of the daily life. In MCDM, the decision maker tries to make decision where there are many alternatives and each of these alternatives has many decision criteria. Each of these criterions has a weight of importance or priority among other criteria of the same alternative, which is called decision weights. MCDM problem can be categorized under two subtitles: Multi - Objective Decision Making (MODM) and Multi - Attribute Decision Making (MADM). In MODM, the decision space is continuous whereas in MADM, the decision space is discrete (Triantaphyllou et al, 1998).

### 3.5.1 Multi-Criteria Decision Making Methods

There have been many methods developed by the researchers for MCDM problem. Some of them can be listed as:

- The Weighted Sum Model
- The Weighted Product Model
- The Analytic Hierarchy Model (AHP) (Saaty, 1980).
- The Elimination and Choice Translating Reality Method (ELECTRE) (Benayoun et al, 1966)
- The Technique for Order Preference by Similarity to Ideal Solution Method (TOPSIS) (Hwang and Yoon, 1981)
- Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE) (Brans and Mareschal, 1982)
- Vise Kriterijumska Optimizacija I Kompromisno Resenje Method (VIKOR) (Yu, 1973)

### 3.5.2 VIKOR Method

The basis of VIKOR Method was established by (Yu, 1973) and (Zeleny, 1982) and later advocated by (Opricovic and Tzeng, 2002 - 2007) also VIKOR Method is firstly suggested by Opricovic and Tzeng for multi-criteria optimization of complex systems.

VIKOR Method is also known as “Compromise Ranking Method”. The compromise-ranking list is determined and the compromise solution is obtained using the initial weights.

Multi-criteria ranking index is based on the measure of “closeness” to the “ideal” solution. VIKOR Method is considered as one applicable technique to implement within MADM. The Lp-metric used in compromised programming method of (Zeleny, 1982) is followed to obtain the multiple attribute merit.

$L_{pj} = \{ \sum_{i=1}^{n} [w_i (f_i^* - f_{ij}) / (f_i^* - f_i^-)]^p \}^{1/p}$

$1 \leq p \leq \infty$ ;

$j = 1,2,\ldots,J$

n represents the number of criteria.

### 3.5.2.1 The Problem Characteristics & VIKOR Method

To be able to handle the MCDM problems with VIKOR Method, the problems should have these characteristics:

- Compromising is satisfactory for conflict resolution.
- The decision maker (DM) wants to find the closest solution to the ideal one.
- The relationship between each criterion function and a decision maker’s utility should be linear.
- The criteria are conflicting and have different units.
- The alternatives are evaluated according to all established criteria .
- The DM’s choice is given, expressed by weights or simulated.
- The DM’s preference must be included while he/she approves the final solution.
- The VIKOR method can be started without interactive participation of DM.

### 3.5.2.2 Comparison of VIKOR Method with other Methods

#### 3.5.2.2.1 VIKOR Method vs. TOPSIS

In order to figure out the crucial points that differentiate the method, comparison with other methods are done.

(Chu, 2007) compared TOPSIS and VIKOR methods and handled the application of the methods to group decision analysis in information society. (Chu, 2007) mentions that:

- VIKOR provides more alternatives than TOPSIS for decision makers.

Opricovic and Tzeng (2007) compared extended VIKOR Method and TOPSIS. In their studies, they specify that:

- VIKOR uses linear normalization; TOPSIS uses vectoral normalization.
- VIKOR is based on the summation function representing the closeness to the ideal solution; whereas TOPSIS defines two reference points for closeness.

Also TOPSIS does not consider the relative importance of the distances to the two reference points.

#### 3.5.2.2.2 VIKOR Method vs. PROMETHEE

(Opricovic and Tzeng, 2007) compared extended VIKOR Method and PROMETHEE. In their studies, they specify that:

- The results obtained from PROMETHEE are based on “maximum group benefit”
- VIKOR Method combines “maximum group benefit” and “minimum individual regret”.

### 3.5.2.3 Steps of VIKOR Method

#### 3.5.2.3.1 Step 1

In the first step the objective is determined. The best and the worst values of all attributes are identified.

For each criterion, the best (fi*) and the worst (fi-) values are determined.

$$f_i^* = max_j \, f_{ij}$$
$$f_i^- = min_j \, f_{ij}$$
$$i = 1, 2, \dots , n$$

- $f_{ij}$ = evaluation result of alternative $a_j$ according to criteria i

### 3.5.2.3.2 Step 2

$S_j$ and $R_j$ values are calculated.

- $S_j$ =the average group score for jth alternative
- $R_j$ = the worst group score for jth alternative
- $w_i$ = weight of criteria i

$$Sj = \sum_{i=1}^{n} w_i \, (f_i^* - f_{ij})/(f_i^* - f^-)$$
$$Rj = \max\left\lfloor w_i \, (f_i^* - f_{ij})/(f_i^* - f^-)\right\rfloor$$

### 3.5.2.3.3 Step 3

Qj values are determined for all alternatives.

- Qj = relative closeness of a alternative j to the ideal solution

$$Q_j = v \, . \, (S_j - S^*) \, / \, (S^- - S^*) + (1-v) \, . \, (R_j - R^*) \, / \, (R^- - R^*)$$
$$S^*=min_j \, S_j$$
$$S^-=max_j \, S_j$$
$$R^*=min_j \, R_j$$
$$R^-=max_j \, R_j$$

**v** is the weight of strategy: “the majority of attributes” .

- v = weight of the strategy that ensures maximum group benefit
- (1-v) =weight of the strategy that ensures minimum individual regret

The Compromise Solution can be obtained by:

- "voting by majority" rule (when $v>0.5$)
- "consensus" (when $v=0.5$)
- "veto" (when $v<0.5$)

#### 3.5.2.3.4 Step 4

Alternatives are ranked according to Sj, Rj and Qj values, in ascending order, in three lists. The list, which is ranked according to Qj value, gives the compromised ranking list for a given v. The alternative with the minimum Qj value is the best alternative.

#### 3.5.2.3.5 Step 5

The compromise solution is proposed for the given attribute weights. The best alternative is $a_k$, if the these conditions are satisfied:

- Condition 1: "Acceptable advantage"
  - $Q(a_k)-Q(a_l)>= 1/(N-1)$
  - $a_l$ is the second best alternative
- Condition 2: "Acceptable stability in decision making"
  - alternative $a_k$ must also be best ranked by $S_j$ and/or $R_j$.

## 3.6 Conclusion

The approach proposed by (Baykasoglu, 2002) has proved its efficiency on the IPPS problems in maintaining the flexibility in the system during scheduling and obtaining better performance. It is easier to represent the generic process plans and making modifications on them during scheduling phase, when this method is used. This is why this method is chosen in this thesis as the main method in process planning phase.

The importance of Giffler & Thompson Algorithm relies on the fact that it generates active schedules for job shop scheduling problems. The schedules of a particular subset of all possible schedules, which are generated without delaying some other operation are called active schedules, thus every optimal schedule is an active optimal schedule. (Giffler and Thompson, 1960), (Octavia, 2003).

The CLONALG algorithm is a fast and efficient algorithm. It has powerful characteristics and improved features over other heuristic algorithms. The cloning and maturation phases of the algorithm are very remarkable in terms of generating new candidates proportional and reversely proportional to their affinity values.

VIKOR method is powerful method used for MCDM problems. In this study, every solution that has been scheduled is a candidate to be chosen and each of these solution strings has three criteria that are going to represent their feasibility, *total flow time, total cost* and *total tardiness & earliness*.

AIS algorithm decides to eliminate or choose from all of the candidates according to their feasibility. The feasibility of a solution string will be calculated and all candidates will be sorted by applying VIKOR Method.

- In the proposed method the weights of each criteria is equal and “consensus” (when v=0.5) is choosen.

# CHAPTER 4

# A SOFT COMPUTING BASED APPROACH TO INTEGRATED PROCESS PLANNING AND SCHEDULING WITH SETUP AND MACHINE CAPACITY CONSIDERATIONS

## 4.1 Introduction

The present work investigates a new approach, which uses the grammatical representation of generic process plans proposed by (Baykasoglu and Ozbakir, 2009) and clonal selection algorithm for the integration of process planning and scheduling. In this approach, Giffler & Thompson Algorithm and VIKOR method is employed, considering setup times and machine capacity constraints. Proposed approach is applied to some literature problems in order to analyze its performance.

## 4.2 The Framework

The framework of the proposed study can be shown in Figure12.

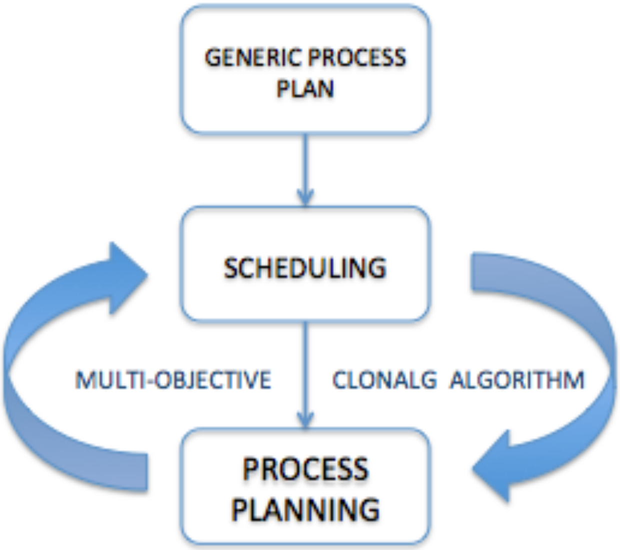


**Figure 12** The framework of the proposed study, adopted from (Baykasoglu and Ozbakır, 2009)

### 4.2.1 Solution Representation

Each solution in the proposed model is represented as a string, which comprises two parts (Table4). The first part includes set of control parameters of the GPP grammar. The second part presents set of dispatching rules.

**Table 4** The string representation of (Baykasoglu, 2002)

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\pi_{11}$ | $\mu_{11}{}^{1}$ | … | $\mu_{n1}{}^{1}$ | | $\pi_{f1}$ | $\mu_{11}{}^{y}$ | … | $\mu_{n1}{}^{y}$ | | 1 | … | 1 |
| $\pi_{12}$ | $\mu_{12}{}^{1}$ | … | $\mu_{n2}{}^{1}$ | | $\pi_{f2}$ | $\mu_{12}{}^{y}$ | … | $\mu_{n1}{}^{y}$ | | 2 | … | 2 |
| … | … | … | | | … | … | … | … | | … | … | … |
| … | … | … | | | … | … | … | … | | … | … | … |
| $\pi_{1q}$ | $M_{1m}{}^{k}$ | … | $\mu_{ni}{}^{k}$ | | $\pi_{fz}$ | $\mu_{1m}{}^{k}$ | … | $\mu_{ni}{}^{k}$ | | z | … | z |

PP1 Ox … Oy … PPf Ox … Oy … p1 … pz

< ----------- PART1 --------- > … < ------- PART n ------- > < dispatching rules >

< --Control parameters of the grammar Generic Process Plans-- >

PPi…PPf : operation sequence alternatives set

Ox…Oy : operation locations

p1…pt : dispatching rule sets

Consider the following string for the previous example mentioned in ‘Representation of Generic Process Plans’:

**1** 1 2 2 **1** 2 1 1 2 1 3 2 1 1

The first entry “1” in the string represents that the operation sequence 1 ($\pi_{11}$) is chosen randomly for P1 among alternative operation sequences. The second entry means that M1 ($\mu^{1}{}_{11}$) is randomly selected for the first operation mentioned in the selected operation sequence of P1 among the alternative machines etc. The entries, which are underlined represent the selected dispatching rules, for example the 10th entry shows that the dispatching rule 1 is randomly selected for the second position of dispatching rule set etc.

In the solution string, the bolded entries represent the operation sequence selection controls ($\pi_{ij}$), between these bolded entries, the machine selection controls ($\mu^k_{ij}$) are given and the underlined entries are for dispatching rule set for the operations.

### 4.2.2 Process Planning & Scheduling Phases

This model has two integrated parts, which act simultaneously. The first part generates final process plans using the GPP grammar. The second part generates feasible schedules from those process plans with the Giffler and Thompson's dispatching rule based heuristic.

Giffler and Thompson (1960)'s algorithm is used in the scheduling part of the proposed study. Using the predefined dispatching rules, active schedules are obtained. There exist many different sets of controls in the GPP grammar and different selections of them will produce different problems. As the set of controls are chosen, the corresponding schedule should be determined and the performance evaluation should be done. The main issue is to choose the optimal combination of these controls, which results in the best performance evaluation.

### 4.2.3 Optimization Phase - CLONALG

Artificial Immune System Algorithm - CLONALG is chosen as the optimization algorithm, which is inspired by the complex and effective vertebrate immune system. CLONALG searches the solution space for the optimal solution, as it generates alternative solutions by selecting different set of controls from the grammar and selecting different dispatching rules used in Giffler and Thompson's algorithm.

