Particle swarm optimisation in development of component families using classification and coding system: a case study in an Indian manufacturing firm

Tamal Ghosh* and Pranab K. Dan

Department of Industrial Engineering and Management,
West Bengal University of Technology,
BF-142, Salt Lake City, Kolkata 700064, India
Fax: +91-33-2334-1030
E-mail: tamal.31@gmail.com
E-mail: danpk.wbut@gmail.com
*Corresponding author

Abstract: Component/part family identification is an NP class problem in the extent of group technology (GT). In preceding literature it has been evidenced that part family identification techniques are ordinarily grounded on production flow analysis which typically studies operational requirements, sequences and time required. Recently, various soft-computing-based techniques are heavily attempted to address such problems. However in designing of parts, process planning, these methods are not convenient. To accomplish such issues coding and classification-based techniques are believed to be extremely proficient. This article portrays a minimal and competent nature inspired heuristic approach based on particle swarm optimisation (PSO) to acquire effective component/part families; exploiting part coding scheme and the technique is verified on top of test data as well as industrial data. The simulation outcomes are assessed with the results achieved using simple heuristic clustering method. The experimental results recommend that the proposed method is more effective in terms of computational efficiency and has outperformed the heuristic technique with enhanced solution quality.

Keywords: component family identification; part family formation; group technology; GT; heuristics; particle swarm optimisation; PSO; soft-computing; part coding analysis; India.

Reference to this paper should be made as follows: Ghosh, T. and Dan, P.K. (xxxx) 'Particle swarm optimisation in development of component families using classification and coding system: a case study in an Indian manufacturing firm', *Int. J. Services and Operations Management*, Vol. X, No. Y, pp.000–000.

Biographical notes: Tamal Ghosh is an MTech from the Department of Industrial Engineering and Management, West Bengal University of Technology, India. He has also obtained his Bachelor of Technology in Computer Science and Engineering from National Institute of Technology Calicut, India. His current research area is the application of soft computing techniques in operational management. He has number of articles published in international journals and conferences.

Pranab K. Dan is a reader in the Department of Industrial Engineering and Management, West Bengal University of Technology, India. He obtained his PhD in Production Engineering in 1996 from Jadavpur University, India. He

Copyright © 200x Inderscience Enterprises Ltd.

obtained his Bachelor of Engineering and Master of Engineering in Mechanical Engineering from Bengal Engineering College, Shibpore affiliated to Calcutta University, India in the year 1980 and 1982, respectively. His research and professional experience is in the area of industrial engineering. He has a number of articles published in international journals and conferences.

1 Introduction

Group technology (GT) is a manufacturing philosophy in the area of cellular manufacturing systems (CMS), which is significant in improving productivity for the manufacturing businesses. GT is a technique which discriminates similar parts and groups them into part families based on their manufacturing designs, attributes and geometric shapes (Burbidge, 1963). GT examines products, parts and assemblies and accumulates homogeneous items in subclasses to simplify design, manufacturing, purchasing and other business processes. GT aids design and manufacturing tasks in several ways. It minimises the time required for practising engineering drawings for homogeneous parts, also minimises the cost and time required for designing supplementary machining apparatus such as typically designed cutting tools, jigs and fixtures and so on. As reported in the literature, a successful implementation of GT can eventually shrink the engineering costs, facilitate cellular manufacturing, quicken product development, enhance costing accuracy, simplify process planning, minimise tooling cost and simplify the overall purchasing process (Galan et al., 2007; Guerrero et al., 2000). A major prerequisite in implementing GT is the identification of part families (Wemmerlov and Hyer, 1987; Kaparthi and Suresh, 1991). A part family is a group of parts sharing homogeneous design and manufacturing attributes. Early research in this domain has been dedicated primarily on the formation of production-oriented part families in which similarities amongst parts are principally recognised on the fact of processing requirements, operation time and operation sequences. However these methodologies are inadequate in achieving the needs of other extents of manufacturing. For example, parts with homogeneous shape, size, dimension or other design characteristics are believed to be clustered in a single family for design justification and elimination of part varieties; however parts which are clustered on the fact of homogeneous routing and tooling needs are convenient to resolve the process planning issues.

Therefore, the scope of this domain of investigation is believed to be expanded and examined to a wider span of part similarities. Part similarities are believed to be identified sooner than the formation of part families. Part attributes such as shape, length/diameter ratio, material type, part function, dimensions, tolerances, surface finishing, process, operations, machine tool, operation sequence, annual production quantity, fixtures needed, lot sizes have been considered as the basis for similarity utilisation (Groover and Zimmers, 1984). The complexity remains in acquiring an appropriate technique which provides an identifying competence of human being, such as identifying patterns in groups, and forming part families with the aid of intelligence (Moon, 1992).

To explain the part family formation problems, Opitz coding and classification system is considered in this paper, which was initially developed by Opitz (1970) at Aachen Technology University, West Germany. The basic code comprises of nine digits that can be extended by additional four letters. The first five digits are called the form code and designate the design or the general form of the part and hence aid in design retrieval. Later, four more digits were added to the coding scheme in order to enhance the manufacturing information of the specific work part. These last four digits are also called special code. All four digits are integers, and respectively represent: dimensions, material, original shape of raw stock, and accuracy of the work part (The Opitz coding schematic is given in Appendix).

