A Generic Model for the Selection of Manufacturing Method Based on Multi Objective and Multi Function Criteria

[1]Girish R. Naik, [2]Dr. V.A.Raikar, [3]Dr. Poornima G. Naik
[1]Production Dept. KIT's College of Engineering, Kolhapur, India
[2]Sanjay Ghodawat Group of Institutions, Atigre, Kolhapur, India
[3]Department of Computer Studies, CSIBER, Kolhapur, India

Abstract
In today's highly competitive era and a dynamic nature of business environment installing a suitable manufacturing method is extremely challenging, expensive and a time consuming project. In order to improve a manufacturing process more than 110 manufacturing methods have been proposed. Based on the objectives aimed at and the functions focused by the organization it is inevitable for every organization to deploy a suitable manufacturing process. The process should be flexible enough to accommodate reasonable changes in design. This poses a great challenge and a responsibility on the shoulders of a manager to select the most effective and an economical manufacturing process under existing circumstances. Recently, authors have developed various models and the tools implementing those models for the selection of manufacturing method based on a single objective and/or single function and a multi objective employing both the crisp and a fuzzy model. Each case is taken care of by a separate model and there is no single generic model taking care of all the cases. In this paper, the authors have devised a novel solution to the same by representing the selected methods in a matrix format. Incorporating the various objectives/functions in a model boils down to simple union operation on a set of matrices with different weights assigned to the method class, method grade and manufacturing objective or a function. Without such a representation, the number of various permutations and combinations to be considered for the selection of manufacturing method based on multi objectives and multi functions grows exponentially. The model is implemented using 3-tier architecture using VB in presentation tier, Java in Business Logic Tier and MS-Access storing domain specific information in data tier. The model is tested for the selection of manufacturing methods based on multi objective and/or multi function criteria in a hypothetical manufacturing organization.

Keywords
Business Logic, Data Tier, Method Grade, Method Selection Matrix, Presentation Tier, Three-tier Architecture.

I. Introduction

The framework for the organization's competitive advantage in the market relies substantially on the strategy and operational tactics employed by the organization. In the early days, the focus was laid on the excellence in manufacturing. This provided many options for the organization to improve manufacturing system. Later on focus shifted to customerization of organization that led to the development of many new advanced manufacturing processes. Installing a right manufacturing process taking into account the organizational objectives is a challenging task as installing right manufacturing method is an expensive and a time consuming task. Hence it needs a meticulous assessment of organization's many-fold objectives and functions for the selection of the most appropriate manufacturing method which conforms to organization's ultimate goals. Further, considering the dynamic nature of business environment and large number of objectives and functions and their impact on various manufacturing methods, a suitable model is required to be developed at first place in order to aid the manager in the quick selection of appropriate manufacturing method. The model is required to be generic enough to account for different organizational objectives and/or functions.

Manufacturing methods are categorized primarily into 5 different categories based on technological solution, software solution, management solutions, philosophical solutions, auxiliary solutions. To assist managers in selecting the best method to achieve certain criteria, two mapping methods are available, one based on the objectives and the second one based on organizational functions. Giden Halevi [1] has proposed 16 different manufacturing objectives, 24 different organizational functions grouped in four categories containing six functions each and 110 different manufacturing methods. The suitability of each method to a specific objective is graded according to the following grades.

- Excellent
- Very Good
- Good
- Fair

In the following section we present a generic model for manufacturing method selection by constructing a method selection matrix.

II. Literature Review

There exist a fistful of papers on selection processes based on multi criteria which focus mainly on software selection, technology selection, system project selection and supplier selection [2,11]. Multi criteria decision making method is employed by R.V.Rao et. al.[12] where they have presented a decision making framework using a multiple criteria decision making method via Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE) which has been integrated with AHP Recently, Khan et al [13] have proposed a a multi- criteria decision making approach for selecting suppliers under uncertainly. Their approach comprises of two phases, in the first phase they have identified the criteria and sub criteria for supplier selection to design the Fuzzy analytical Hierarchy process tree structure. In phase two, Fuzzy analytical Hierarchy process (FAHP) is used in determining of weights of main criteria sub-criteria and alternatives. They have successfully applied FAHP to a supplier selection problem in a gear manufacturing company. NTM (Non Traditional Manufacturing) processes consume high power and are too expensive which necessitates one optimum NTM process for a given material for economical machining. However, the selection of optimum process is a tedious task. It consists of various factors that influence the selection method of machining process. Chauhan et. al [14] in their paper have proposed a combined method using the Technique for Order Preference by Similarity to Ideal Solution(TOPSIS) and an "Analytical Hierarchy Process" (AHP) to select the most suitable non-traditional process for given material and shape feature combination while taking various attributes affecting the NTM selection determination. Evaluation and selection of a machine tool is a complex decision-making problem involving multiple conflicting criteria. A research by Athawale [15] presents a logical procedure to evaluate the CNC machines in terms of system specifications and cost by using TOPSIS method. Due to highly competitive global market, the organizations are now forced to focus more on increasing productivity while decreasing cost by the right selection of machine tools. Proper selection of machine tool justifies labor saving, improved product quality and increased production rate with enhanced overall productivity.