#### 4.2.3.1 Maturation

In this work, maturated solutions are generated with the same method in (Baykasoglu and Ozbakir, 2009)'s study:

> **n** candidate parts are selected. The operation sequences of each part are reselected according to the predetermined probabilities. Then, the machines of **m** number of operations are reselected and **k** numbers of dispatching rules are changed from dispatching rules set.

A control mechanism is developed for detecting the infeasible solutions, which may occur during this process. The operation sequences for **n** parts are reselected. As the new machines are being selected for **m** number of operations of **n** number of parts, the reselected machines can be incompatible with the newly selected operation sequence. At this point, the control mechanism checks each generated neighborhoods. Then, it corrects infeasible ones with the randomly chosen compatible machines. As a result this movement strategy generates always-feasible solutions and it is easy to implement. Each solution is sent to scheduling part to be scheduled via Giffler and Thompson's heuristic and its performance is obtained via VIKOR method.

**Table 5** Maturation of the solution string

| **Before the Maturation** | **After the Maturation** |
|---|---|
| Solution string:<br>**1** 1 2 2 **1** 2 1 1 1 2 3 2 1 1 | Solution string:<br>**2** 2 2 2 **2** 1 2 1 3 1 3 2 1 1 |

An example of maturated candidate for the previous example mentioned in 'Representation of Generic Process Plans' is shown in Table5.

In this example, n=2 which means, operation sequence of the first and second parts will be reselected. m =2, so new machines will be selected for the first two operations in the new operation sequence of the selected parts and first two dispatching rules will be reselected because k=2.

### 4.2.3.2 Affinity Values and VIKOR Method

Performance criterion of each solution is based on three objective functions in the proposed system. These are "total cost of process plans, total flowtime and total tardiness/earliness.

Total flow time = $\sum_{i=1}^{n} Ci$

where, Ci is completion time of part i .

The total of fixed and variable costs gives the total cost of process plans. In this model, the operation's processing time and the part's scheduling priority are considered while determining the variable cost $VC_{jk}$. The fixed cost is independent of these considerations (Wong et al., 2006). The variable cost is determined using the currency conversion function of (Krothapalli and Deshmukh, 1999).

The variable cost function of processing an operation j on machine k is calculated as:

$VC_{jk}=2\tan^{-1}.(e^{t_{jk}A})$

$t_{jk}$ = The processing time for performing operation j on machine k.

A = Shape factor which determines the curvature of currency function (Wong et al., 2006).

The shape factor A is a function of the part's critical ratio cr, as shown in Eq. below.

$A=0.11-((1+\tanh(cr-1))/20)$

cr = is the critical ratio, (the ratio between the due date, d, and the processing time of the part, T . cr = d/T . )

The total cost of a process plan is given by Eq. below:

Total cost of a process plan=$\sum_j \sum_k VC_{jk}$

The total earliness/tardiness is calculated using the below formula:

The total earliness/tardiness = $\sum_{i=1}^{n}|Ci-di|$

Ci = completion time of part i,

di = due date of part i,

The affinity values of each candidate will be calculated by sending each of these performance criteria as an input to VIKOR Method. Each candidate will represent an alternative and each performance value will represent the criteria in VIKOR Method structure. The calculated Qj values will represent the affinity of each candidate. In order to find the best candidate from the population, each alternative (candidate) will be sorted according to Qj values (affinity values) in ascending order. The alternative with minimum Qj value will be the best candidate.

### 4.3 Setup Times

In the scheduling process via Giffler & Thompson algorithm, a setup must be done to the related machine before assigning the operation-part couple (for ex. O1P1). This is a time duration needed for making the required preparation (such as installation of required elements, cleaning the machine…etc.) to make the machine ready to process the upcoming operation-part couple.

The issues should be considered during adding these setup times to the machines are:

- What is the last operation-part couple processed on that machine?
- Which operation-part duple will be processed next on that machine?
- After assigning the part-operation duple to the compatible machine and adding the setup time, the time limit for a day should not be exceeded. (for ex. consider the production time is 12 hours/day )

**Minor Setups:** If operations of the same part are processed consecutively on the same machine, the setup time will be minor. (For ex. O1P1-> O2P1 on machine1)

**Major Setups:** If a part being processed is different from the previous part processed on the same machine, then a major setup is needed. (For ex. O2P2->O2P1 on machine1)

For each machine, a table is needed that includes the information about setup times between each operation-part duple. The table below is an example setup-time table of machine1, for the previous example mentioned in 'Representation of Generic Process Plans':

**Table 6** Setup-time table of machine1

| Machine1 | O1P1 | O2P1 | O3P1 | O1P2 | O2P2 | O3P2 |
|---|---|---|---|---|---|---|
| O1P1 | - | 2 | 3 | 2 | 1 | 1 |
| O2P1 | 2 | - | 3 | 1 | 2 | 1 |
| O3P1 | 2 | 1 | - | 2 | 1 | 2 |
| O1P2 | 2 | 1 | 2 | - | 1 | 1 |
| O2P2 | 2 | 4 | 3 | 3 | - | 1 |
| O3P2 | 2 | 1 | 1 | 2 | 2 | - |

### 4.4 Machine Capacity Constraints

In the proposed model, it is considered that each machine has limited time for working per day; also the production time per day is limited too. It means that when a machine exceeds the time it can work daily, no more operation can be processed on that machine. This machine capacity constraint should be considered during the scheduling part in Giffler & Thompson algorithm, while the operation-part couple is being assigned to a machine.

The issues should be considered during the assignment of a part-operation couple to a machine are:

- Do the remaining working time of the machine is enough for the processing time of the part-operation couple?
- After assigning the part-operation couple to the compatible machine, the time limit for a day should not be exceeded. (for ex. consider the production time is 12 hours/day )

If the remaining working time of the machine is enough for processing the part-operation couple and the day limit is not exceeded, then the assignment can be done. If the part operation couple can't be assigned to the machine due to capacity constraints, then the alternative machines, which have enough time to process that part-operation couple, are considered. If there exists another available machine, which that part-operation couple can be processed, then assignment is done but as a result of this change in the machines, the solution string should be updated. If there are no machine left for that machine then the next part-operation couple in the conflicting set $C_t$ , is tried to be assign to the machines which are available. If no possible action is left, then the production stops for that day and on the next day it starts again with full capacity in each machine.

## 4.5 Conclusion

In this study, a new approach which makes use of Baykasoglu's grammatical representation of generic process planning and CLONALG is proposed to solve integrated process planning and scheduling problems considering setup times and machine capacity constraints. Total flow time, total processing cost and total tardiness/earliness are considered as performance criteria in searching for optimal solutions. Algorithm performance is tested on case studies and it is possible to obtain better performance and to realize the flexibility available in the system during scheduling by making use of the proposed approach in environments where there exists setup times and machine capacity constraints.

# CHAPTER 5

# EXAMPLE APPLICATIONS, RESULTS AND SOFTWARE DEVELOPED

## 5.1 Introduction

C# computer programming language is used to encode the proposed model. MSQL Database is used store the part, operation and machine data as well as the calculations, process plans, production schedules, candidate and clone sets during the application of the proposed model. In this chapter, the execution of the program is represented with case studies.

## 5.2 Case 1

The same problem of (Baykasoglu and Ozbakir, 2009) is tested with the developed program. There are 2 parts, 3 machines, 3 operations and 9 different dispatching rules to be used in the solution string as shown in Table 8. The data sets are shown in Tables (1,2,3,6,7,8,9,10).

**Table 7** Machine capacities

| **Machine** | **Capacity/day** |
|---|---|
| $M_1$ | 10 hours/day |
| $M_2$ | 8 hours/day |
| $M_3$ | 6 hours/day |

**Table 8** Part due dates for case 1

| **Part** | **Due Date** |
|---|---|
| $P_1$ | 15 hours |
| $P_2$ | 10 hours |

One-day working time is considered as 12 hours in the example, and the due dates for the parts are determined as 15 and 10 hours respectively. The capacities for the machining times are determined as 10, 8 and 6 hours/day. For ex. M1 can be processed for 10 hours maximum per day. While assigning an operation to a machine remaining machining time of that machine is considered.

**Table 9** Dispatching rules

| **DISPATCHING RULES EXPLANATION** | |
|---|---|
| 1) SPT | Shortest process time |
| 2) EDD | Earliest due date |
| 3) LPT | Longest process time |
| 4) MWRT | Most work remaining time |
| 5) LWRT | Least work remaining time |
| 6) PDR | Process time/Remaining time |
| 7) ERD | Earliest release date |
| 8) MS | Minimum Slack |
| 9) LNS | Least number of remaining process |

Since there are three machines in this example; three setup-time tables are needed. Setup-time table of M1 is given in Table 6.

**Table 10** Setup-time table of machine2

| $M_2$ | O1P1 | O2P1 | O3P1 | O1P2 | O2P2 | O3P2 |
|---|---|---|---|---|---|---|
| O1P1 | - | 1 | 2 | 2 | 1 | 1 |
| O2P1 | 2 | - | 1 | 3 | 2 | 2 |
| O3P1 | 2 | 1 | - | 2 | 3 | 2 |
| O1P2 | 2 | 2 | 2 | - | 1 | 1 |
| O2P2 | 2 | 3 | 3 | 1 | - | 1 |
| O3P2 | 2 | 1 | 1 | 2 | 2 | - |

**Table 11** Setup-time table of machine3

| $M_3$ | O1P1 | O2P1 | O3P1 | O1P2 | O2P2 | O3P2 |
|---|---|---|---|---|---|---|
| O1P1 | - | 2 | 2 | 2 | 1 | 1 |
| O2P1 | 1 | - | 1 | 3 | 2 | 1 |
| O3P1 | 2 | 1 | - | 2 | 3 | 2 |
| O1P2 | 2 | 1 | 2 | - | 1 | 1 |
| O2P2 | 3 | 4 | 3 | 2 | | 1 |
| O3P2 | 2 | 2 | 2 | 2 | 2 | - |

**Table 12** AIS CLONALG Parameter Settings

| | |
|---|---|
| Number of iterations | 20 |
| Population Size | 20 |
| Number of candidates chosen for cloning in each iteration | 5 |
| Number of clones to be maturated in each iteration | 16 |
| Number of selected clones for adding to generation in each iteration | 5 |
| Number of killed candidates from generation in each iteration | 10 |
| Number of newly generated candidates in each iteration | 5 |
| *k , m , n* parameters for Maturation | 2, 2, 3<br>1, 2, 3<br>1, 2, 2<br>1, 1, 2<br>1, 1, 1 |

The number of iterations and initial population size is selected small sized because the case problem is also a small size problem. When the initial population is generated using Grammatical Representation Method, then each solution string (candidate) is sent to Giffler & Thompson Scheduling phase. During the scheduling phase the machine capacities and setup times are considered, and the possible changes are reflected to the process-planning phase to be updated in the solution string. Then each scheduled candidate is sent to affinity calculation module, the affinities of each candidate are calculated by applying VIKOR Method considering total cost, total flow time and total tardiness/earliness as performance criteria.

Given a solution string as **1** 1 3 2 **2** 2 2 1 5 5 1 3 1 4 selected randomly from the population, the schedule is obtained from the scheduling module and each performance criterion is calculated in affinity calculation module. The total flow time of the candidate is 11 hours. The completion time for P1 is 11 and P2 is 10 hours. Therefore the total earliness/tardiness is 4 hours with an exact time for P2 but 4 hours earliness for P1. Total cost is calculated by the proposed variable cost function and found as 26,47819 unit.

When all candidates' performance criteria are calculated, Sj and Rj values are calculated for each of them. Finally when Qj values (affinity values) are calculated using Sj and Rj values, the candidates are ranked in an ascending order according to the Qj values.

Using the parameters in Table 12, 5 candidates with least Qj values are selected for cloning module. Each candidate is cloned proportional to its affinity value. For example the best candidate from the five selected candidates is cloned 5 times, the second best candidate is cloned 4 times and with a decreasing number in cloning like this all the clones are generated. For this case problem, 16 clones are generated from five selected candidates per iteration. Then these clones are sent to maturation module to be maturated with an inverse ratio to their affinity values. That means, the better the candidate, the maturation level is low; worse the candidate, the maturation level is high. In this case problem there are five maturation levels, which means the clones originated from each selected candidate will be belong to different maturation levels. For example, the best candidate of five selected candidates is cloned 5 times; these 5 clones will be belonging to maturation level five with minimum maturation parameters. After the maturation is completed, the maturated clones are sent to scheduling phase to be scheduled, in the scheduling module each clone's possibility of realization is checked. For example during the maturation of a candidate, the operation sequence of the first part is randomly changed and the machine of an operation belonging to that operation sequence is changed randomly. In such a case, if the random selected machine is not in the set of alternative machines of that operation belonging to that operation sequence, then it means, that process plan of the candidate cannot be realized. The control module in scheduling phase controls each maturated clone and corrects if there is an unsuitable assignment made in the maturation phase. When all the maturated clones are scheduled, their affinity values are calculated in affinity calculation module. The best 5 clones are selected to the population. The worst 10 candidate of the population is killed and 5 randomly generated new candidates are added to the population. As a result, the first iteration of CLONALG is completed, with the same population size.