The interpretations of first nine digits are:

Digit 1 General shape of work-piece, otherwise called ‘part-class’: This is further subdivided into rotational and non-rotational classes and further divided by size (length/diameter or length/width ratio).

Digit 2 External shapes and relevant form: Features are recognised as stepped, conical, straight contours. Threads and grooves are also important.

Digit 3 Internal shapes: Features are solid, bored and straight or bored in stepped diameter. Threads and grooves are integral part.

Digit 4 Surface plane machining, such as internal or external curved surfaces, slots, splines.

Digit 5 Auxiliary holes and gear teeth.

Digit 6 Diameter or length of work-piece.

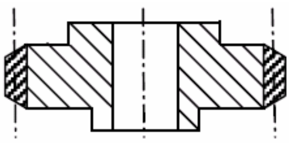
Digit 7 Material used.

Digit 8 Shape of raw materials, such as round bar, sheet metal, casting, tubing, etc.

Digit 9 Work-piece accuracy.

All the nine digits are interpreted numerically (0 to 9). An example of spur gear is shown in Figure 1 in this context.

Figure 1 Spur gear



The Opitz code of spur gear could be 04166 2409. Its attributes are denoted as a1 to a9 which indicate,

a1 0 (rotational component with L/D <= 2)

a2 4 (stepped both ends, no shape element)

a3 1 (stepped to one end, no shape element)

a4 6 (internal plane surface and/or slot)

a5 6 (with gear teeth, auxiliary hole is axial and/or radial with PCD)

a6 2 (400 mm. < diameter or length of edge < = 600 mm) a7 4 (non-ferrous material)

a8 0 (round bar)

a9 9 [(2+ 3+) +4 +5 form code requiring more accuracy].

The part family formation problem stated in this research can be formulated using a part-attribute incidence matrix $B = [b_{ij}]$, of size $m \times n$, where m is the number of parts and n is the number of attributes of that part. b_{ij} represents the coding value (0 to 9) of j^{th} attribute of i^{th} part. A 55 × 9 test problem based on Opitz coding system is generated for experimental purpose (Figure 2).

Figure 2 (55 × 9) dataset

	a1	a2	a3	a4	a5	a6	a7	a8	a9
p1	3	5	6	2	2	1	0	3	0
p2	6	6	8	5	4	2	5	0	0
p3	1	9	5	1	3	0	4	3	0
p4	0	6	6	2	1	0	4	3	0
p5	0	8	9	9	6	0	6	0	0
p6	0	8	3	9	6	0	6	0	0
p7	1	2	0	9	0	0	6	0	0
p8	0	9	5	8	4	1	5	2	0
p9	2	2	0	9	0	0	0	0	0
p10	0	6	6	0	1	1	0	7	0
p11	1	2	0	9	0	0	6	0	0
p12	2	2	0	3	0	0	6	0	0
p13	1	3	6	5	2	0	5	0	0
p14	0	2	0	9	0	0	5	0	0
p15	0	3	2	2	2	0	5	3	0
p16	1	8	3	3	4	0	5	3	0
p17	2	6	8	2	4	0	5	3	0
p18	2	0	3	3	3	0	5	3	0
p19	1	6	0	3	0	0	5	0	0
p20	1	2	0	4	0	0	0	0	0
p21	1	3	0	5	0	0	5	0	0
p22	1	2	0	3	0	0	5	0	0
p23	0	6	6	9	5	0	5	0	0
p24	0	8	2	0	7	1	7	3	0
p25	1	3	3	8	2	1	9	3	0
p26	0	9	5	9	6	1	9	3	0
p27	0	5	4	9	8	1	9	3	0
p28	1	0	3	1	2	0	0	7	0
p29	1	2	0	2	5	0	6	0	0
p30	2	5	0	9	5	0	5	0	0
p31	1	2	3	9	7	1	5	0	0
p32	0	9	6	6	8	1	0	0	0
p33	2	9	6	6	6	0	0	0	0
p34	0	9	6	6	6	1	0	0	0
p35	2	2	0	9	6	0	5	0	0
p36	0	2	0	9	0	0	5	0	0
p37	1	4	6	9	1	0	5	0	0
p38	1	4	6	3	1	1	0	0	0
p39	0	0	4	3	1	0	5	0	0
p40	2	6	6	1	4	0	5	3	0
p41	3	2	0	2	0	0	0	0	0
p42	1	4	6	0	1	0	5	3	0
p43	1	4	5	8	2	0	5	3	0
p44	0	4	5	4	2	0	5	0	0
p45	2	0	0	2	0	0	0	0	0
p46	1	5	6	8	5	0	5	0	0
p47	2	0	0	2	0	0	5	0	0
p48	1	2	3	9	0	0	5	3	0
p49	1	0	6	1	0	0	5	3	0
p50	1	5	0	9	0	0	5	0	0
p51	2	5	6	1	2	0	9	7	0
p52	1	2	6	9	2	0	9	3	0
p53	2	0	5	1	2	0	5	3	0
p54	0	4	6	2	2	1	5	7	0
p55	2	5	0	4	0	0	5	0	0

The solution to the problem is to form the families of parts in such a way that the sum of similarities among each pair of parts in a same family would be maximised. Therefore, PSO-based method is proposed in this article which generally maximises the objective function values (sum of similarities) and obtains improved and near optimal solutions.