Authors have developed different models for the selection of manufacturing method based on multi objectives [16-17]. The authors have designed and implemented a tool to enable the end user a quick selection of appropriate manufacturing method based on multiple objectives weighted differently. A tool provides two different types of interfaces to an end user. One interface is GUI based which is user friendly and provides a simple drag and drop operation for the selection of manufacturing method based on multiple objectives. The second method is command-line interface enabling the end user to query the database using human like query language termed as Manufacturing Query Language (MQL) designed by the authors. In most of the organizations data is stored in a persistent storage medium such as database which demands the knowledge of 'Structured Query Language' (SQL) for information retrieval, which involves complex constructs for sorting, filtering and joining data from multiple tables. This poses a great challenge to the manager for extracting useful information from the database. To address this issue, the authors have developed a grammar for human interface to database which involves execution of human like queries for information retrieval from the database based on multiple objectives. Human like queries can be constructed based on this grammar which is parsed using finite state automata. The state table and state graphs are developed for different queries. State information is stored in a database as a measure towards improving efficiency. These queries in turn are evaluated by mapping to the corresponding SQL commands internally. Hence human like query is a layer on top of SQL to render the end user free from intricacies involved in SQL. To parse MQL queries, the authors have designed and implemented an NLP parser by defining a finite set of symbols, words and language rules to constitute a MQL grammar. The parse tree is constructed based on the grammar specified. The NLP query is parsed using NLP parser and the queries which are successfully parsed are evaluated by mapping them to the corresponding prolog query using Java interface to Prolog (JPL).

III. Generic Model for Selection of Manufacturing Method based on one or more objectives and/or one or more functions

A. Method Selection Matrix

The possible manufacturing methods suitable for a specific objective are organized in a matrix of order 11x10 in a method selection matrix. The method selection matrix has an element 1 if a particular manufacturing method is suitable for that objective, irrespective of its grade, otherwise the corresponding entry has a value 0.

Let M(k) represent a method selection matrix of order 11x10 for k^{th} objective where the ijth element of M(k) is given by

$m_{ij}^{(k)} = 1$ if $(i+j)^{th}$ method has a grade 'a' or 'b' for the objective k, where 1 <= k <=16 or a function f where f є [1.1-1.6,2.1-2.6,3.1-3.6,4.1-4.6]

in the case of single objective and/or single function criteria,

$= g * c * o$ if the $(i+j)^{th}$ method is applicable to objective k in the case of multi objective criteria and

$= g * c * f$ if the $(i+j)^{th}$ method is applicable to function k in the case of multi objective and multi function criteria.

In the above expressions g, c, o and f refer to grade weight, class weight, objective weight and function weight, respectively.

We define the union operation on the method selection matrices M and M' as follows.

Let $M = [m_{ij}]$ and $M' = [m'_{ij}]$ represent two method selection matrices.

$M \cup M' = m_{ij} + m'_{ij}$ iff mij<>0 and m'ij<> 0

$= 0,$ otherwise.

With this type of representation, where various methods available for each objective and function are represented in a matrix form, it becomes plausible to address various cases pertaining to the manufacturing method selection for

- Single Objective
- Single Function
- Single Objective and Single function
- MultiObjective
- MultiFunction
- MultiObjective and MultiFunction

Without such a representation, the number of various permutations and combinations to be considered in various cases are depicted Table 1.

Table 1: No of Permutations for Selection of Manufacturing Method in various Scenarios

Selection Type	No. of Permutations
Single Objective	16
Single Function	24
Single Objective and Single Function	384
MultiObjective	${}^{16}C_1 + {}^{16}C_2 + {}^{16}C_3 + \ldots + {}^{16}C_{16} = 2^{16}-1 = 65,536$
MultiFunction	${}^{24}C_1 + {}^{24}C_2 + {}^{24}C_3 + \ldots + {}^{24}C_{24} = 2^{24}-1 =$ 1,67,77,216
MultiObjective and MultiFunction	$(2^{16}-1) * (2^{24}-1) = 10,99,51,16,27,776$

Hence no of permutations for multi objective, multi-function criteria boils down to 10 trillion.

Design Framework

3-Tier Architecture for Model Implementation

The three-tier architecture is employed for implementation of the model where the presentation tier with which the end user interacts is implemented in VB, the business logic in the middle tier is implemented in Java and domain information is stored in MS-Access in a data tier as depicted in Fig. 1.

Presentation Tier (VB)
Business Logic (Java)
Data Tier (MS-Access)

Fig. 1: 3-Tier Architecture for Model implementaion.

The various components in different layers and a detailed work-flow of the model in depicted in Fig. 2 and the class diagram for the implementation of business logic in Java is shown in Fig. 3.