**Table 13** Results

| | |
|---|---|
| Total Cost | 23.3244 |
| Total Tardiness/Earliness | 5 |
| Total Flowtime | 10 |

After all the iterations are completed, the solution string found as the best candidate is **1** 2 3 2 **1** 1 2 3 3 4 3 4 1 2 , with a total cost of 23.3244, flowtime of 10 hours and total tardiness/earliness of 5 hours. The string represents that the operation sequence 1 is chosen for P1 and P2. M2, M3 and M2 are selected for the corresponding operations in operation sequence 1 for part1. M1, M2 and M3 are selected for the corresponding operations in operation sequence 1 for P2. Dispatching rules 3,4,3,4,1,2 are selected for the dispatching rules set.

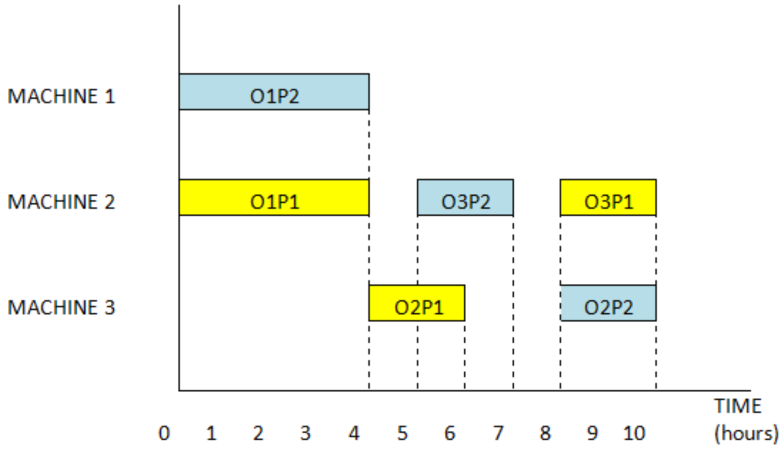


**Figure 13** Final Schedule of the best candidate on timetable for case 1

The capacity of M1 is 10 hours and after processing O1P2 (operation1 of part2), the capacity remained on machine1 is 6 hours. The capacity of M2 is 8 hours and after processing O1P1, the remaining capacity becomes 4. A setup process must be done on M2 to be able to process O3P2 after O1P1and this setup process lasts for 1 hour. Then O3P2 is processed and, the remaining capacity for M2 becomes 2. Another setup process must be done on M2 to be able to process O3P1 after O3P2and this setup process also lasts for 1. Then when O3P1 is processed, the remaining capacity for M2 becomes 0. The capacity of machine3 is 6 hours and after processing O2P1, the remaining capacity becomes 4. A setup process must be done on M3 to be able to process O2P2 after O2P1and this setup process lasts for 2 hours. Then O2P2 is processed and, the remaining capacity for M2 becomes 2.

### 5.3 Case 2

In the second case problem, there are 4 parts, 4 machines, 4 operations and 9 different dispatching rules to be used in the solution string as shown in Table 8. The data sets are shown in Tables 14, 15, 16 and 17.

**Table 14** Part due dates for case 2

| **Part** | **Due Date** |
|---|---:|
| P1 | 25 hours |
| P2 | 24 hours |
| P3 | 27 hours |
| P4 | 33 hours |

**Table 15** Machines data for case problem 2

| **Machine** | **Capacity/day** |
|---|---:|
| M1 | 6 hours/day |
| M2 | 8 hours/day |
| M3 | 8 hours/day |
| M4 | 10 hours/day |

**Table 16** Part operation alternatives

| PartID | Alternative_Seq | O1 | O2 | O3 | O4 |
|---:|---:|---:|---:|---:|---:|
| 1 | 1 | 1 | 2 | 3 | 4 |
| 1 | 2 | 2 | 1 | 4 | 3 |
| 1 | 3 | 4 | 3 | 2 | 1 |
| 2 | 1 | 1 | 2 | 4 | 3 |
| 2 | 2 | 2 | 1 | 3 | 4 |
| 2 | 3 | 4 | 2 | 1 | 3 |
| 2 | 4 | 4 | 3 | 1 | 2 |
| 3 | 1 | 1 | 2 | 4 | 3 |
| 3 | 2 | 2 | 3 | 1 | 4 |
| 4 | 1 | 2 | 3 | 4 | 1 |
| 4 | 2 | 1 | 4 | 2 | 3 |
| 4 | 3 | 1 | 3 | 2 | 4 |

**Table 17** Processing times for operation of parts on machines

| OpID | PartID | MachID | Processing Time |
|---:|---:|---:|---:|
| 1 | 1 | 1 | 2 |
| 1 | 1 | 2 | 3 |
| 1 | 1 | 3 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 3 |
| 1 | 3 | 1 | 3 |
| 1 | 3 | 4 | 2 |
| 1 | 4 | 2 | 2 |
| 1 | 4 | 4 | 4 |
| 2 | 1 | 2 | 1 |
| 2 | 1 | 3 | 2 |
| 2 | 1 | 4 | 4 |
| 2 | 2 | 1 | 2 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 3 |
| 2 | 3 | 1 | 2 |
| 2 | 3 | 3 | 2 |
| 2 | 3 | 4 | 3 |
| 2 | 4 | 1 | 2 |
| 2 | 4 | 2 | 4 |
| 3 | 1 | 3 | 2 |
| 3 | 1 | 4 | 3 |
| 3 | 2 | 1 | 2 |
| 3 | 2 | 2 | 1 |
| 3 | 3 | 1 | 2 |
| 3 | 3 | 2 | 3 |
| 3 | 3 | 3 | 1 |
| 3 | 4 | 1 | 2 |
| 3 | 4 | 2 | 1 |
| 3 | 4 | 4 | 3 |
| 4 | 1 | 2 | 1 |
| 4 | 1 | 3 | 2 |
| 4 | 2 | 3 | 1 |
| 4 | 2 | 4 | 2 |
| 4 | 3 | 3 | 1 |
| 4 | 4 | 2 | 2 |
| 4 | 4 | 3 | 3 |

**Table 18** AIS CLONALG Parameter Settings for Case 2

| | |
|---|---|
| Number of iterations | 50 |
| Population Size | 50 |
| Number of candidates chosen for cloning in each iteration | 10 |
| Number of clones to be maturated in each iteration | 56 |
| Number of selected clones for adding to generation in each iteration | 10 |
| Number of killed candidates from generation in each iteration | 20 |
| Number of newly generated candidates in each iteration | 10 |
| $k$ , $m$ , $n$ parameters for Maturation | 4, 3, 6<br>3, 2, 5<br>2, 2, 4<br>2, 1, 4<br>1, 1, 2 |

**Table 19** Results for Case 2

| | |
|---|---|
| Total Cost | 75.3228 |
| Total Tardiness/Earliness | 24 |
| Total Flowtime | 26 |

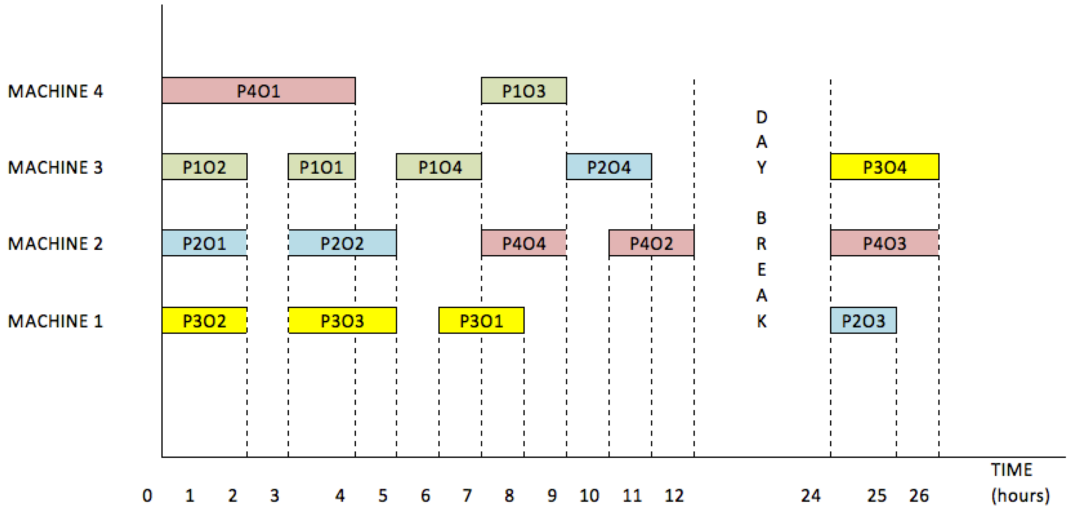


**Figure 14** Final Schedule of the best candidate on timetable for case problem 2

After all the iterations are completed, the solution string found as the best candidate is : **2** 3 3 3 4 **1** 2 2 3 1 **2** 1 1 1 3 **2** 4 2 2 2 1 3 4 2 3 1 4 2 1 3 3 4 1 4 1 2 . As it can be seen from the Figure 14, when M1 completed processing P3O1, it has worked for 6 hours. That means, the left capacity of M1 is 0, thus P2O3 cannot be assigned to M1. The other machine, which P2O3 can be processed on, is M2. M2 is not available for P2O3, because P4O3 has already been assigned to M2. As a result P2O3 is left for the following day due to capacity constraints. P3O4 and P4O3 are also left for the following day because when they cannot be assigned to any machines without violating the day constraint, which is 12 hours working time per day. So after 12 hours break time, the new working day starts and the processing of the remained part-operation couples continue.

### 5.3 Case 3

In the third case problem, there are 5 parts, 5 machines, 5 operations and 9 different dispatching rules to be used in the solution string as shown in Table 8. The data sets are shown in Tables 20, 21, 22 and 23.

**Table 20** Part due dates for case 3

| Part | Due Date |
|---|---:|
| P1 | 25 hours |
| P2 | 24 hours |
| P3 | 29 hours |
| P4 | 34 hours |
| P5 | 22 hours |

**Table 21** Machines data for case problem 3

| Machine | Capacity/day |
|---|---:|
| M1 | 8 hours/day |
| M2 | 8 hours/day |
| M3 | 8 hours/day |
| M4 | 8 hours/day |
| M5 | 8 hours/day |

There are 5 parts in this case problem. P1, P2, P3 and P4 of them have 5 operations to be processed and P5 have 4 operations to be processed. The different operation sequences of each part is shown in Table 22 and the processing times of each operation of each part in the available machines are given in Table 23.