According to Burbidge (1996), two classical approaches are outlined in past literature in order to form part families, first is production flow analysis (PFA) which deals with processing requirements of parts, operational sequences and operational time of the parts on the machines. These methods are heavily practised in cellular manufacturing system designs (Sangwan and Kodali, 2011; Manimaran et al., 2010; SudhakaraPandian and Mahapatra, 2010; SudhakaraPandian and Mahapatra, 2008). Second approach is the classification and coding (CC) system which utilises predefined coding schemes to facilitate the process using several attributes of parts such as geometrical shapes, materials, design features and functional requirements, etc., (Mitrofanov, 1959).

CC is exposed in this study as an essential and effective tool for successful implementation of GT concept. A code may be numbers (numerical) or alphabets (alphabetical) or a hybridisation of numbers and alphabets (alphanumerical) which are allotted to the parts to process the information (Ham et al., 1985). Parts are categorised based on significant attributes such as dimensions, type of material, tolerance, operations required, basic shapes, surface finishing, etc. In this approach, some typical code is assigned to each part which is a string of numerical digits that stores information about the part. Generally coding systems depict either hierarchical structure (monocode), or chain structure (polycode) or hybrid mode structure mixed with monocode and polycode (Singh and Rajamani, 1996).

Several CC systems have been developed, e.g., Opitz (Opitz, 1970; Opitz and Wiendahl, 1971), MICLASS (TNO, 1975), DCLASS (Gallagher and Knight, 1985) and FORCOD (Jung and Ahluwalia, 1992) which are being exploited heavily in past literature. Part families could be established more realistically by practicing the CC due to the advantage of using the manufacturing and design attributes concurrently (Han and Ham, 1986). Offodile (1992) reported a similarity metric based on the numeric codes for any pair of parts which could be utilised to an appropriate clustering method such as agglomerative clustering algorithm to form efficient part families.

Application of metaheuristics in GT problems is evolving slowly, which mimics the biological phenomena to find ‘fittest’ solution by incorporating ‘survival of the fittest’ theory proposed by Darwin (1929). These techniques have the capabilities to solve the NP-complete problems. These techniques constitute genetic algorithm (GA), simulated annealing (SA), tabu search (TS), particle swarm optimisation (PSO), etc. A detail review of metaheuristics in cellular manufacturing could be obtained from a recent study proposed by Ghosh et al. (2011a).

In the area of GT Lee-Post (2000) first proposed that GT coding system (DCLASS) could be efficiently used with simple GA method to cluster part families which is well suited for part design and process planning in production.

Ghosh et al. (2011b) recently proposed a novel approach based on SA namely SAPFOCS exploiting the part coding analysis technique. Further, Taguchi’s orthogonal design method is used to select the parameters to the proposed algorithm. The technique is therefore tested on five different datasets and compared with traditional clustering technique. The results are extremely effective in terms of the quality of the solutions.

In this paper an objective function is used based on the similarity measure amongst parts and a PSO based approach is demonstrated to investigate the nature of similarities amongst parts by improving the fitness value of the solution. One important fact to be mentioned is, this novel technique (PSO) is being used for the first time to solve the CC systems-based problems.

2 Problem formulation

2.1 Notations

S_{ij} similarity measure between part i and part j

C_2^{Pn} number of pairwise combinations formed in part family n, and P_n is the number of parts in family n

N number of component families

S_{ijk} is similarity measured between part i and part j on attribute k

K is total number of attributes considered

b_{ik} is part coding for part i on attribute k

b_{jk} is part coding for part j on attribute k

R_k is range of possible part codings for all parts on attribute k.

Maximisation of the sum of similarities could be utilised as the evaluation criteria to calculate the fitness of each solution string. This evaluation criterion is expressed mathematically (Lee-post, 2000),

$$\text{Max } f = \sum_{n=1}^{N} S_n \tag{1}$$

where

$$S_n = \frac{\sum i \in n, j \in n S_{ij}}{0.001 + C_2^{P_n}} \tag{2}$$

$$\textit{Perfection Percentage} = \frac{\sum_{n=1}^{N} S_n}{N} \tag{3}$$

Definition of S_{ij} is adopted from Offodile (1992) to accommodate numeric part coding, and is defined as follows:

$$S_{ij} = \frac{\sum_{k=1}^{K} S_{ijk}}{K} \tag{4}$$

where

$$S_{ijk} = 1 - \frac{\left|b_{ik} - b_{jk}\right|}{R_k} \tag{5}$$

3 Research methodology

In order to adopt any metaheuristic approach as a solution methodology, an initial solution should be generated quickly. There could be many techniques available in literature such as similarity coefficient method (McAuley, 1972), rank order clustering method (King, 1980), etc. Although every method has certain time and spatial complexities when coded as some computer programme, and moreover the generated initial solution is not essentially be the near optimal solution. In this paper the initial solution is generated using some random number generator routine to minimise the computational effort. Thereafter the PSO-based metaheuristic approach is applied to improve the quality of that generated solution gradually.