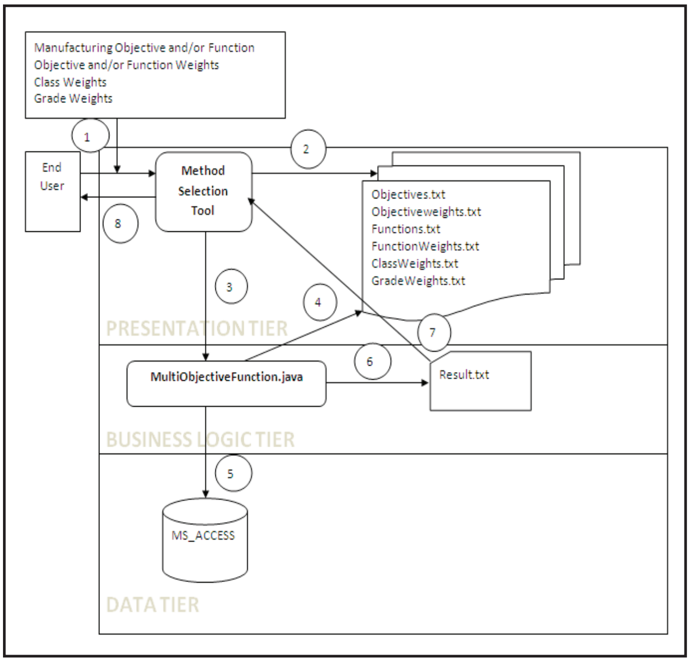
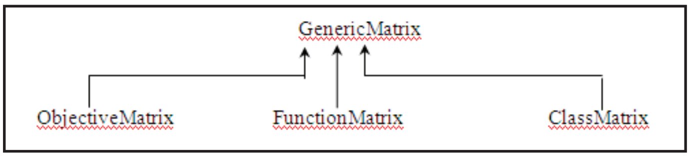


Fig. 2: Work-Flow of the Model

Class Diagram for Implementation of Business Logic

GenericMatrix
name : String grade : int[][]
GenericMatrix(name:String, grades:int[][]) toString(void):String
Union(GenericMatrixgm):GenericMatrix findMethod(om:ObjectiveMatrix):Method



ObjectiveMatrix, FunctionMatrix and ClassMatrix are simply the symbolic names for Generic Matrix having the same structure as GenericMatrix.

Method
methodNumber : int methodName : String classificationCode : String value : int
Method(int,String,String,int):void toString(void):String

MultiObjectiveFunction
grade : int[][]
main(String[]):void createObjectiveGradeMatrix():void createFunctionGradeMatrix():void findMethods(type:String):String[]

Fig. 3: Class Diagram for Implementation of Business Logic in Java

Algorithm in C-Style

Algorithm for Single Objective

/*Any high level language interfacing with back end database management system provides high level API for primitive database functions such as creating a connection object, checking the current position of the resultset pointer and selecting a set of rows based on the given criterion. Hence this algorithm assumes some standard functions as shown below:

Standard Functions used in the Algorithm

getConnection - is a built-in function returning Connection object for a given connection string selectRows() - is a built-in function retrieving matching rows for given column names and a table name.

isEOF(ResultSet) - is a function which returns a boolean value, indicating whether the resultset pointer is at the beginning of or at the end of resultset.

*/

```
/*   conString : Connection String for connecting to a back end.
*/

/*   Input : objective number
     Output : List of methods conforming to input objective
 */

/*  Variable Declaration Block */
int method,row,col;
int onum;
int[] selectedMethods;
ObjectiveMatrix om;
ObjectiveMatrix cm;
int count;
int grades[11][10];
int cgrades[11][10];

/* Variable Initialization Block */
classM=0, classP=0, classS=0, classT=0, classX=0;
for(int i=1;i<argc;i++)
    {
charch=argv[i][0];
switch(ch)
     {
case 'M':
            classM=1;
            break;
case 'P':
            classP=1;
            break;
case 'S':
            classS=1;
            break;
case 'T':
            classT=1;
            break;
case 'X':
            classX=1;
            break;
     }
```

```
  }
for (int i=0;i<11;i++)
  {
      for(int j=0;j<10;j++)
      {
        grades[i][j]=0;
        cgrades[i][j]=0;
      }
  }
con=getConnection(conString);
query={ "select methodnumber from objectivegrading where objectivenumber=" + onum +" and (grade='a' or grade='b') order by methodnumber";}; Q1
resultSet=getResultSet(con,query);
while((isEOF(resultSet) == false)
  {
      methodno=resultSet("methodnumber");
      query2={select classificationcode from method where methodnumber = "+methodno;};
      resultSet2=getResultSet(con,query2);
      row=grade/10;
      col=grade%10;
      grades[row][col]=1;
      ccode=resultSet2("classificationcode");
      if (ccode.equals("M"))
            cgrades[row][col]=classM;
      if (ccode.equals("P"))
            cgrades[row][col]=classP;
      if (ccode.equals("S"))
            cgrades[row][col]=classS;
      if (ccode.equals("T"))
            cgrades[row][col]=classT;
      if (ccode.equals("X"))
            cgrades[row][col]=classX;
  }

om=new ObjectiveMatrix(objectivename,grades);
cm=new ObjectiveMatrix("classes",cgrades);
ObjectiveMatrix result=om.union(cm);
for (int i=0;i<12;i++)
  {
      for(int j=0;j<10;j++)
      {
        if (result.grades[i][j]==1)
            selectedMethods[count++]=10*i+j;
      }
  }
```

Modification of the Algorithm for Selection of Manufacturing Methods Based on Single Function

The only change required in the above algorithm is replacing the query Q1 on ObjectiveGrading table with the query on FunctionGrading table as given below:

```
query={select methodnumber from functiongrading where functionnumber=" + fnum +" and (grade='a' or grade='b') order by methodnumber};
```