**Table 22** Part operation alternatives for case problem 3

| PartID | Alternative_Seq | Op1 | Op2 | Op3 | Op4 | Op5 |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 2 | 3 | 4 | 5 |
| 1 | 2 | 2 | 1 | 4 | 5 | 3 |
| 2 | 1 | 1 | 3 | 5 | 4 | 2 |
| 2 | 2 | 5 | 4 | 3 | 2 | 1 |
| 2 | 3 | 3 | 2 | 1 | 4 | 5 |
| 3 | 1 | 4 | 3 | 2 | 1 | 5 |
| 3 | 2 | 1 | 2 | 5 | 4 | 3 |
| 4 | 1 | 3 | 2 | 5 | 1 | 4 |
| 4 | 2 | 5 | 3 | 1 | 2 | 4 |
| 5 | 1 | 1 | 4 | 3 | 2 |  |
| 5 | 2 | 4 | 2 | 1 | 3 |  |

**Table 23** AIS CLONALG Parameter Settings for Case 3

| | |
|---|---|
| Number of iterations | 100 |
| Population Size | 50 |
| Number of candidates chosen for cloning in each iteration | 10 |
| Number of clones to be maturated in each iteration | 56 |
| Number of selected clones for adding to generation in each iteration | 10 |
| Number of killed candidates from generation in each iteration | 20 |
| Number of newly generated candidates in each iteration | 10 |
| *k* , *m* , *n* parameters for Maturation according to affinities of clones | 5, 4, 10<br>4, 3, 8<br>3, 2, 6<br>2, 1, 4<br>1, 1, 2 |

**Table 24** Processing times for operations of parts on machines

| OpID | PartID | MachID | Processing Times |
|---:|---:|---:|---:|
| 1 | 1 | 1 | 2 |
| 1 | 1 | 2 | 3 |
| 1 | 1 | 3 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 3 |
| 1 | 2 | 5 | 2 |
| 1 | 3 | 1 | 3 |
| 1 | 3 | 4 | 2 |
| 1 | 3 | 5 | 3 |
| 1 | 4 | 2 | 2 |
| 1 | 4 | 4 | 4 |
| 1 | 5 | 2 | 2 |
| 1 | 5 | 4 | 1 |
| 2 | 1 | 2 | 1 |
| 2 | 1 | 3 | 2 |
| 2 | 1 | 4 | 4 |
| 2 | 1 | 5 | 2 |
| 2 | 2 | 1 | 2 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 3 |
| 2 | 3 | 1 | 2 |
| 2 | 3 | 3 | 2 |
| 2 | 3 | 4 | 3 |
| 2 | 4 | 1 | 2 |
| 2 | 4 | 2 | 4 |
| 2 | 4 | 5 | 3 |
| 2 | 5 | 1 | 1 |
| 2 | 5 | 3 | 2 |
| 2 | 5 | 5 | 2 |
| 3 | 1 | 3 | 2 |
| 3 | 1 | 4 | 3 |
| 3 | 2 | 1 | 2 |
| 3 | 2 | 2 | 1 |
| 3 | 2 | 5 | 3 |
| 3 | 3 | 1 | 2 |
| 3 | 3 | 2 | 3 |
| 3 | 3 | 3 | 1 |
| 3 | 4 | 1 | 2 |
| 3 | 4 | 2 | 1 |
| 3 | 4 | 4 | 3 |
| 3 | 5 | 1 | 3 |
| 3 | 5 | 5 | 2 |
| 4 | 1 | 2 | 1 |
| 4 | 1 | 3 | 2 |
| 4 | 1 | 5 | 3 |
| 4 | 2 | 3 | 1 |
| 4 | 2 | 4 | 2 |
| 4 | 2 | 5 | 1 |
| 4 | 3 | 1 | 2 |
| 4 | 3 | 3 | 1 |
| 4 | 4 | 2 | 2 |
| 4 | 4 | 3 | 3 |
| 4 | 5 | 1 | 1 |
| 4 | 5 | 3 | 2 |
| 4 | 5 | 5 | 3 |
| 5 | 1 | 1 | 2 |
| 5 | 1 | 5 | 3 |
| 5 | 2 | 3 | 2 |
| 5 | 2 | 4 | 1 |
| 5 | 3 | 1 | 2 |
| 5 | 4 | 3 | 3 |

**Table 25** Results for Case 3

| | |
|---|---|
| Total Cost | 112.348 |
| Total Tardiness/Earliness | 27 |
| Total Flowtime | 30 |

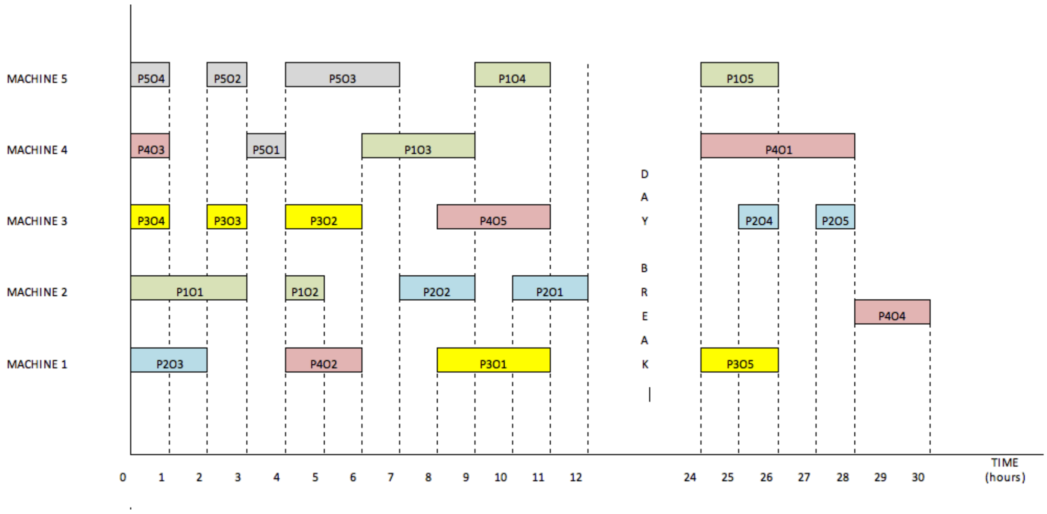


**Figure 15** Final Schedule of the best candidate on timetable for case problem 3

After all the iterations are completed, the solution string found as the best candidate is : **1** 2 2 4 5 5 **3** 1 2 2 3 3 **1** 3 3 3 1 1 **1** 4 1 3 4 **2** 5 5 4 5 3 5 2 1 1 5 6 3 7 2 9 8 8 4 3 2 3 1 1 3 4 5 4

## 5.4 Benchmark Problems

The methodology is tested on benchmark problems of (Kacem et al., 2002a,b) in order to compare the performance. The setup times are considered as zero and the machine capacities are considered as limitless to be able to make comparison on an equal basis. The results are shown in Table 26.

**Table 26** The results of the benchmark problems

| | **Completion Time** | **Maximum Machine Load** | **Total Machine Load** |
|---|---|---|---|
| **Problem 1** | | | |
| Lower Bounds | 16 | 7 | 32 |
| Kacem et al. (2002 a,b) | 18 | 8 | 32 |
| | 18 | 7 | 33 |
| | 16 | 9 | 35 |
| | 16 | 10 | 34 |
| Baykasoglu & Ozbakır (2009) | 16 | 8 | 32 |
| | 16 | 7 | 33 |
| Proposed Study | 16 | 8 | 33 |
| **Problem 2** | | | |
| Lower Bounds | 15 | 9 | 60 |
| Kacem et al. (2002 a,b) | 15 | 11 | 61 |
| | 17 | 10 | 64 |
| | 18 | 10 | 63 |
| | 16 | 10 | 66 |
| | 16 | 12 | 60 |
| Baykasoglu & Ozbakır (2009) | 15 | 10 | 62 |
| | 15 | 11 | 61 |
| | 16 | 12 | 60 |
| Proposed Study | 15 | 11 | 61 |
| **Problem 3** | | | |
| Lower Bounds | 23 | 10 | 91 |
| Kacem et al. (2002 a,b) | 23 | 11 | 95 |
| | 24 | 11 | 91 |
| Baykasoglu & Ozbakır (2009) | 23 | 12 | 92 |
| | 25 | 10 | 95 |
| | 23 | 13 | 91 |
| | 23 | 11 | 93 |
| Proposed Study | 24 | 10 | 92 |
| **Problem 4** | | | |
| Lower Bounds | 7 | 5 | 41 |
| Kacem et al. (2002 a,b) | 8 | 7 | 41 |
| | 8 | 5 | 42 |
| | 7 | 5 | 45 |
| Baykasoglu & Ozbakır (2009) | 8 | 5 | 42 |
| | 8 | 7 | 41 |
| | 7 | 7 | 43 |
| | 7 | 5 | 44 |
| Proposed Study | 8 | 5 | 42 |

The results showed that, the methodology with CLONALG algorithm is competitive among the other algorithms and has a good performance on finding the lower bounds of the benchmark problems.

### 5.5 User Interface of The Developed Program

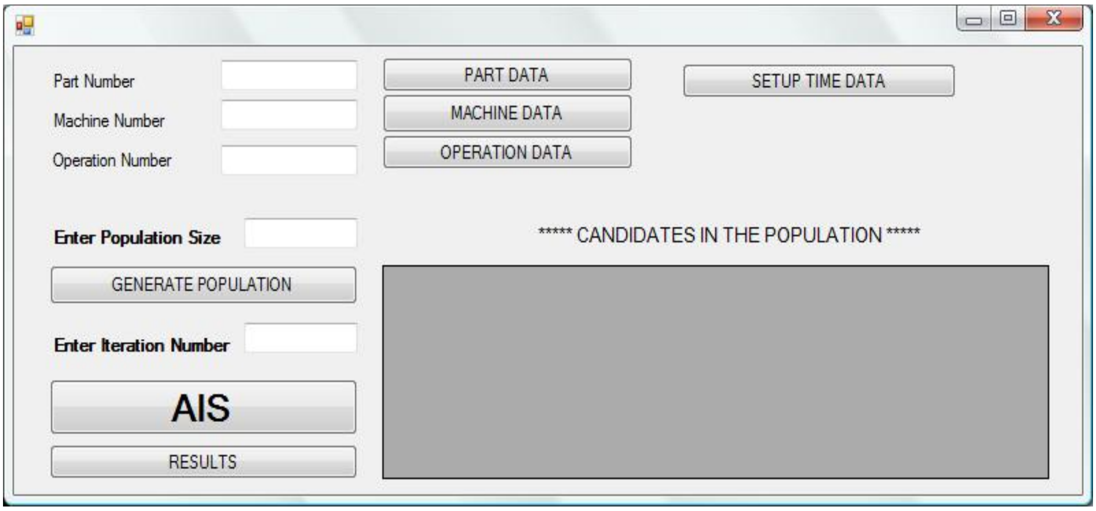
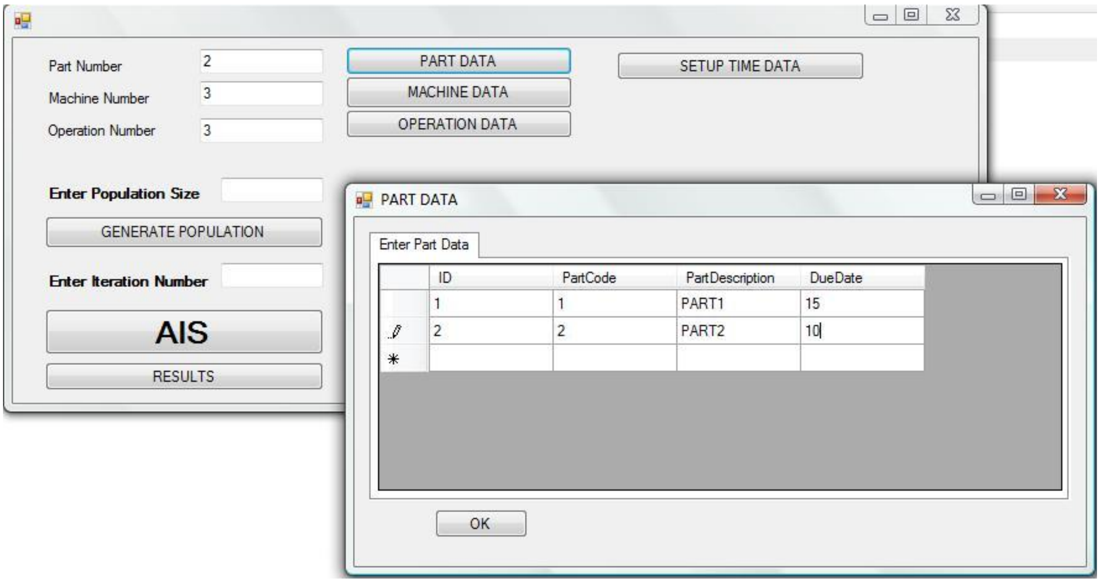


**Figure 16** The User interface of the developed program

A user-friendly interface is developed for the proposed method. The required values are obtained from the user such as number of parts, number of machines, number of operations, operation sequences for each part, due dates, machine capacities and setup times. The developed program works in coordination with MYSQL Database in order to store and retrieve the required information during the execution of the program.

When the user enters the part number in the text box, in order to enter detailed information about the part he/she should click on the PART DATA Button.

In the popped up form, part description and due dates for each part is entered by the user.