PSO simulates the behaviours of bird flocking (Kennedy and Eberhart, 1995). PSO is used to solve optimisation problems with the concept of particles. All of particles have fitness values which are evaluated by the fitness function to be optimised, and have velocities which determine the flying direction of the particles. PSO is initialised with a group of random particles (solutions) and then performs the searching of near optima by updating generations. Each particle is updated by following two ‘best’ values during every iteration. The first one is the best solution (fitness) it has achieved so far. The fitness value is stored, which is called pbest (local best). Another ‘best’ value that is tracked by the particle swarm optimiser is the best value obtained so far by any particle in the population. This best value is a global best and called gbest. After finding the two best values, the particle updates its velocity and positions with following equations (6) and (7),

$$v_i \leftarrow \omega \times v_i + \varphi_p \times r_p \times \left(p_i - x_i\right) + \varphi_g \times r_g \left(g - x_i\right) \tag{6}$$

$$x_i \leftarrow x_i + v_i \tag{7}$$

3.1 Initial solution representation PSO

In this article an initial solution is represented using a bit string of length *P*, where *P* stands for the number of parts to be clustered. The initial solution for the problem of Figure 2 could be represented as,

‘1212132411235452511234351233542154322431143552124532125’

which means assigning parts 1, 3, 5, 9, 10, 18, 19, 25, 32, 40, 41, 47, 53 are in cell 1, parts 2, 4, 7, 11, 16, 20, 26, 31, 36, 37, 46, 48, 52, 54 are in cell 2, parts 6, 12, 21, 23, 27, 28, 35, 39, 43, 51 are in cell 3, parts 8, 14, 22, 30, 34, 38, 42, 49 are in cell 4 and parts 13, 15, 17, 24, 29, 33, 44, 45, 50, 55 are in cell 5 respectively. This initial solution is generated randomly for both the proposed approaches, which might not be the best or near best solution to the problem. Therefore, to understand the goodness of solution some fitness function is required.

3.2 Fitness evaluation function

The fitness value of each string is a measure of how well the part families are formed. The objective of part family formation is to maximise the sum of similarities of parts. Therefore, equation (1) could be utilised as the evaluation criteria to calculate the fitness of each solution string.

Most important steps in proposed meta-heuristic technique are the evaluation of the obtained solutions. In this step, the goodness (or fitness) of the solution is calculated and based on the result the solution may be deleted, kept, or marked as good. The PSO technique continuously keeps a record of best fitted particle (global best solution) and tries to achieve more improved solutions by updating the velocity and position towards global best solution (Kennedy and Eberhart, 1995). When a new solution is obtained, the goodness (fitness) function is applied, and based on the result, algorithm decides to add the solution to the elite list, or omit the solution and generate another one. During the heuristic and PSO iterations, the goodness of each solution is calculated using equation (1).

4 Results and discussion

4.1 Experiment with simulated data

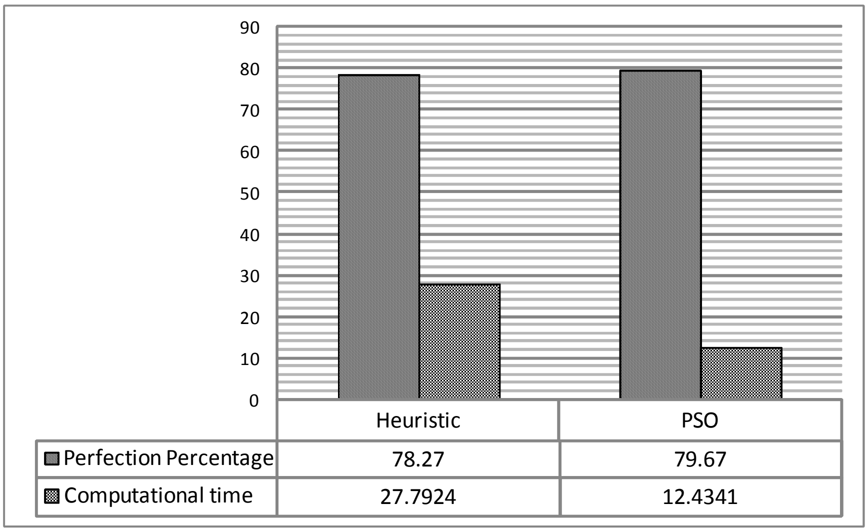
The proposed techniques are tested on the problem datasets of size 55 × 9 shown in Figure 2, which is generated using Opitz coding system (Haworth, 1968). The parts considered for this purpose are presented in Appendix. The proposed algorithm is coded in Matlab 7.1 and executed on Intel Core 2 Duo T6570 laptop computer with 2 GB RAM. The results are compared with a published heuristic technique (Ghosh et al., 2011c) and shown in Table 1.