Modification of the Algorithm for Selection of Manufacturing Methods Based on Single Objective and Single Function

Replace a single query with two queries being fired on ObjectiveGrading and FunctionGrading tables as shown below to construct two method selection matrices represented programmatically by an instance of ObjectiveMatrix and FunctionMatrix classes. Two more method selection matrices are constituted taking into account the class weight of the corresponding methods. The union operation is then performed on the four matrices to form the resultant matrix.

```
query1={select methodnumber from objectivegrading where objectivenumber=" + onum +" and (grade='a' or grade='b') order by methodnumber};

query2={select methodnumber from functiongrading where functionnumber=" + fnum +" and (grade='a' or grade='b') order by methodnumber};

ObjectiveMatrix result=om.union(cm);
result=result.union(fm);
result=result.union(cfm);
```

Algorithm for Selection of Manufacturing Methods based on Multiple Objectives.

In this case a business component is dynamically generated based on the objectives input by the user. The business component is divided into 3 parts of which the first and the third parts are static while the second part is framed dynamically at runtime based on the user input which is then concatenated with its static counterparts as shown in Fig. 4.

Static Component 1	Dynamic Component 2	Static Component 3

Fig. 4: Structure of Java Business Component

The algorithm employed is shown below:

```
ObjectiveMatrix[] om=new ObjectiveMatrix[16];
for(int onum=0;onum<selected_objectives.length;onum++)
{
      query1={select methodnumber from objectivegrading where objectivenumber=" +      selected_objecives[onum] + " order by methodnumber"};
      om[onum]=new ObjectiveMatrix(objectivename,methods);
      result=om[selected_objectives[0]];
      for (int onum=1;onum<selected_objectives.length;onum++)
      {
        result = result.union(om[selected_objectives[i]]);
      }
}
```

Algorithm for Selection of Manufacturing Methods based on Multiple Objectives and Multiple Functions.

An additional logic to take care of multiple functions is appended as shown below:

```
/* Function Grading Starts */
int ipart,fpart;
float func;
int fnum1;
String functionname;
for(int fnum=0;fnum<functions.length;fnum++)
```

```
{
  fnum1=frequired[fnum];
  ipart=(fnum1/6)+1;
  fpart=fnum1 % 6;
  if (fpart==0)
  {
        fpart=6;
        ipart--;
  }
func=ipart+((float)fpart/10);
query={select methodnumber,grade from functiongrading where functionnumber=" + func +" order by methodnumber};
query1={select functionname from function,functiongrading where function.functionnumber = functiongrading.functionnumber and functiongrading.functionnumber="+func};
resultSet =getResultSet(query,con);
resultSet1=getResultSet(query1,con);
if( isEOF(resultSet1)==false)
        functionname=rs1.getString(1);
else
        functionname="";
int methods[][]=new int[12][10];
for(int i=0;i<12;i++)
        grades[i]=new int[12];

for (int i=0;i<12;i++)
{
        for(int j=0;j<10;j++)
        {
           methods[i][j]=0;
        }
}
char clas;
float grade1;
while (isEOF(resultSet)==false)
{
methodno=resultSet("methodnumber");
query2={select classificationcode from method,functiongrading where method.methodnumber
        =functiongrading.methodnumber and functiongrading.methodnumber="+methodno;

resultSet2 =getResultSet(query2,con);
String class1=resultSet2("classificationcode");
if (class1.equals("M"))
        classwt=classweights[0];
if (class1.equals("P"))
        classwt=classweights[1];
if (class1.equals("S"))
        classwt=classweights[2];
if (class1.equals("T"))
        classwt=classweights[3];
if (class1.equals("X"))
        classwt=classweights[4];
row=grade/10;
col=grade%10;
clas=resultSet("grade");
methods[row][col]=gradeweights[clas-97]*functionweights[fnum1]*classwt;
}
fm[fnum1]=new ObjectiveMatrix(functionname,grades);
}
/* Function Grading Ends */
result=om[selected_objectives[0]];
for (int onum=1;onum<selected_objectives.length;onum++)
{
  result = result.union(om[selected_objectives[i]]);
}

for (int i=0;i<functions.length;i++)
{
        f=functions[i];
        ipart1=(int)f;
        fpart1=(int)((f*10)%10);
        fnew=6*(ipart1-1)+fpart1;
        result=result.union(fm[fnew]);
}
```

Application of the Model for Multi Objective Criteria

The model developed above is applied for the selection of manufacturing method based on multi objectives.

Problem Definition

Select a set of manufacturing methods for a hypothetical organization conforming to the following organizational objectives.
Objective 2 - Reduce Production costs
Objective 3 - Rapid response to market demands - product design
Objective 6 – Progress towards zero inventory - increase inventory turnaround
Objective 7 – Improve management knowledge and information - enterprise communication
Objective 13 - Improve enterprise integration - improving supply chain globally,
for the weights assigned to different objectives, classes and grades are depicted in Tables 2-4.