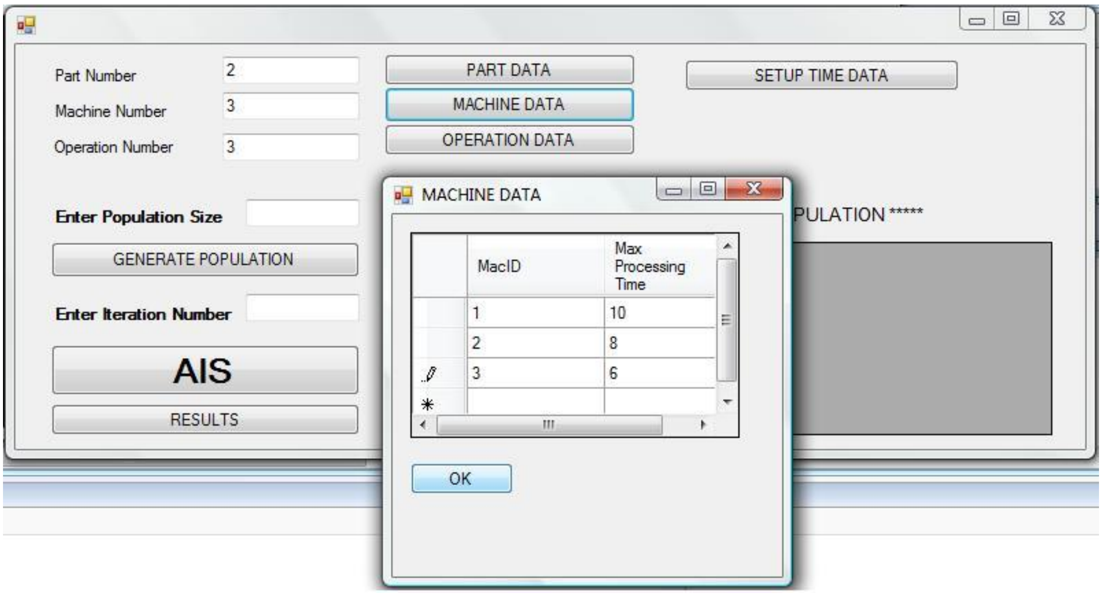


Part Number 2
Machine Number 3
Operation Number 3

| ID | PartCode | PartDescription | DueDate |
|---|---|---|---|
| 1 | 1 | PART1 | 15 |
| 2 | 2 | PART2 | 10 |

**Figure 17** Part Data Form of the developed program.

Then, the number of machines and the working capacities of the machines per day are entered by the user.

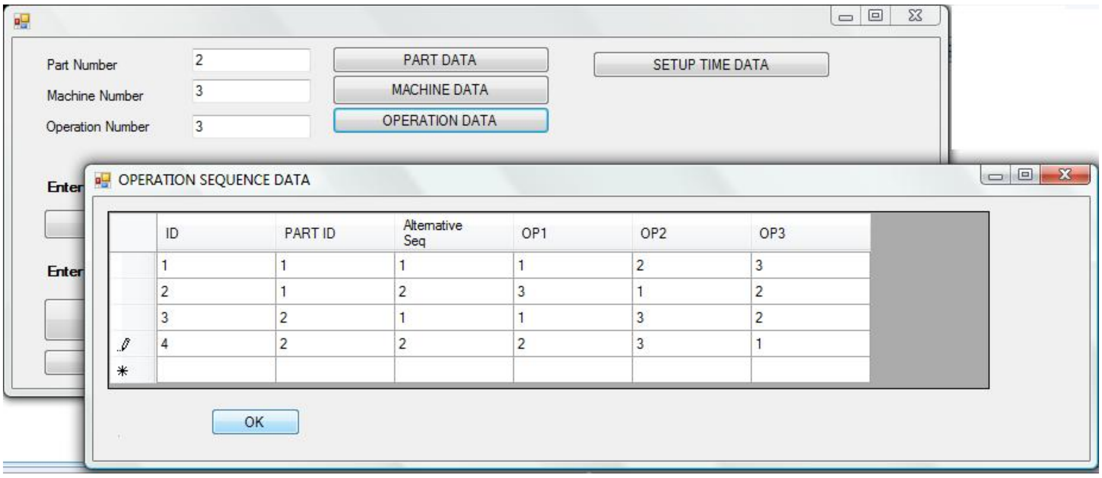


Part Number 2
Machine Number 3
Operation Number 3

| MacID | Max Processing Time |
|---|---|
| 1 | 10 |
| 2 | 8 |
| 3 | 6 |

**Figure 18** Machine Data Form of the developed program.

The operation sequences for each part are entered in the OPERATION DATA Form.

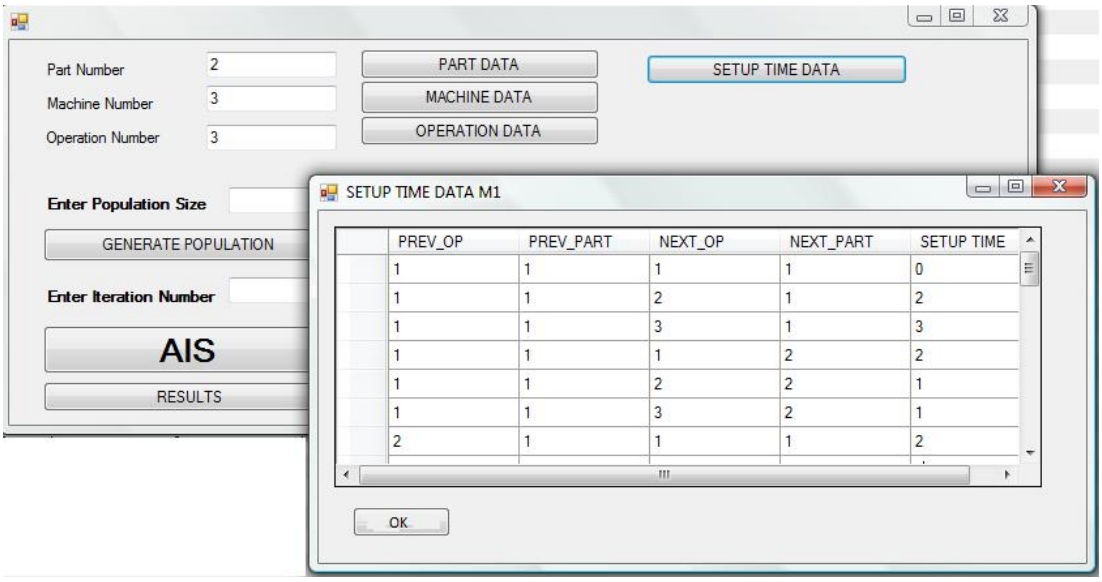


Part Number 2
Machine Number 3
Operation Number 3

| ID | PART ID | Alternative Seq | OP1 | OP2 | OP3 |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 2 | 3 |
| 2 | 1 | 2 | 3 | 1 | 2 |
| 3 | 2 | 1 | 1 | 3 | 2 |
| 4 | 2 | 2 | 2 | 3 | 1 |

**Figure 19** Part Data Form of the developed program.

The setup time for each machine is entered, by clicking on the SETUP TIME DATA on the form.

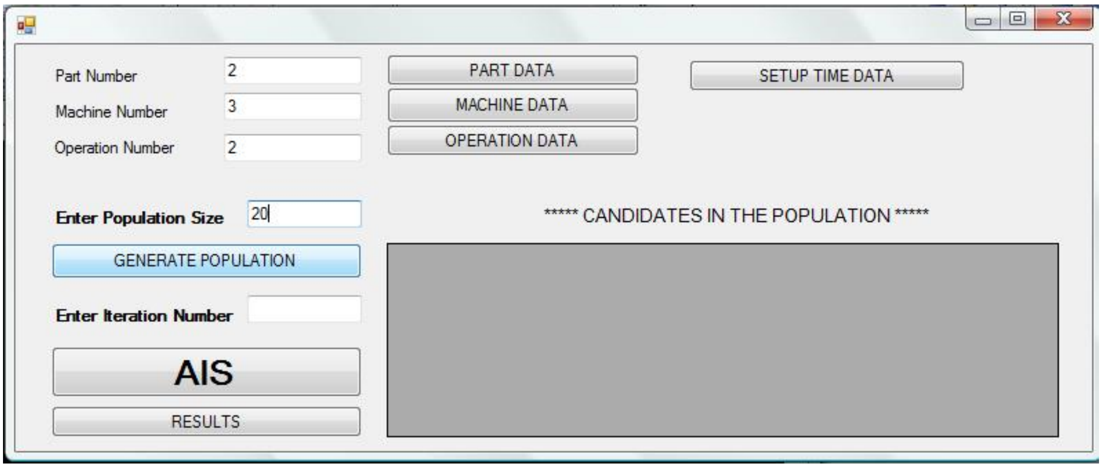


Part Number 2
Machine Number 3
Operation Number 3
Enter Population Size
Enter Iteration Number

| PREV_OP | PREV_PART | NEXT_OP | NEXT_PART | SETUP TIME |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 2 | 1 | 2 |
| 1 | 1 | 3 | 1 | 3 |
| 1 | 1 | 1 | 2 | 2 |
| 1 | 1 | 2 | 2 | 1 |
| 1 | 1 | 3 | 2 | 1 |
| 2 | 1 | 1 | 1 | 2 |

**Figure 20** Setup Time Data Form of the developed program.

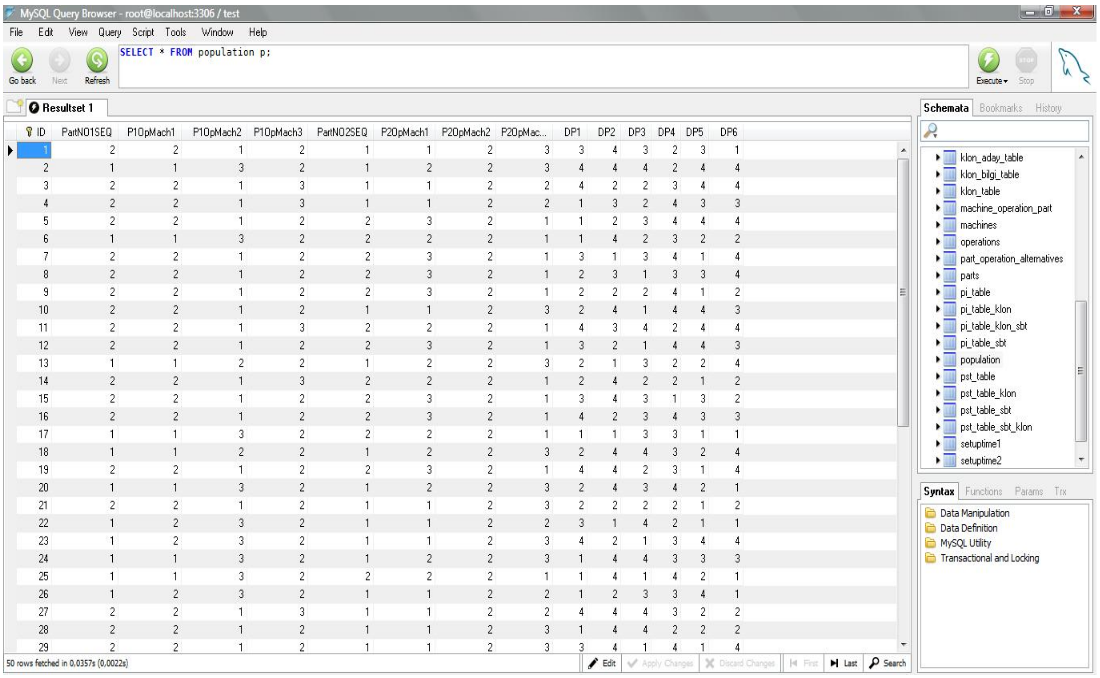
Then the population size is entered. By clicking on the GENERATE POPULATION Button, the program generates candidates, which are random generic process plans.



Part Number: 2
Machine Number: 3
Operation Number: 2
**Enter Population Size**: 20
**Enter Iteration Number**:

**Figure 21** Generating population in the developed program.

The generated population can be viewed in datagram, which is called CANDIDATES IN THE POPULATION on the main form of the program as well as it can be viewed from MYSQL Database in detail.