Table 1 Performance comparison between both the techniques

	Heuristic	*PSO*
Component/ part families (PF)	PF1 (12, 14, 15, 17, 19, 23, 41, 46, 48, 54, 55) PF2 (7, 11, 18, 20, 21, 25, 30, 35, 44) PF 3 (16, 26, 27, 38, 52) PF4 (2, 5, 9, 24, 28, 29, 31, 32, 33, 34, 37, 39, 42, 45, 47, 50, 51) PF5 (1, 3, 4, 6, 8, 10, 13, 22, 36, 40, 43, 49, 53)	PF1 (47, 53) PF2 (6, 13, 29, 35, 36, 44, 46, 49, 51, 52, 54) PF3 (1, 3, 4, 7, 8, 12, 14, 15, 16, 17, 21, 22, 23, 28, 30, 32, 34, 37, 38, 43, 48, 50, 55) PF4 (2, 5, 9, 10, 18, 20, 24, 25, 26, 27, 33, 39, 40, 41, 42) PF5 (11, 19, 31)
Sum of similarities	3.9136	3.9836
Perfection percentage	78.27	79.67
Computational time (CPU sec.)	27.7924	12.4341

Table 1 depicts that the problem solved with PSO approach outperforms the population-based heuristic technique (Ghosh et al., 2011c) in terms of sum of similarity value. The heuristic method is provided in Appendix in detail. The part families obtained by both the approaches are also shown. According to Lee-Post (2000), maximum similarity value would produce better quality of solution, i.e., part families. Therefore, Table 1 demonstrates that part families obtained by PSO are better than the solution produced by heuristic technique. For all the solutions obtained, the perfection percentage is also depicted in Table 1, which authenticates the dominance of PSO over the heuristic algorithm. PSO has depicted a 1.4% improvement for the test problem, is also computationally efficient as it took only the half of the CPU time consumed by the heuristic algorithm. Figure 3 further presents a clear pictorial view of the level of enhancements shown by PSO over heuristic technique. Hence, the proposed PSO approach is established as an effective part grouping method and could be utilised further for more complex and real life problems of GT.

Figure 3 Improvement shown by PSO over heuristic in terms of solution quality and computational time

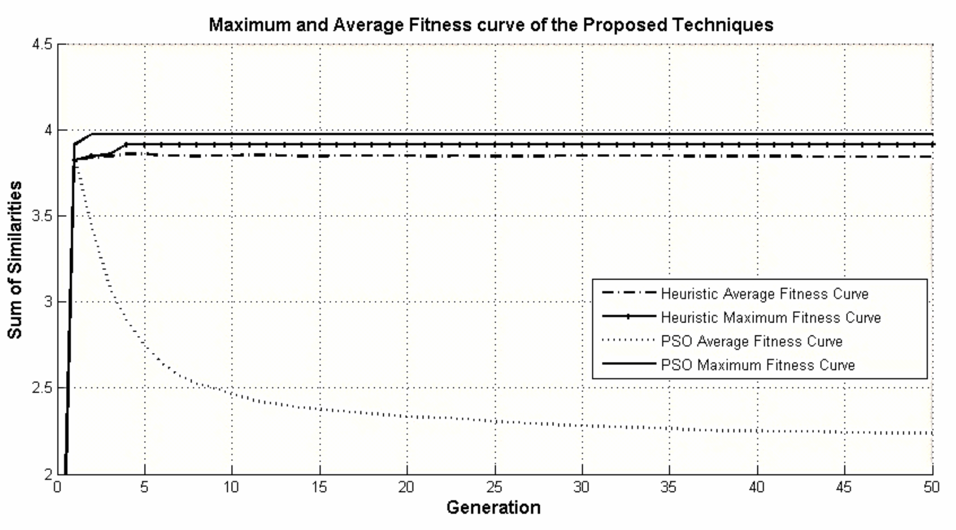


4.2 Convergence analysis

Convergence studies are almost equivalent for both the solution methodologies. Convergence properties during iterations of the heuristic technique and PSO are shown in Figure 4.

For the heuristic approach for the first iteration the objective function *f* attained a value of 3.715. Since the computer programme is designed to maximise the objective function with the iteration counts therefore at 4th iteration it attained the value of 3.9136, an increase of 5.4%. This is the final solution achieved by the heuristic procedure. It took 27.7924 CPU seconds to attain the best objective value. For the PSO method for the first iteration the objective function *f* attained a value of 3.9124 and at 3rd iteration it attained the value of 3.9839, an increase of 1.8%. It took 12.4341 CPU seconds to reach the final iteration. Based on the exhaustive experimentation for the dataset reported in this article, it is observed that the objective value is increased with the iteration counts till it reaches the best objective value at some iteration and thereafter the objective value continues to remain constant even though the number of iterations is increased. Both the stated approaches are executed for 50 iterations.