Objective	Weight
2	10
3	8
6	8
7	6
13	6

Table 2. Objective Weights

Class	Weight
M	4
P	3
S	5
T	5
X	1

Table 3. Class Weights

Grade	Weight
a	6
b	4
c	3
d	1

Table 4. Grade Weights

Solution

The method selection matrix is formulated for each of the objectives taking into account, corresponding objective weights, weight of the class to which the method belongs and objective's grade weight as given in Table 2-4.
The method selection matrices for objectives 2 and 3 are depicted in Fig. 4 and similar matrices exist for other objectives.

Note : Matrix starts with zero index

Method Numbers for Objective 2					
1	22	43	57	77	94
3	24	44	58	79	95
6	26	46	60	82	97
8	27	47	62	84	98
9	30	48	67	86	99
13	33	50	68	87	101
14	34	51	69	88	103
16	37	52	70	91	105
19	38	53	73	92	107
20	39	56	76	93	

0	150	0	120	0	0	120	0	90	120
0	0	0	120	40	0	90	0	0	150
150	0	30	0	150	0	160	90	0	0
90	0	0	50	150	0	0	30	10	200
0	0	0	120	120	0	90	30	90	0
120	160	90	120	0	0	160	90	120	0
40	0	120	0	0	0	0	90	200	120
120	0	0	90	0	0	30	120	0	120
0	0	50	0	150	0	90	30	90	0
0	40	150	120	120	150	0	30	40	40
0	160	0	120	0	30	0	90	0	0
0	0	0	0	0	0	0	0	0	0

Method Numbers for Objective 3						
2	21	38	50	67	83	98
3	23	39	51	68	85	101
4	25	42	52	69	86	102
7	27	43	53	70	88	103
8	29	44	54	73	90	104
9	31	45	57	75	91	105
12	32	46	58	77	92	106
14	34	47	61	79	93	109
15	35	48	62	80	94	110
20	37	49	65	82	95	

0	0	32	96	24	0	0	160	24	128
0	0	160	0	24	40	0	0	0	0
40	160	0	72	0	24	0	72	0	24
0	160	240	0	120	144	0	24	24	160
0	0	24	128	96	240	24	24	24	32
72	128	24	96	128	0	0	96	32	0
0	24	128	0	0	72	0	72	120	128
128	0	0	24	0	32	0	128	0	96
24	0	160	96	0	96	24	0	24	0
160	24	40	32	128	160	0	0	128	0
0	128	160	128	160	96	160	0	0	96
72	0	0	0	0	0	0	0	0	0

Fig. 4: Method Selection Matrices for Objectives 2 and 3.

Fig. 5 depicts the application of union operation on the method selection matrices for objectives 2 and 3.

Objective 2 U Objective 3

Method Numbers for Objective 2 and Objective 3			
3	43	58	86
8	44	62	88
9	46	67	91
14	47	68	93
20	48	69	95
27	50	70	
34	51	73	
37	52	77	
38	53	79	
39	57	82	

0	0	0	216	0	0	0	0	114	248
0	0	0	0	64	0	0	0	0	0
190	0	0	0	0	0	0	162	0	0
0	0	0	0	270	0	0	54	34	360
0	0	0	248	216	0	114	54	114	0
192	288	114	216	0	0	0	186	152	0
0	0	248	0	0	0	0	162	320	248
248	0	0	114	0	0	0	248	0	216
0	0	210	0	0	0	114	0	114	0
0	64	190	152	248	310	0	0	168	0
0	288	0	248	0	126	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Fig. 5: Application of Union Operation on the Method Selection Matrices for Objectives 2 and 3

Method Numbers for Objective 2 and Objective 3 and Objective 6 and Objective 7 and Objective 13			
9	98		
14			
20			
37			
38			
39			
43			
50			
68			
94			

0	0	0	0	0	0	0	0	0	544
0	0	0	0	138	0	0	0	0	0
470	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	134	100	760
0	0	0	544	0	0	0	0	0	0
390	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	690	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	544	0	0	0	440	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

IV. Results and Analysis

The generic model presented above is implemented in Java in a business tier with MS-Access as back end for storing the domain specific information. The structure of the database is shown in Fig. 7. The Graphical User Interface is developed in VB which invokes the business logic for generating the required output. The output is routed to a text file which is retrieved and displayed by the presentation tier logic. The graphical user interface of presentation tier is depicted in Fig. 8.

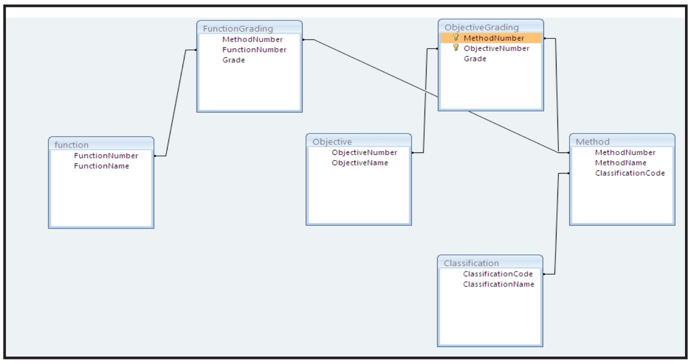


Fig. 7: Structure of Manufacturing Database

Fig. 8: Graphical User Interface for Presentation Tier

Figs. 10, 11 and 12 depict the selection of manufacturing method based on a single objective for all classes, class M and class S, respectively.