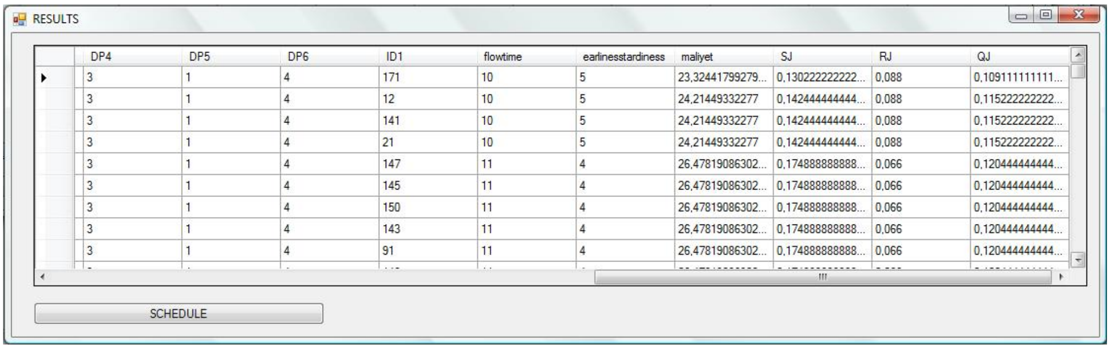
```
SELECT * FROM population p;
```

| ID | PartNO1SEQ | P1OpMach1 | P1OpMach2 | P1OpMach3 | PartNO2SEQ | P2OpMach1 | P2OpMach2 | P2OpMac... | DP1 | DP2 | DP3 | DP4 | DP5 | DP6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 3 | 4 | 3 | 2 | 3 | 1 |
| 2 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 4 | 4 | 4 | 2 | 4 | 4 |
| 3 | 2 | 2 | 1 | 3 | 1 | 1 | 2 | 2 | 4 | 2 | 2 | 3 | 4 | 4 |
| 4 | 2 | 2 | 1 | 3 | 1 | 1 | 2 | 2 | 1 | 3 | 2 | 4 | 3 | 3 |
| 5 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 1 | 2 | 3 | 4 | 4 | 4 |
| 6 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 4 | 2 | 3 | 2 | 2 |
| 7 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 3 | 1 | 3 | 4 | 1 | 4 |
| 8 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 2 | 3 | 1 | 3 | 3 | 4 |
| 9 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 2 | 2 | 2 | 4 | 1 | 2 |
| 10 | 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 2 | 4 | 1 | 4 | 4 | 3 |
| 11 | 2 | 2 | 1 | 3 | 2 | 2 | 2 | 1 | 4 | 3 | 4 | 2 | 4 | 4 |
| 12 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 3 | 2 | 1 | 4 | 4 | 3 |
| 13 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 3 | 2 | 2 | 4 |
| 14 | 2 | 2 | 1 | 3 | 2 | 2 | 2 | 1 | 2 | 4 | 2 | 2 | 1 | 2 |
| 15 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 3 | 4 | 3 | 1 | 3 | 2 |
| 16 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 4 | 2 | 3 | 4 | 3 | 3 |
| 17 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 3 | 3 | 1 | 1 |
| 18 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 4 | 4 | 3 | 2 | 4 |
| 19 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 1 | 4 | 4 | 2 | 3 | 1 | 4 |
| 20 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 2 | 4 | 3 | 4 | 2 | 1 |
| 21 | 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 2 | 2 | 2 | 2 | 1 | 2 |
| 22 | 1 | 2 | 3 | 2 | 1 | 1 | 2 | 2 | 3 | 1 | 4 | 2 | 1 | 1 |
| 23 | 1 | 2 | 3 | 2 | 1 | 1 | 2 | 3 | 4 | 2 | 1 | 3 | 4 | 4 |
| 24 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 1 | 4 | 4 | 3 | 3 | 3 |
| 25 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 4 | 1 | 4 | 2 | 1 |
| 26 | 1 | 2 | 3 | 2 | 1 | 1 | 2 | 2 | 1 | 2 | 3 | 3 | 4 | 1 |
| 27 | 2 | 2 | 1 | 3 | 1 | 1 | 2 | 2 | 4 | 4 | 4 | 3 | 2 | 2 |
| 28 | 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 4 | 4 | 2 | 2 | 2 |
| 29 | 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 3 | 4 | 1 | 4 | 1 | 4 |

```
50 rows fetched in 0,0357s (0,0022s)
```

**Figure 22** Initial population of the case study.

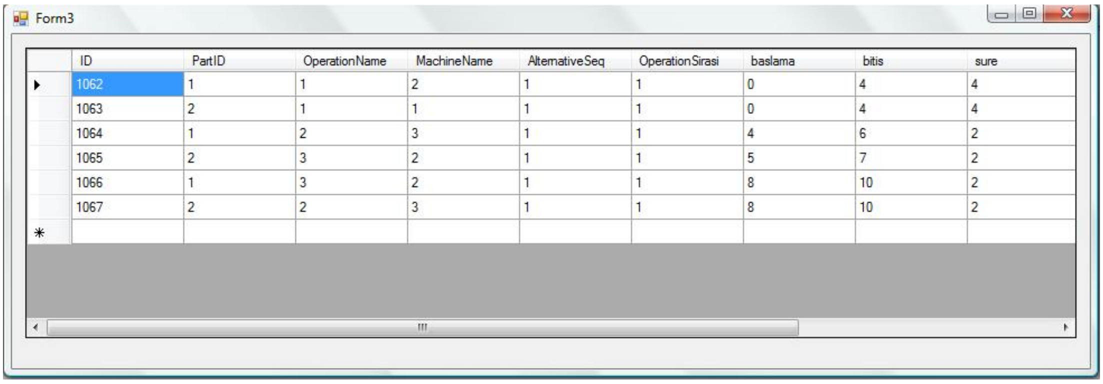
The user enters the iteration number and clicks on the AIS Button in order to start CLONALG algorithm. To see the results, the user should click on the RESULTS Button.



| DP4 | DP5 | DP6 | ID1 | flowtime | earlinesstardiness | maliyet | SJ | RJ | QJ |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 1 | 4 | 171 | 10 | 5 | 23,32441799279... | 0,130222222222... | 0,088 | 0,109111111111... |
| 3 | 1 | 4 | 12 | 10 | 5 | 24,21449332277 | 0,142444444444... | 0,088 | 0,115222222222... |
| 3 | 1 | 4 | 141 | 10 | 5 | 24,21449332277 | 0,142444444444... | 0,088 | 0,115222222222... |
| 3 | 1 | 4 | 21 | 10 | 5 | 24,21449332277 | 0,142444444444... | 0,088 | 0,115222222222... |
| 3 | 1 | 4 | 147 | 11 | 4 | 26,47819086302... | 0,174888888888... | 0,066 | 0,120444444444... |
| 3 | 1 | 4 | 145 | 11 | 4 | 26,47819086302... | 0,174888888888... | 0,066 | 0,120444444444... |
| 3 | 1 | 4 | 150 | 11 | 4 | 26,47819086302... | 0,174888888888... | 0,066 | 0,120444444444... |
| 3 | 1 | 4 | 143 | 11 | 4 | 26,47819086302... | 0,174888888888... | 0,066 | 0,120444444444... |
| 3 | 1 | 4 | 91 | 11 | 4 | 26,47819086302... | 0,174888888888... | 0,066 | 0,120444444444... |

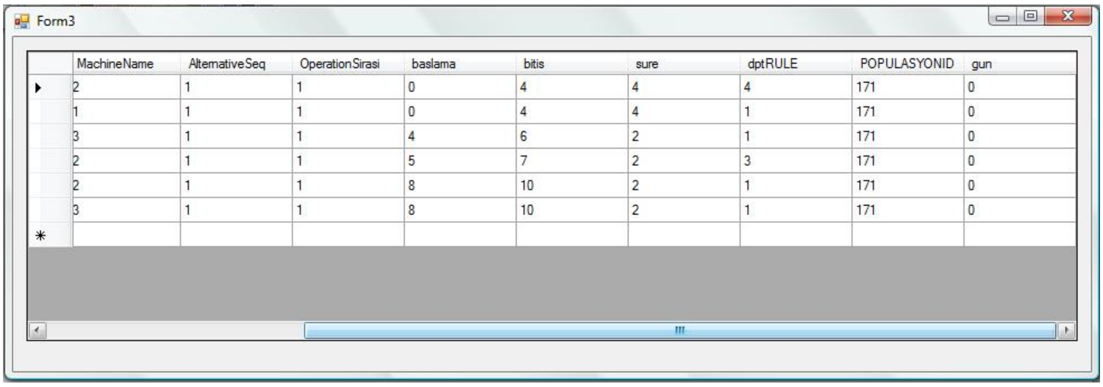
**Figure 23** Results of the case study found by the developed program.

All the candidates are listed in the RESULTS Form ranked according to their Qj (affinity values). The detailed information about each candidate is also showed in the datagram. To see a schedule of a particular candidate in the form, the user should click on the ID of the candidate and then click on the SCHEDULE Button.



| ID | PartID | OperationName | MachineName | AlternativeSeq | OperationSirasi | baslama | bitis | sure |
|---|---|---|---|---|---|---|---|---|
| 1062 | 1 | 1 | 2 | 1 | 1 | 0 | 4 | 4 |
| 1063 | 2 | 1 | 1 | 1 | 1 | 0 | 4 | 4 |
| 1064 | 1 | 2 | 3 | 1 | 1 | 4 | 6 | 2 |
| 1065 | 2 | 3 | 2 | 1 | 1 | 5 | 7 | 2 |
| 1066 | 1 | 3 | 2 | 1 | 1 | 8 | 10 | 2 |
| 1067 | 2 | 2 | 3 | 1 | 1 | 8 | 10 | 2 |

**Figure 24** Schedule of the best candidate found by the developed program.



| MachineName | AlternativeSeq | OperationSirasi | baslama | bitis | sure | dptRULE | POPULASYONID | gun |
|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 0 | 4 | 4 | 4 | 171 | 0 |
| 1 | 1 | 1 | 0 | 4 | 4 | 1 | 171 | 0 |
| 3 | 1 | 1 | 4 | 6 | 2 | 1 | 171 | 0 |
| 2 | 1 | 1 | 5 | 7 | 2 | 3 | 171 | 0 |
| 2 | 1 | 1 | 8 | 10 | 2 | 1 | 171 | 0 |
| 3 | 1 | 1 | 8 | 10 | 2 | 1 | 171 | 0 |

**Figure 25** Schedule of the best candidate found by the developed program (Cont.)

The schedule of a candidate is given in detail in the SCHEDULE Form. The starting and ending times, the dispatching rules used in each step are shown in the datagram.

# CHAPTER 6

# CONCLUSIONS AND RECOMMENDATIONS

## 6.1 Conclusions

A literature survey has been carried out with a statistical study in this thesis. Each study is explained in detail and the objective of each study is mentioned. The examined studies are classified according to integration approaches and implementation models/architectures. The distribution of the studies over years, the percentage of the used approaches and methods are shown with figures. The results of the survey showed that most of the studies used DPP and NLPP approaches and during the latest years DPP approach has been more studied than NLPP. The most common performance criteria in the overall studies was minimizing the makespan. The most of the studies have preferred using agent-based architecture with different optimization algorithms. The most preffered algorithm was GA with a huge gap between the other algorithms. This popularity of GA may be because of the easy implementation and the comprehensible structure of the algoritm. All the statistics showed that there have been a lot of effort on the IPPS problem over years and this effort have increased a lot during the latest years.

The IPPS problem with setup and machine capacity constraints was dealt in order to minimize total flow time, total cost and total tardiness/earliness. The approach of (Baykasoğlu and Özbakır, 2009), "The grammatical representation of generic process planning" is adopted for generating generic process plans. The CLONALG is used as the optimization algorithm for the proposed IPPS problem, where the performance criteria are total flow time, total processing cost and total tardiness/earliness.

A computer program is developed in C# programming language, with a user-friendly interface. The inputs (part, machine, operation informations; setup times; operation sequences, parts' due dates, population size, iteration number...) are entered by the user, and when the program is run, it gives the best processing plan and schedule as an output for the entered problem.

Algorithm performance is tested on case studies. The results of the studies presented in this thesis confirm that, it is possible to obtain better performance and to realize the flexibility available in the system during scheduling by making use of the proposed approach in environments where there exists setup times and machine capacity constraints.

### 6.2 Recommendations

The proposed method is applicable for non-dynamic production environment where the number of parts, machines and operations are predefined in the system. The system knows the production requirements before starting the process planning and the scheduling phases. The future studies can focus on the dynamic scheduling of the parts, where a new order might come during the production, and the previous schedule is updated according to the new coming order immediately, considering the required priorities.

The proposed case study problems are created for this study. The parameters and values such as due dates of the parts are determined approximately, considering the number of operations, number of machines and working hour percentage in a day. A more advanced method can be developed for the parameter and value determinations.

The developments in the technology is growing rapidly, thus creates more complex production environments with increased flexibilities and requirements in shop floors.

As a result, the studies on integrating the process planning and scheduling phases become more important and applicable on real life problems.

# REFERENCES

Aldakhilallah, K.A., Ramesh, R.(1999), Computer-integrated process planning and scheduling (CIPPS): intelligent support for product design, process planning and control. *International Journal of Production Research.* **37: 3**, pp. 481 - 500

Amin-Naseri, M.R., Afshari, A.J. (2011). A hybrid genetic algorithm for integrated process planning and scheduling problem with precedence constraints. *International Journal of Advanced Manufacturing Technology.* DOI 10.1007/s00170- 001-3488-y

Baker R.P. and Maropoulos P.G. (2000). An architecture for the vertical integration of tooling considerations from design to process planning. *Robotics and Computer Integrated Manufacturing.* **16,** pp. 121 - 131

Baykasoğlu, A. (2002). Linguistic based meta-heuristic optimization model for flexible job shop scheduling. International Journal of Production Research, 40:17, pp. 4523 - 4543

Baykasoğlu, A., Ozbakir L. (2009). A grammatical optimization approach for integrated process planning and scheduling. *Journal of Intelligent Manufacturing.* **20**, pp. 211 - 221

Benayoun, R., Roy, B., Sussman, N. (1966). Manual de reference du programme electre, Note de Synthese et Formation. *Direction Scientifique SEMA, Paris, Franch*, **25**

Brans, J.P. (1982). L'ingéniérie de la décision. Elaboration d'instruments d'aide à la décision. Méthode PROMETHEE . *Université Laval, Québec, Canada.*

Brownlee J. (2005). Clonal Selection Theory & CLONALG The Clonal Selection Classification Algorithm (CSCA). *Faculty of ICT, Swinburne University of Technology, Australia, Technical Report* **1:2**

Cai, N., Wang, L., Feng, H. (2009). GA-based adaptive setup planning toward process planning and scheduling integration. *International Journal of Production Research.* **47: 10**, 2745 - 2766

Carton, B. A., Ray, S. T. (1991). ALPS—A language for process specification. *International Journal of Computer Integrated Manufacturing,* 4, 105–113.