Figure 4 Convergence analysis curves of both the techniques



4.3 Experiment with industrial data

With the view to make this study more practical and realistic, a survey-based case study had been conducted at Westing House Saxby Farmer Ltd., which is an ISO 9001:2000 company, India's pioneer manufacturer of railway equipment, electro pneumatic regenerative with brake blending and parking brakes of EMU/DMU/MEMU coaches, signalling equipment, semi-permanent automatic coupler. They also execute route relay interlocking projects and all types of civil engineering works. Authors have collected design specifications of each component from its manufacturing units and further a nine digit Opitz coding system was developed and above mentioned techniques were applied so as to compare their results and to form competent part families. A problem dataset of size 50 × 9 was framed using Opitz coding system (Appendix). The detailed specifications of the component along with the names are furnished in Table 2.

The proposed PSO is used to investigate the nature of similarities and to describe the effectiveness of the technique in solving the problem in hand.

In this study an initial solution is represented using a bit string of length *P*, where *P* stands for the number of parts to be clustered. The initial solution for the problem could be represented as,

'1212132411 2354525112 3435123354 21543224311435521245'

The fitness value of each string is a measure of how well the part families are formed. The objective of part family formation is to maximise the sum of similarities of parts. Therefore, maximisation of the sum of similarities could be utilised as the evaluation criteria to calculate the fitness of each solution string. This evaluation criterion is expressed mathematically as,

Table 2 Opitz coding for the 50 parts

1	Cap – 606710891	26	Piston trunk – 643110532
2	Plain pins – 600110502	27	Cap double check valve – 756900802
3	Cap – 623810891	28	Piston double check valve – 600930802
4	Washer – 006150502	29	Piston – 600900802
5	Cam roller1 – 146610502	30	Bottom cap – 670900891
6	Cam roller2 – 046610502	31	Cap – 600900891
7	Cam roller3 – 346610502	32	Piston – 640900802
8	Special pin – 600990502	33	Operating lever – 609910891
9	Spigot tube – 600900802	34	Bulb exhaust valve bush – 606110802
10	Valve – 800100891	35	Slug – 266110902
11	Choke plugs – 123910802	36	Piston cap – 003912802
12	Dummy coupling – 909910072	37	Lever – 646992891
13	Drain cap – 120200502	38	Plunger – 100110002
14	Piston – 633210891	39	Valve seat – 116810832
15	Bushing – 046230432	40	Piston rod – 020100832
16	Cover clamp – 608930562	41	Piston – 100900802
17	Exhaust valve piston – 106990891	42	Valve bush – 201510802
18	Double contact self-lapping EP brake valve – 609910072	43	Bush – 146810802
19	Bottom cap – 620900072	44	Secondary piston – 016811872
20	Valve seat – 646910802	45	Hexagonal headed cap – 190901402
21	Piston bush – 666110832	46	Bush – 166810802
22	Washer – 006110852	47	Isolating cock switch – 106110872
23	Valve stop – 120900832	48	Cap – 020220822
24	Nipples – 150900502	49	Washer – 066610802
25	Cap upper valve seat – 652900002	50	Valve bush – 201510802

Table 3 exhibits that both the methodologies are substantially proficient to achieve good solutions and effective in constructing part families. Both the above mentioned clustering techniques are tested on the same dataset retaining dissimilar results since each of them are based on distinct principals. Table 3 also depicts that the PSO-based soft-computing technique outperforms the heuristic clustering technique in terms of sum of similarity value. Consequently, the perfection percentage illustrated, authenticates the dominance of PSO over the heuristic clustering methods. In terms of computational time the proposed PSO method is good and took minimum CPU time (8.3825 CPU seconds) for the dataset tested which is better than the heuristic method (25.6739 CPU seconds). Therefore it can be stated that the proposed PSO approach is an effective part grouping method and could be utilised further for more complicated problems of GT.

Table 3 Comparison of performance shown by Heuristic and PSO

Data set size	*Part families obtained*		*Maximum similarities*		*Perfection percentage*	
	PSO	*Heuristic*	*PSO*	*Heuristic*	*PSO*	*Heuristic*
(50 × 9)	PF1 (24, 26, 27) PF2 (2, 5, 6, 8, 23, 32, 34, 46) PF3 (10, 28) PF4 (1, 3, 4, 13, 15, 16, 17, 18, 25, 33, 37, 38, 39, 40, 42, 43, 44, 48, 49) PF5 (7, 9, 11, 12, 14, 19, 20, 21, 22, 29, 30, 31, 35, 36, 41, 45, 47, 50)	PF1 (1, 17, 37) PF2 (3, 12, 14, 16, 30, 31, 33) PF3 (2, 4, 8, 9, 10, 13, 15, 18, 34, 40, 48) PF4 (11, 12, 22, 26, 27, 35, 39, 43, 44, 46, 47, 49) PF5 (5, 6, 7, 19, 20, 21, 23, 24, 25, 28, 29, 32, 36, 38, 41, 42, 45, 50)	4.2174	4.0991	81.982	84.348

5 Conclusions

Component family identification problems are crucial problems in the domain of GT. In past literature, it has been substantiated that the component family identification techniques are principally based on PFA. Part coding and classification-based approaches are believed to be the truly competent method to identify the component families, are merely adopted in designing production units on shop-floor. This study exploits a novel soft-computing approach based on PSO to develop efficient component families. The part coding scheme followed in present study is based on Opitz coding system. In order to understand the goodness of the solutions achieved, Offodile's similarity metric is exploited in this paper. The test dataset has been generated first using 55 components which are usually manufactured on factory shop-floor. The proposed soft-computing techniques are tested on the stated dataset and the obtained results are compared. The experimental results demonstrate that the PSO method is more effective in terms of computational efficiency and has outperformed the published heuristic technique. Further, this proposed method is applied on the industrial data collected from manufacturing company to show its practical implications.