Fig. 10: Selection of Manufacturing Methods for a Single Objective for all classes

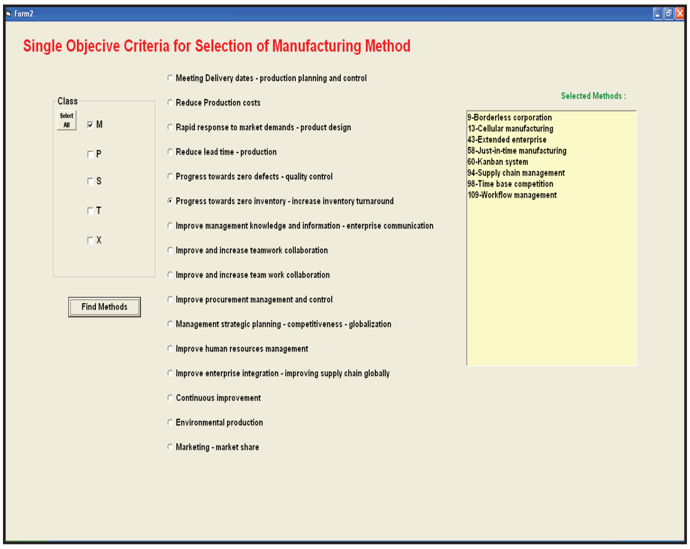


Fig. 11: Selection of Manufacturing Methods for a Single Objective for Class M

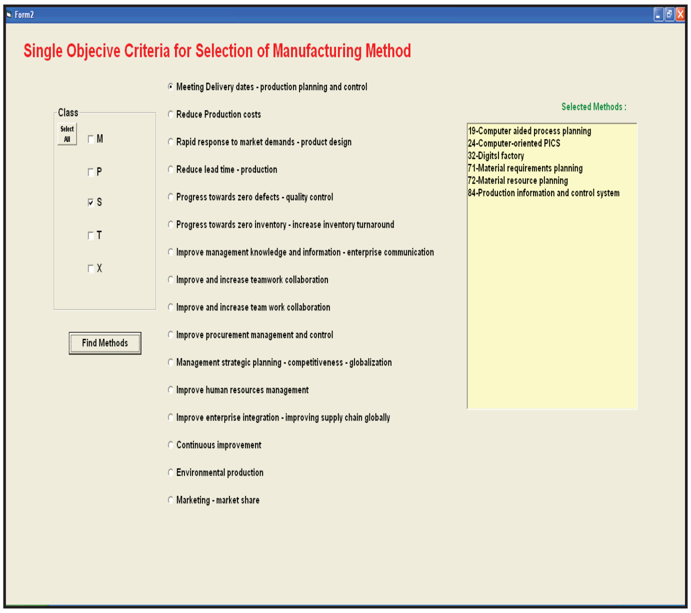


Fig. 12: Selection of Manufacturing Methods for a Single Objective for Class S

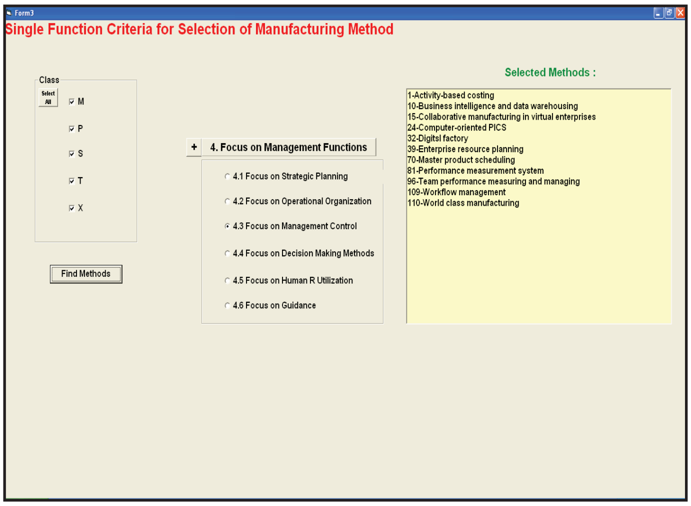


Fig. 13: (a). Selection of Manufacturing Methods for a Single Function 4.3

Fig. 13 (a) depicts the selection of manufacturing methods for the function 4.3, Focus on Management Control with Fig. 13 (b) depicting the execution of business logic in middle tier.

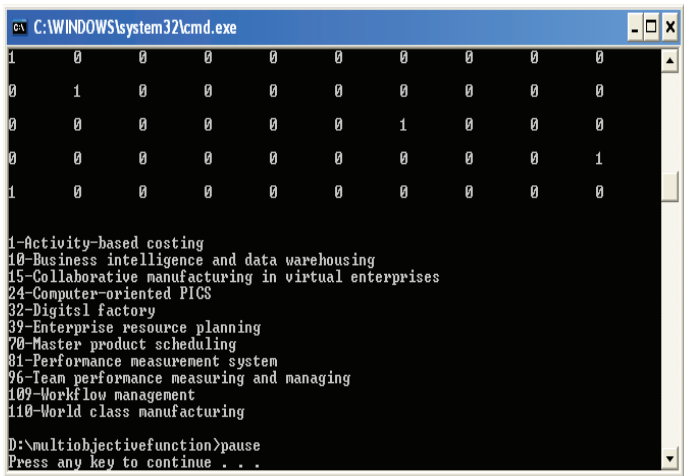


Fig. 13: (b). Execution of Business Logic in Middle Tier

Fig. 14 (a) and 15 (a) depict the selection of manufacturing methods based on multi objectives and multi objectives and multi functions, respectively. Figures 14 (b) and 15 (b) show the execution of business logic in the respective cases.