Chan, F.T.S., Zhang, J., Li, P. (2001). Modelling of integrated, distributed and cooperative process. *Proceedings of the Institution of Mechanical Engineers.* **215:10**, pp. 1437 - 1451

Chan, F.T.S., Kumar, V., Tiwari, M.K. (2006). Optimizing the performance of an integrated process planning and scheduling problem an AIS FLC based approach. *IEEE Conference on Cybernetics and Intelligent Systems.* pp. 1-8

Chang, T.C., Wysk, R.A. (1985). An introduction to automated process planning systems. California: Prentice Hall, 230 p.

Cho, H., Smith, J.S., Wysk R. A. (1997). An intelligent workstation controller for integrated planning and scheduling of FMS cell. *Production Planning & Control,* **8: 6**, pp. 597 - 607

Cho K.K., Oh, J.S., Ryu, K.R., Choi, H.R. (1998). An integrated process planning and scheduling system for block assembly in shipbuilding. *CIRP Annals – Manufacturing Technology*, **47:1**, pp. 419 - 422.

Chryssolouris, G., Chan, S. (1985). An integrated approach to process planning and scheduling. *CIRP Annals – Manufacturing Technology*, **34:1**, pp. 413 – 417.

Chu, M.T., Shyu, J., Tzeng, G.H., Khosla, R. 2007. Comparison Among Three Analytical Methods for Knowledge Communities Group-Decision Analysis. *Expert Systems with Applications*. 33, pp. 1011 - 1024

Conway R.W., Maxwell W.L., Miller L.W. (1967). Theory of scheduling Massachusetts: Addison-Wesley

De Castro, L.N., Von Zuben, F.J. (2000). The Clonal Selection Algorithm with Engineering Applications. *GECCO 2000, Las Vegas, Nevada, USA, July 8*

De Castro, L.N., Von Zuben F.J. (2002). Learning and optimization Using the Clonal Selection Principle. *In the Special Issue on Artificial Immune Systems of the Journal IEEE Transactions on Evolutionary Computation*. **6**:3.

Dong J., Jo H.H. and Parsaei H.R. (1992). A feature-based dynamic process planning and scheduling. *Computers & Industrial Engineering.* **23**, pp. 141-144.

Engin, O., Döyen, A. (2004). Artificial Immune Systems And Applications In Industrial Problems. *G.U. Journal of Science*. **17:1**, pp. 71 - 84

Engin, O., Döyen, A. (2007). A New Approach to Solve Flow Shop Scheduling Problems by Artificial Immune Systems. *Doğuş Üniversitesi Dergisi*. 8:1, pp.12- 27

Fujii, N., Inoue, R., Ueda, K. (2008). Integration of process planning and scheduling using multi-agent learning. *The 41st CIRP Conference on Manufacturing Systems.* **7**, pp. 297 - 300

Gan, P.Y., Lee, K.S. (2002). Scheduling of flexible sequenced process plans in a mould manufacturing shop. *International Journal of Advanced Manufacturing Technology.* **20**, pp. 214–222

Gao H., Zeng J. and Sun G. (2002). Multi-agent approach for planning and scheduling of integrated steel processes. *IEEE International Conference on Systems, Man and Cybernetics.* **6**, pp 1 - 6.

Giffler, B., & Thompson, G. (1960). Algorithms for solving production scheduling problems. *Operations Research*, **8**, pp. 487–503.

Gindy, N., Saad, S., Yue, Y. (1999). Manufacturing responsiveness through integrated process planning and scheduling. *International Journal of Production Research.* **37:11**, pp. 2399-2418.

Gu P., Balasubramanian S. and Norrie D.H. (1997). Bidding-based process planning and scheduling in a multi-agent system. *Computers & Industrial Engineering.* **32:2**, pp. 477-496

Guo, Y. W. , Li, W. D. , Mileham, A. R. andOwen, G. W., (2009a). Optimisation of integrated process planning and scheduling using a particle swarm optimisation approach. *International Journal of Production Research.* **47: 14**, pp. 3775 - 3796

Guo, Y.W., Li, W.D., Mileham, A.R., Owen, G.W., (2009b). Applications of particle swarm optimization in integrated process planning and scheduling. *Robotics and Computer-Integrated Manufacturing.* **25**, pp. 280 - 288

Haddadzade M., Razfar M.R., and Farahnakian M., (2009). Integrating process planning and scheduling for prismatic parts. *World Academy of Science, Engineering and Technology.* **51,** pp. 64 - 67

Hashemipour, M. (2004). Integration of process planning and scheduling for outsourcing in the apparel industry. *Journal of the Textile Institute.* **95: 1**, pp. 9-18

Hildum, D.W., Sadeh, N.M., Laliberty, T.J., Smith, S.E., McA'Nulty, J., Kjenstad D. (1996). Mixed - initiative management of integrated process planning and production scheduling solutions. *Proceedings of the AI and Manufacturing Research Planning Workshop, AIII.*

Hildum D.W., Sadeh N.M., Laliberty T.J., Smith S.E., McA'Nulty J., Kjenstad D. (1997). Blackboard agents for mixed-initiative management of integrated process-planning/production-scheduling solutions across the supply chain. *Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence (IAAI-97)*

Huang S.S., Zhang H.C., Smith M.L. (1995). A progressive approach for the integration of process planning and scheduling. *IIE Transactions.* **27**, pp. 456 -464.

Hwang, C.L., Yoon, P. (1981). Multiple Attribute Decision Making In: Lecture Notes in Economics and Mathematical Systems. *Springer-Verlag-Berlin*

Jablonski S., Reinwald B., Ruf T. (1990). Integration of process planning and job shop scheduling for dynamic and adaptive manufacturing control. *IEEE Proceedings Rensselaers Second International Conference on Computer Integrated Manufacturing.*

Jain A., Jain P. K., Singh I. P. (2006). An integrated scheme for process planning and scheduling in FMS. *International Journal Advanced Manufacturing Technologies.* **30,** pp. 1111–1118

Kacem, I., Hammadi, S., Borne, P. (2002a). Pareto-optimality approach for flexible job-shop scheduling problems: Hybridization of evolutionary algorithms and fuzzy logic. *Mathematics and Computers in Simulation*. **60**, pp. 245 - 276.

Kacem, I., Hammadi, S., Borne, P. (2002b). Approach by localization and multi-objective evolutionary optimization for flexible job shop scheduling problems. *IEEE Transactions on Systems, Man and Cybernetics-Part C*. **32:1**, 1 - 13.

Kayacan, M.C., Filiz, I.H., Sönmez, A.I., Baykasoglu, A., Dereli T. (1996). OPPS-ROT An optimized process planning system for rotational parts. *Computers in Industry.* **32**, pp. 181 - 195.

Kim, Kun-Hyung, Egbelu Pius, J. (1998). A mathematical model for job shop scheduling with multiple process plan consideration per job. *Production Planning & Control.* **9: 3,** pp. 250 - 259

Kim, K.H., Egbelu, P. J. (1999). Scheduling in a production environment with multiple process plans per job. *International Journal of Production Research.* **37: 12**, pp. 2725 - 2753

Kim, Y.K., Park, K., Ko, J. (2003). A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling. *Computers & Operations Research.* **30,** pp. 1151 - 1171

Kis T., Kiritsis D. and Xirouchakis P. (2000). A petri net model for integrated process and job shop production planning. *Journal of Intelligent Manufacturing* **11**, pp. 191 - 207

Krothapalli, N., Deshmukh, A. (1999). Design of negotiation protocols for multi-agent manufacturing systems. *International Journal of Production Research*. **37,** pp. 1601 - 1624.

Kumar M., Rajotia S. (2003). Integration of scheduling with computer aided process planning. *Journal of Materials Processing Technology.* **138**, pp. 297 - 300

Kumar M., Rajotia S. (2005). Integration of process planning and scheduling in a job shop environment. *International Journal of Advanced Manufacturing Technology.* **28**, pp. 109 - 116

Lee H., Kim S.-S. (2001). Integration of process planning and scheduling using simulation based genetic algorithms. *International Journal of Advanced Manufacturing Technology.* **18**, pp. 586 - 590

Leung, C.W., Wong, T.N., Mak, K.L., Fung, R.Y.K. (2009). Integrated process planning and scheduling by an agent-based ant colony optimization. *Computers & Industrial Engineering.* **59:1**, pp.166 - 180.

Li, W. D. and McMahon, C. A. (2007). A simulated annealing-based optimization approach for integrated process planning and scheduling. *International Journal of Computer Integrated Manufacturing*. **20: 1**, pp. 80 - 95

Li, W.D., Gao, L., Li, X.Y., Guo, Y. (2008a). Game theory-based cooperation of process planning and scheduling. *12th International Conference on Computer Supported Cooperative Work in Design.* pp. 841 - 845

Li, X., Gao, L., Zhang, G., Zhang, C., Shao, X. (2008b). Genetic algorithm for integration of process planning and scheduling problem. *ICIRA 2008, Part II, LNAI* **5315**, pp. 495 - 502

Li, X., Gao, L., Shao, X., Zhang, C., Wang, C. (2009a). Mathematical modeling and evolutionary algorithm-based approach for integrated process planning and scheduling. *Computers &Operations Research.* ***37**, pp. 656 - 667*

Li, X., Li, W., Gao, L., Zhang, C., Shao, X. (2009b). Multi-agent based integration of process planning and scheduling. *Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design.*

Li, Z., Ierapetritoua, M. (2009c). Integration of planning and scheduling and consideration of uncertainty in process operations. *Computer Aided Chemical Engineering. In: 10^th^ international symposium on process systems engineering: Part A, 16-20 August, Salvador-Bahia, Brazil.* **27**, pp.87-94.

Li, X., Gao, L., Zhang, C., Shao, X. (2010a). A review on integrated process planning and scheduling. *International Journal of Manufacturing Research.* **5:2**, pp. 161 - 180.

Li, X.Y., Shao, X.Y., Gao, L., Qian, W.R. (2010b). An effective hybrid algorithm for integration of process planning and scheduling. *International Journal of Production Economics.* **126**, pp. 289 - 298

Li, X., Zhang, C., Gao, L., Li, W. (2010c). An agent-based approach for integrated process planning and scheduling. *Expert Systems With Applications,* **37**, pp. 1256 - 1264.

Li X. , Gao L., Li W. (2012). Application of game theory based hybrid algorithm for multi-objective integrated process planning and scheduling. *Expert Systems with Applications.* **39**, pp. 288 - 297

Lian, K., Zhang, C., Gao, L., Li, X. (2011). Integrated process planning and scheduling using an imperialist competitive algorithm. *International Journal of Production Research.* DOI :10.1080/00207543.2011.622310

Lihong, Q., Shengping, L.V. (2012). An improved genetic algorithm for integrated process planning and scheduling. *The International Journal Of Advanced Manufacturing Technology.* 58, pp. 5 - 8

Lim, M.K., Zhang, Z. (2003). A multi-agent based manufacturing control strategy for responsive manufacturing. *Journal of Materials Processing Technology.* 139 pp. 379 - 384

Lim, M.K., Zhang, Z. (2004). An integrated agent-based approach for responsive control of manufacturing resources. *Computers & Industrial Engineering.* **46**, pp. 221 - 232

Mamalis, A.G., Malagardis, I., Kambourşs, K. (1996). On-line integration of a process planning module with production scheduling. *International Journal of Advance Manufacturing Technology.* **12**, pp. 330 - 338

Moon, C., Lee, Y.H., Jeong, C.S., Yun, Y. (2008). Integrated process planning and scheduling in a supply chain. *Computers & Industrial Engineering.* **54**, pp. 1048 - 1061

Moon, C., Kim, J., Hur, S. (2002). Integrated process planning and scheduling with minimizing total tardiness in multi-plants supply chain. *Computers & Industrial Engineering.* **43**, pp. 331 - 349

Moon C., Seo Y. (2005). Evolutionary algorithm for advanced process planning and scheduling in a multi-plant. *Computers & Industrial Engineering.* **48**, pp. 311 - 325

Morad, N., Zalzala, A., (1999). Genetic algorithms in integrated process planning and scheduling. *Journal of Intelligent Manufacturing.* **10**, pp. 169 – 179