References

Burbidge, J.L. (1963) 'Production flow analysis', *Production Engineers*, Vol. 42, pp.742–752.

Burbidge, J.L. (1996) *Production Flow Analysis for Planning Group Technology*, Oxford University Press, USA.

Darwin, C. (1929) *The Origin of Species by Means of Natural Selection or the Preservation of Favored Races in the Struggle for Life*, The Book League of America, (Originally published in 1859).

Galan, R., Racero, J., Eguia, I. and Garcia, J.M. (2007) 'A systematic approach for product families formation in reconfigurable manufacturing systems', *Robotics and Computer-Integrated Manufacturing*, Vol. 23, pp.489–502.

Gallagher, C.C. and Knight, W.A. (1985) *Group Technology Production Methods in Manufacture*, Chichester, Ellis Horwood.

Ghosh, T., Sengupta, S., Chattopadhyay, C. and Dan, P.K. (2011a) 'Meta-heuristics in cellular manufacturing: a state-of-the-art review', *International Journal of Industrial Engineering Computations*, Vol. 2, No. 1, pp.87–122.

Ghosh, T., Modak, M. and Dan, P.K. (2011b) 'SAPFOCS: a metaheuristic based approach to part family formation problems in group technology', *International Journal of Management Science and Engineering Management*, Vol. 6, No. 3, pp.231–240.

Ghosh, T., Modak, M. and Dan, P.K. (2011c) 'Coding and classification based heuristic technique for workpiece grouping problems in cellular manufacturing system', *International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies,* Vol. 2, No. 1, pp.53–72.

Groover, M.P. and Zimmers, E.W. Jr. (1984) *CAD/CAM: Computer-Aided Design and Manufacturing*, Prentice-Hall, New Jersey, USA.

Guerrero, L.C., Lozano, S., Canca, D., Guerrero, F., Larrañeta, J. and Onieva, L. (2000) 'Cell formation using sequence information and neural networks', *Proceedings of the 10th International Conference Flexible Automation and Intelligent Manufacturing*, pp.566–575.

Ham, I., Hitomi, K. and Yoshida, T. (1985) *Group Technology: Applications to Production Management*, Kluwer-Nijhoff Press, Boston.

Han, C. and Ham, I. (1986) 'Multiobjective cluster analysis for part family formation', *Journal of Manufacturing Systems*, Vol. 5, pp.223–229.

Haworth, E.A. (1968) 'Group technology-using the Opitz system', *Production Engineer*, Vol. 47, pp.25–35.

Jung, J. and Ahluwalia, R.S. (1992) 'FORCOD: a coding and classification system for formed parts', *Journal of Manufacturing Systems*, Vol. 10, pp.223–232.

Kaparthi, S. and Suresh, N. (1991) 'A neural network system for shape-based classification and coding of rotational parts', *International Journal of Production Research*, Vol. 29, pp.1771–1784.

Kennedy, J. and Eberhart, R.C. (1995) 'Particle swarm optimization', *Proceedings of IEEE International Conference on Neural Networks*, pp.1942–1948.

King, J.R. (1980) 'Machine-component grouping in production flow analysis: an approach using a rank order-clustering algorithm', *International Journal of Production Research*, Vol. 18, pp.213–232.

Lee-Post, A. (2000) 'Part family identification using a simple genetic algorithm', *International Journal of Production Research*, Vol. 38, pp.793–810.

Manimaran, A., Prabhakaran, G., Venkumar, P. and SudhakaraPandian, R. (2010) 'Manufacturing cell formation using graph decomposition', *International Journal of Services and Operations Management*, Vol. 7, No. 3, pp.300–316.

McAuley, J. (1972) 'Machine grouping for efficient production', *Production Engineer*, Vol. 51, pp.53–57.

Mitrofanov, S.P. (1966) *Scientific Principles of Group Technology*, Part I, National Lending Library of Science and Technology, (Originally published in 1959 as Russian text), Boston.

Moon, Y.B. (1992) 'Establishment of a neurocomputing model for part family/machine group identification', *Journal of Intelligent Manufacturing*, Vol. 3, pp.173–182.

Offodile, O.F. (1992) 'Application of similarity coefficient method to parts coding and classification analysis in group technology', *Journal of Manufacturing Systems*, Vol. 10, pp.442–448.

Opitz, H. (1970) *A Classification System to Design Workpieces*, Pergamon Press, UK.

Opitz, H. and Wiendahl, H. (1971) 'Group technology and manufacturing systems', *International Journal of Production Research*, Vol. 9, pp.181–203.