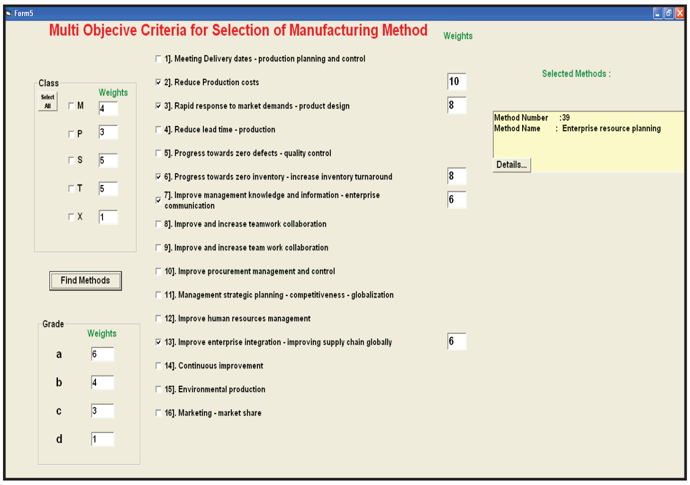


Fig. 14: (a). Selection of Manufacturing Methods Based on Multi Objectives

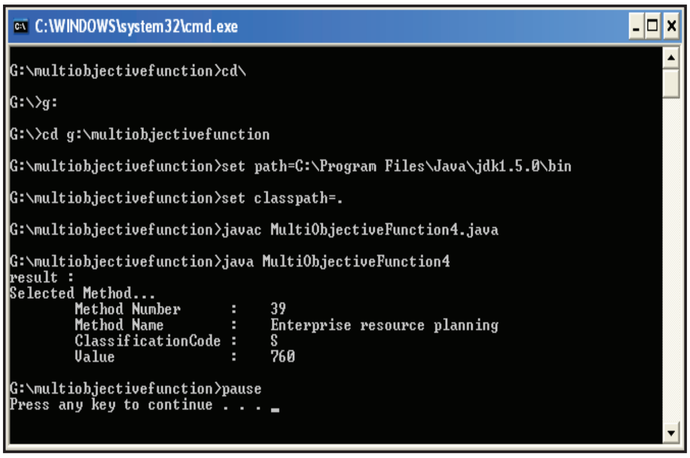


Fig. 14: (b). Execution of Business Logic in Multi objective Method Selection

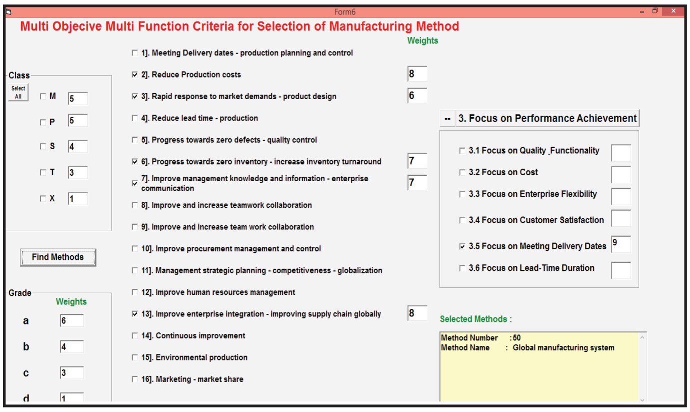


Fig. 15: (a). Selection of Manufacturing Methods Based on Multi Objectives and Multi Functions

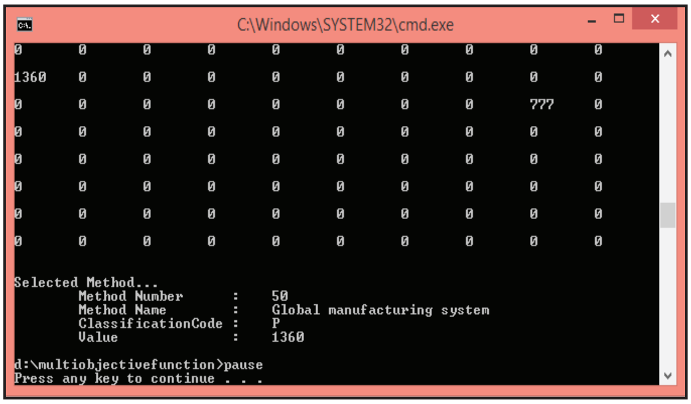


Fig. 15: (b). Execution of Business Logic in Multi objective and Multi Function Method Selection

V. Conclusion and Scope for Future Work

Globalization and dynamic nature of business environment demands a quick decision making on the part of the manager for the selection of a suitable manufacturing method based on organizational objectives and functions. In order to improve a manufacturing cycle more than 110 manufacturing methods have been proposed and this set is constantly increasing. This poses a great challenge to the manager for the selection of most effective and economical manufacturing process under dynamically changing business scenario. Different models exist for the selection of different objectives and functions. No single model can address different issues. In this paper, the authors have designed and developed a model for selection of manufacturing method based on multiple objectives and/or multiple functions. A set of selected methods for each objective/function is represented in a matrix format and incorporating a new objective and/or function in an existing matrix simply evaluates to a union operation on a set of matrices. Without such a representation, the number of options grow exponentially each time a new method is added to a set of available manufacturing methods. The model is implemented in a 3-tier architecture with VB in presentation tier, Java in business logic tier and MS-Access for storing domain specific information.