Nejad, H.T.N., Sugimura, N., Iwamura, K., Tanimizu, Y. (2008). Integrated dynamic process planning and scheduling in flexible manufacturing systems via autonomous agents. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*. **2:4,** pp. 719 - 734

Nejad H.T.N., Sugimura N., Iwamura K., Tanimizu Y. (2008). Agent-based dynamic process planning and scheduling in flexible manufacturing systems. *Manufacturing Systems And Technologies For The New Frontier.* **7**, pp. 269 - 274

Nejad H.T.N., Sugimura N., Iwamura K. (2011). Agent-based dynamic process planning and scheduling in flexible manufacturing systems. International Journal of Production Research, **49: 5**, 1373 – 1389

Nwana, H.S., Ndumu, D.T. (1997). An introduction to agent technology. *Software Agents and Soft Computing.* pp. 42 – 58

Octavia, T. (2003). A modified Giffler and Thompson algorithm combined with dynamic slack time for solving dynamic schedule problems. *Jurnal Teknik Industri,* **5**, pp. 71 - 81

Omar, M.K., Teo, S.C. (2007). Hierarchical production planning and scheduling in a multi-product, batch process environment. *International Journal of Production Research.* **45: 5**, pp. 1029 - 1047

Opricovic, S., ,Tzeng, G. H. (2002). Multicriteria planning of post-earthquake sustainable reconstruction. *Computer-Aided Civil and Infrastructure Engineering.* **17:3**, pp. 211 - 220

Opricovic, S.,, Tzeng, G. H. (2003). Defuzzification within a fuzzy multicriteria decision model. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems.* **11:5,** pp. 635 - 652

Opricovic, S.,Tzeng, G. H. (2004). Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. *European Journal of Operational Research,* **156:2**, pp. 445 - 455

Opricovic, S., Tzeng, G. H. (2007). Extended VIKOR method in comparison with outranking methods. *European Journal of Operational Research.* 178:2, pp. 514 - 529

Palmer, G. (1996). A simulated annealing approach to integrated production scheduling. *Journal of Intelligent Manufacturing.* **7**, pp. 163 - 176.

Park, B.J., Choi, H.R. (2006a). A genetic algorithm for integration of process planning and scheduling in a job shop. *AI 2006: Advances In Artificial Intelligence.* **4304,** pp. 647 - 657

Park B.J., Choi H.R. (2006b). Integration of process planning and job shop scheduling using genetic algorithm. *Proceedings of the 6th WSEAS International Conference on Simulation, Modelling and Optimization.* pp. 13 - 18

Pawlewski, P., Golinska, P., Fertsch, M., Trujillo, J.A., Pasek, Z.J. (2009). Multiagent approach for supply chain integration by distributed production planning, scheduling and control system. *International Symposium On Distributed Computing And Artificial Intelligence.* **50**, pp. 29 – 37

Phanden R.K., Jain A., Verma R. (2011). Integration of process planning and scheduling: a state-of-the-art review. *International Journal of Computer Integrated Manufacturing.* **24:6**, pp. 517 - 534

Rajkumar, M., Asokan, P., Page, T., Arunachalam, S. (2010). A GRASP Algorithm for the integration of process planning and scheduling in a flexible job-shop. *International Journal of Manufacturing Research.* **5:2**, pp. 230 - 251

Saaty, L.T. (1980). The Analytic Hierarchy Process. *McGraw - Hill Comp., U.S.A.*

Sadeh, N.M., Hildum, D.W., Laliberty, T.J., McA'Nulty, J., Kjenstad, D., Tseng, A. (1998). A blackboard architecture for integrating process planning and scheduling. *Concurrent Engineering: Research and Applications.* **6:2**, pp. 88 - 100

Saygin, C., Kilic, S.E. (1999). Integrating flexible process plans with scheduling in flexible manufacturing systems. *International Journal of Advanced Manufacturing Technology.***15,** pp. 268 - 280

Shao, X., Li, X., Gao, L., Zhang, C. (2009). Integration of process planning and scheduling - A modified genetic algorithm-based approach. *Computers & Operations Research.* **36**, pp. 2082 - 2096

Shen, W., Wang, L., Qi Hao, Q. (2006). Agent-based distributed manufacturing process planning and scheduling, a state-of-the-art survey. *IEEE Transactions on Systems, Man, And Cybernetics—Part C: Applications and Reviews*. **36:4**, pp. 563 - 577

Shobrys, D.E., White, D.C. (2000). Planning, scheduling and control systems: Why can they not work together? *Computers and Chemical Engineering*. **24**, pp. 163 - 173

Shrestha, R., Takemoto, T., Ichinose, K., (2008). A study on integration of process planning and scheduling system for holonic manufacturing with modification of process plans. *International Journal of Manufacturing Technology and Management.* **14:3-4,** pp. 359 - 378

Shukla, S., Tiwari, M., Son, Y. (2008). Bidding-based multi-agent system for integrated process planning and scheduling: a data mining and hybrid tabu-SA algorithm oriented approach. *International Journal of Advanced Manufacturing Technology.* **38**, pp. 163 - 175

Sugimura, N., Hino, R., Moriwaki, T. (2001). Integrated process planning and scheduling in holonic manufacturing systems. *Proceedings of the 4th IEEE International Symposium on Assembly and Task Planning Soft Research Park.* pp. 250 - 255

Sugimura, N., Shrestha, R., Inoue, J. (2003). Integrated process planning and scheduling in holonic manufacturing systems - optimization based on shop time and machining cost. *Proceedings of the 5th IEEE International Symposium on Assembly and Task Planning.* pp. 36 - 41

Sugimura N., Shrestha R., Tanimizu Y., Iwamura K. (2006). A study on integrated process planning and scheduling system for holonic manufacturing. *Integrated Process Planning and Scheduling for Holonic Manufacturing – Book Chapter #13*

Tan, W., Khosnevis, B. (2000). Integration of process planning and scheduling— a review. *Journal of Intelligent Manufacturing.* **11**, pp. 51 - 63

Tan W., Khosnevis B. (2004). A linearized polynomial mixed integer programming model for the integration of process planning and scheduling. *Journal of Intelligent Manufacturing.* **15**, pp. 593 - 605

Tönshoff, H.K., Woelk, P.-O., Herzog, O., Timm, I.J. (2001). Integrated process planning and production control - a flexible approach using co-operative agent systems. *Initiatives of Precision Engineering at The Beginning of A Millennium.* **4**, pp. 857 – 861

Tonshoff, H. K.,Beckendorff,U., Andress, N. (1989). FLEXPLAN—A concept for intelligent process planning and scheduling. In Proceedings of the CIRP International Workshop on Computer Aided Process Planning, Hanover University, Germany, Sept. 21–22.

Triantaphyllou, E., Shu, B., Sanchez, N., Ray, T. (1998). Multi-Criteria Decision Making: An Operations Research Approach. *Encyclopedia of Electrical and Electronics Engineering, (J.G. Webster, Ed.), John Wiley & Sons, New York, NY,* **15**, pp. 175 - 186

Uckun, S., Bagchi, S., Kawamura, K. (1993). Managing genetic search in job shop scheduling. *IEEE Expert.* **8:5,** pp. 15 - 24

Ueda, K., Fujii, N., Inoue, R. (2007). An emergent synthesis approach to simultaneous process planning and scheduling. *Annals of the CIRP.* **56:1**

Wang, Y.F., Zhang, Y.F., Fuh, J.Y.H. (1998). A web-based integrated process planning and scheduling system. *4th IEEE Conference on Automation Science and Engineering.* pp. 662 - 667

Wang, J., Zhang, Y. and Nee, A. (2002). Integrating process planning and scheduling with an intelligent facilitator. *10th International Manufacturing Conference in China (IMCC2002).* **2**, pp. 152 - 156.

Wang, Y.F. (2008). An integrated approach to reactive scheduling subject to machine breakdown. *Proceeding of IEEE International Conference on Automation and Logistics, 1-3 September Qingdao, China, pp. 542-547.*

Wang, Y.F., Zhang, Y.F., Nee, A.Y.C., Wang, Y.F., Fuh, J.Y.H. (2009a). Reducing tardy jobs by integrating process planning and scheduling functions. *International Journal of Production Research.* **47:21**, pp. 6069-6084

Wang, Y.F., Zhang, Y.F., Fuh, J.Y.H. (2009b). An agent-based distributed process planning and scheduling system.

Wang, L., Feng, H.-Y., Cai, N. (2003). Architecture design for distributed process planning. *Journal of Manufacturing Systems.* **22:2**, pp. 99 - 115

Weintraub A., Cormier D., Hodgson T., King R., Wilson, J., Zozom, A. (1998). Scheduling with alternatives a link between process planning and scheduling. *IIE Transactions*. **31:11**, pp. 1093 - 1102

Wong, T. N., Leung, C., W., Mak, K. L., Fung, R. Y. K. (2005). An agent-based negotiation approach to integrate process planning and scheduling. *International Journal of Production Research.* **44: 7**, pp. 1331 - 1351

Wong, T. N., Leung, C. W., Mak, K. L., Fung, R. Y. K. (2006). Integrated process planning and scheduling/rescheduling—an agent-based approach. *International Journal of Production Research.* **44: 18**, pp. 3627 - 3655

Wu, S.H., Fuh, J.Y.H., Nee, A.Y.C. (2002). Concurrent process planning and scheduling in distributed virtual manufacturing. *IIE Transactions.* **34**, pp. 77 - 89

Yang, Y.-N., Parsaei, H.R., Leep, H.R. (2001). A prototype of a feature-based multiple-alternative process planning system with scheduling verification. *Computers & Industrial Engineering.* **39**, pp. 109 - 124

Yu, P. L. (1973). A class of solutions for group decision problems. *Management Science.* 19:8, pp. 936 - 946

Zattar, I.C., Ferreira, J.C.E., Granado J.G.G., de Sousa C.H.B. (2007). Integrating Manufacturing Process Planning with Scheduling via Operation-Based Time-Extended Negotiation Protocols. *Complex Systems Concurrent Engineering.* **6**, pp. 329 - 336

Zattar, I.C., Ferreira, J.C.E., Rodrigues, J.G.G.G., de Sousa, C.H.B. (2010). A multi-agent system for the integration of process planning and scheduling using operation-based time-extended negotiation protocols. *International Journal of Computer Integrated Manufacturing,* **23: 5**, pp. 441 - 452

Zeleny, M. (1982). Multiple criteria decision making. *New York: McGraw-Hill.*

Zhang, H. C., Alting, L. (1989). Computer aided process planning: The state-of-the-art survey. *International Journal of Production Research.* 27:4, pp. 553 – 585

Zhang, H. C., Alting, L. (1994). Computerized manufacturing process planning systems. *London: Chapman and Hall.*

Zhang, W.J., Xie, S. Q. (2007). Agent technology for collaborative process planning: a review. International Journal of Advanced Manufacturing Technology. **32**, pp. 315- 325

Zhang, D., Zhang, H.C. (1999). A simulation study of an object-oriented integration testbed for process planning and production scheduling. *The International Journal of Flexible Manufacturing Systems.* **11**, pp. 19 - 35

Zhang, D., B.E., M.E. (1998). Object-oriented modeling for the integrated process planning and production scheduling system. *The Graduate Faculty of Texas Tech University in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy*

Zhang, Y. F., Saravanan, A. N., Fuh, J. Y. H. (2003a). Integration of process planning and scheduling by exploring the flexibility of process planning. *International Journal of Production Research.* **41: 3**, pp. 611 - 628

Zhang, J., Gao, L., Chan, F.T.S., Li, P. (2003b). A holonic architecture of the concurrent integrated process planning system. *Journal of Materials Processing Technology.* 139, pp. 267–272

Zhang, H.C., Merchant, M.E. (1993). IPPM – a prototype to integrate process planning and job scheduling functions. *CIRP Annals – Manufacturing Technology,* **42:1**, pp. 513 - 518.

Zhanjie, W., Ju, T. (2008). The research about integration of process planning and production scheduling based on genetic algorithm. *International Conference on Computer Science and Software Engineering.* **1**, pp. 9 - 12

Zhao, F., Hong, Y., Yu, D., Yang, Y. (2004). A genetic algorithm based approach for integration of process planning and scheduling. *Proceedings of the 2004 International Conference on Intelligent Mechatronics and Automation.*

Zhao F.-Q., Zhang Q.-Y., Yang Y.-H. (2006). An improved particle swarm optimization algorithm and fuzzy inference system based approach to process planning and production scheduling integration in holonic system. *Proceedings of the Fifth International Conference on Machine Learning and Cybernetics. pp. 396 - 401*

Zhao, F., Hong, Y., Yu, D. , Yang, Y., Zhang, Q. (2010). A hybrid particle swarm optimization algorithm and fuzzy logic for process planning and production scheduling integration in holonic manufacturing systems. *International Journal of Computer Integrated Manufacturing.* **23:1**, 20 - 39