Sangwan, K.S. and Kodali, R. (2011) 'An intelligent hybrid model for the integrated design of cellular manufacturing systems', *International Journal of Services and Operations Management*, Vol. 9, No. 2, pp.202–226.

Singh, N. and Rajamani, D. (1996) *Cellular Manufacturing Systems: Design, Planning and Control*, Chapman and Hall Press, London.

SudhakaraPandian, R. and Mahapatra, S.S. (2008) 'Cell formation with ordinal-level data using ART1-based neural networks', *International Journal of Services and Operations Management*, Vol. 4, No. 5, pp.618–630.

SudhakaraPandian, R. and Mahapatra, S.S. (2010) 'Cell formation with operational time using ART1 networks', *International Journal of Services and Operations Management*, Vol. 6, No. 4, pp.377–397.

TNO (1975) *An Introduction to MICLASS*, Organization for Industrial Research, MA, USA.

Wemmerlov, U. and Hyer, N.L. (1987) 'Research issues in cellular manufacturing', *International Journal of Production Research*, Vol. 25, pp.413–432.

Appendix

The heuristic procedure

The heuristic technique is presented as (Ghosh et al., 2011c)

Step 1 *Input* the initial solution string 's_0' obtained randomly and set *max_iterations*

Step 2 *Calculate* the objective value 'f' for input string using equation (2)

Step 3 *Store* $f \leftarrow$ *best_objective_value*

Step 4 $s_0 \leftarrow$ *best_solution*

Step 5 *While* $i <=$ *max_iterations*

Step 6 *Create* initial set 'S' of randomly generated strings ($s_i \in S$, $_{i = 1, 2, \ldots, n}$)

Step 8 *For* $i = 1$ to n

Step 9 *Calculate* objective value f_i for $s_i \in S$

Step 10 *Compute* $\delta = (f_i - f)$

Step 11 *If* $\delta >$ small random no. (1.0000e-006)

Step 12 *best_solution* $= s_i$

Step 13 *best_objective_value* $= f_i$

Step 14 *Else*

Step 15 *Pick* a part randomly and put it to another family in s_i (interchange the positions of two elements of s_i with a small probability p_x)

Step 16 *Repeat* step 9 to 15

Step 17 *Accept* the arrangement

Step 18 *Else* eliminate the solution string

Step 19 $i = i + 1$.

Industrial dataset (part-attribute incidence matrix for 50 × 9 dataset)

	a1	a2	a3	a4	a5	a6	a7	a8	a9
p1	6	0	6	7	1	0	8	9	1
p2	6	0	0	1	1	0	5	0	2
p3	6	2	3	8	1	0	8	9	1
p4	0	0	6	1	5	0	5	0	2
p5	1	4	6	6	1	0	5	0	2
p6	0	4	6	6	1	0	5	0	2
p7	3	4	6	6	1	0	5	0	2
p8	6	0	0	9	9	0	5	0	2
p9	6	0	0	9	0	0	8	0	2
p10	8	0	0	1	0	0	8	9	1
p11	1	2	3	9	1	0	8	0	2
p12	9	0	9	9	1	0	0	7	2
p13	1	2	0	2	0	0	5	0	2
p14	6	3	3	2	1	0	8	9	1
p15	0	4	6	2	3	0	4	3	2
p16	6	0	8	9	3	0	5	6	2
p17	1	0	6	9	9	0	8	9	1
p18	6	0	9	9	1	0	0	7	2
p19	6	2	0	9	0	0	0	7	2
p20	6	4	6	9	1	0	8	0	2
p21	6	6	6	1	1	0	8	3	2
p22	0	0	6	1	1	0	8	5	2
p23	1	2	0	9	0	0	8	3	2
p24	1	5	0	9	0	0	5	0	2
p25	6	5	2	9	0	0	0	0	2
p26	6	4	3	1	1	0	5	3	2
p27	7	5	6	9	0	0	8	0	2
p28	6	0	0	9	3	0	8	0	2
p29	6	0	0	9	0	0	8	0	2
p30	6	7	0	9	0	0	8	9	1
p31	6	0	0	9	0	0	8	9	1
p32	6	4	0	9	0	0	8	0	2
p33	6	0	9	9	1	0	8	9	1
p34	6	0	6	1	1	0	8	0	2
p35	2	6	6	1	1	0	9	0	2
p36	0	0	3	9	1	2	8	0	2
p37	6	4	6	9	9	2	8	9	1
p38	1	0	0	1	1	0	0	0	2
p39	1	1	6	8	1	0	8	3	2
p40	0	2	0	1	0	0	8	3	2
p41	1	0	0	9	0	0	8	0	2
p42	2	0	1	5	1	0	8	0	2
p43	1	4	6	8	1	0	8	0	2
p44	0	1	6	8	1	1	8	7	2
p45	1	9	0	9	0	1	4	0	2
p46	1	6	6	8	1	0	8	0	2
p47	1	0	6	1	1	0	8	7	2
p48	0	2	0	2	2	0	8	2	2
p49	0	6	6	6	1	0	8	0	2
p50	2	0	1	5	1	0	8	0	2

Opitz coding system