Our future work focuses on developing a supervised neural network which will be trained for selection of an appropriate manufacturing method. Further, the hybrid models such as neuro-fuzzy enable the close examination of the methods so that the methods which are ruled out in the current model and are close to the selected methods will be reconsidered based on the infrastructure available and taking into account the additional cost incurred for shifting from the current to the new manufacturing method.

VIII. References

[1] Gideon Halevi, Handbook of Production Management Methods, Butterworth Heinemann publications, ISBN 0 7506 5088 5.

[2] L. Mikhailov, M. G. Singh,"Fuzzy analytic network process and its application to the development of decision support systems," IEEE Transactions on Systems, Man, and Cybernetics, Part C. Applications and Reviews, Vol. 33, No. 1, pp. 33–41, 2003.

[3] R. Santhanam and G. J. Kyparisis, "A multiple criteria decision model for information system project selection," Computers & Operations Research, Vol. 22, No. 8, pp. 807–818, 1995.

[4] V. S. Lai, K. W. Bo, and W. Cheung, “Group decision making in a multiple criteria environment: A case using the AHP in software selection,” European Journal of Op-erational Research, Vol. 137, No. 1, pp. 34–144, 2002.
[5] C. C. Wei, C. F. Chien, and M. J. J. Wang, “An AHP- based approach to ERP system selection,” International Journal of Production Economics, Vol. 96, No. 1, pp. 47–62, 2005.
[6] J. P. Brans, B. Mareschal, and P. Vincke, “PROMETHEE: A new family of outranking methods in multicriteria analysis,” Operational Research, Vol. 3, pp. 477–490. 1984.
[7] R. V. Rao, “Decision making in the manufacturing environment using graph theory and fuzzy multiple attribute decision making methods,” Springer-Verlag, London, 2007.
[8] R. Santhanam, G. J. Kyparisis, “A multiple criteria decision model for information system project selection,” Computers & Operations Research, Vol. 22, No. 8, pp. 807–818, 1995.
[9] Dhananjay R. Kalbandeand G.T.Thampi, Multi-attribute and Multi-criteria Decision MakingModel for technology selection using fuzzy logic, International Journal of Computing Science and Communication Technologies, Vol. 2, No. 1, July 2009.
[10] Journal of Micromechanics and Microengineering, Xuan F Zha and H Du, Manufacturing process and materialselection in concurrent collaborativedesign of MEMS devices, 13, pp. 509–522, 2003.
[11] Chenhui Shaoa, Kamran Paynabarb, Tae HyungKima, Jionghua (Judy) Jinc, S. Jack Hua, J. Patrick Spicerd, HuiWangd, Jeffrey A. Abelld, Feature selection for manufacturing process monitoring using cross-validation, Journal of Manufacturing Systems, Volume 32, Issue 4, October 2013, pp. 550–555
[12] R. V. Rao, T. S. Rajesh, Software Selection in Manufacturing Industries Using a Fuzzy Multiple Criteria Decision Making Method, PROMETHEE, Intelligent Information Management, 2009, 1, 159-165, December 2009
[13] Mohsin Khan, Arvind Jayant , Veepan kumar, Multi-Criteria Supplier Selection Using Fuzzy-AHP Approach: A Case Study of Manufacturing Company, IJRMET Vol. 5, Issue 1, spl - 1 November 2014 - April 2015, pp 73-79.
[14] Ashish Chauhan, M. K. Pradhan, SELECTION OF NON-CONVENTIONAL MANUFACTURING PROCESS: A COMBINE TOPSIS-AHP APPROACH, 5th International & 26th All India Manufacturing Technology, Design and Research Conference (AIMTDR 2014) December 12th–14th, 2014, IIT Guwahati, Assam, India
[15] Vijay Manikrao Athawale and Shankar Chakraborty,"A TOPSIS Method-based Approach to Machine Tool Selection", Proceedings of the 2010 International Conference on Industrial Engineering and Operations Management Dhaka, Bangladesh, January 9 – 10, 2010
[16] Girish R. Naik, V.A.Raikar and Poornima G. Naik, Multi Objective Criteria for Selection of Manufacturing Method, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 7, July 2014. ISSN:2277 128X.
[17] Girish R. Naik, V.A.Raikar and Poornima G. Naik, Multi Objective Criteria for Selection of Manufacturing Method using NLP Parser, International Journal on Recent and Innovation Trends in Computing and Communication, Vol 2, Issue 11, Nov 2014, pp. 3484 – 3493.