APEM journal
Advances in Production Engineering & Management
Volume 19 | Number 2 | June 2024 | pp 197–208
https://doi.org/10.14743/apem2024.2.501
ISSN 1854-6250
Journal home: apem-journal.org
Original scientific paper

A modified bi-objective NSGA-II approach to sustainability in reconfiguration planning of dynamic cellular manufacturing systems

Sibanda, M.M.[a,*], **Padayachee, J.**[a]

[a]School of Mechanical Engineering, Howard college, University of Kwazulu-Natal, Durban, South Africa

A B S T R A C T

Manufacturing plant layouts are developed to facilitate optimal process flow. Modern manufacturing systems must meet present production demands and be adaptable to changes in process flow in the future. Dynamic Cellular Manufacturing Systems (DCMS) increase the flexibility of layouts by reconfiguring cell structure and equipment distribution, to effectively adjust part routings for optimal process flow. Frequent reconfiguring of plant layout may not always be feasible or economical, however, when new product releases are planned, reconfiguring the plant layout to optimise the workflow may be extremely beneficial. This paper presents a Non-dominated Sorting Genetic Algorithm (NSGA-II) approach to solving a DCMS problem in a sustainable, and responsible manner. A bi-objective integer programming model was developed over multiple planning horizons with fluctuating product demands. This model aims to achieve sustainability by reducing the cost of production, mitigating the environmental impact of production, and minimise negative social impacts on labourers that work in such environments. A penalty function approach was used to enforce the model constraints during optimisation. This study details trade-offs between the economic factors of a DCMS, the environmental implications of reconfiguring such a system, and the social impacts of reconfigurations on the workforce.

A R T I C L E   I N F O

Keywords:
Sustainable manufacturing;
Cellular manufacturing systems;
Plant layout reconfiguration;
Optimisation;
Non-dominated Sorting Genetic Algorithm (NSGA-II);
Penalty approach;
Bi-objective integer programming

**Corresponding author:*
217019040@stu.ukzn.ac.za
(Sibanda, M.M.)

Article history:
Received 17 January 2024
Revised 18 July 2024
Accepted 20 July 2024

1. Introduction

Cellular Manufacturing Systems (CMS) are a widely adopted method of manufacturing. The first step in developing a CMS is the process of Cell Formation (CF), followed by layout design [1]. CF is composed of two primary tasks: machine-cell allocation and part-family formation. Machine-cell allocation involves optimally grouping machines within cells. Part-family formation groups parts to be produced according to their production requirements, size, shape, or other geometric characteristics. Layout design also has two main tasks, namely intercell and intracell layout design. Batch manufacturing is used for several reasons including, but not limited to, efficient production, product tracking, short production cycles, short delivery time and volume constraints. It accounts for a significant fraction of all production as it helps manufactures cope with a wide variety of small manufacturing lot sizes. As the demand for these small lot sizes vary over time, CMS must be reconfigured to ensure that the current layout stays relevant and optimal for each demand period. Although a layout may be optimal for a particular demand period, when the demand changes that system may not be optimal for the new demand period. This would result in bottlenecks and inefficient production.

According to Lokesh and Jain [1] about 75 % of the revenue of Hewlett Packard was from products that were manufactured less than 3 years prior. This highlights a trend of short product lifecycles for many manufacturers today. A study from Defersha and Chen [2] revealed that considerable financial resources are spent annually on reconfiguring the layouts of production plants to improve the optimality of CMS. These great efforts are regularly made because part and material handling costs account for 20-50 % of production costs, suggesting that CMS need to be operated with optimum configurations to reduce production costs. They further suggested that these production costs could be reduced by 10-30 % by dynamically changing the layout of CMS.

In keeping with the world's Sustainable Development Goals (SDG), particularly goal 8 and 12, manufacturers may need ways to comply with new or stricter sustainability regulations [3]. Sustainability is an increasingly important consideration for manufacturers, as it involves the ability to produce goods and services in ways that meet present needs without compromising the ability of future generations to meet their own needs. The three pillars of sustainable manufacturing are economic limitations, environmental constraints, and social influences [3]. We acknowledge that sustainability can be measured in many ways, for our application we will consider carbon emissions as a general, all-inclusive measure of environmental sustainability, and the workload on machine operators as our limiting metric for social responsibility.

The motivation for this work stems from the need to address sustainability and social responsibility in manufacturing operations. In today's competitive market, cost optimisation is a key business concern, hence, this aspect of manufacturing is, and always has been, the primary focus in most research. However, we recognise that single cost objective models could have adverse environmental impacts, hence, incorporating environmental sustainability as a second objective in the proposed model provides a suitable trade-off to these related aspects of manufacturing. Introducing environmental considerations could lead to solutions with a lower carbon footprint, reduced energy consumption and waste generation. Furthermore, we understand that the implementation of DCMS can have significant social implications on the workforce. By incorporating social constraints into a DCMS model, researchers and organizations can identify solutions that minimise negative impacts on the well-being of employees; promote worker satisfaction, improve safety, and facilitate a smooth transition between DCMS periods. Ultimately, this research contributes to a more sustainable future for the manufacturing sector as a whole.

This paper is organised into seven sections and presents a bi-objective mathematical model that utilizes a penalty approach to minimise the operating costs and negative environmental impacts of a DCMS. A unique NSGA-II heuristic is used for the optimisation. The permissible social impact on the work force is introduced in the model through a set of inequality constraints. The remainder of the paper is organised as follows: section 2 presents a literature review while section 3 details the problem description. The mathematical model is described in section 4, and section 5 presents the solution approach. In section 6 we present and discuss the results from an optimisation study. This paper concludes with future work and recommendations in section 7.

2. Literature review

CMS can be formulated under static or dynamic conditions. Static conditions imply that product demand is known and does not change over the planning horizon. Dynamic conditions imply that the demand not only changes over the planning horizon, but can be modelled as either deterministic or probabilistic. When dynamic conditions are used for the modelling of CMS, it is then termed as a DCMS. The concept of DCMS was introduced by Rheault *et al.* [4] to overcome the limitations of CMS that only focused on static demand periods. Both cell formation and layout designs are reconfigured in each period to optimise the factory layout and reduce production costs. The reconfigurations often include adding and/or removing machines to or from the system. Although the dynamic modelling of CMS is more realistic, constraints on time, finances and the physical arrangement of the manufacturing system can make frequent reconfiguration infeasible or uneconomical.

Various approaches have been used to solve DCMS models across literature. Each method having its own advantages and limitations depending on the problem being addressed, and the number of factors modelled. Table 1 details the factors considered in the formulation of DCMS models by other researchers. Models are typically formulated as nonlinear optimisation problems. The speed in which a solution is generated, and the quality of the solution produced have been the primary attributes benchmarked across the different techniques. It was noted by Defersha and Chen [2] and is commonly acknowledged among researchers that some techniques are only suitable for certain sized problems. This has been observed in cases where some solvers did not produce a solution for a large problem regardless of an extended computational time allocation. It is noted that the computational capacity of the machines used for the study could have influenced the outcome, however, the results suggest that this could not be the main reason. Some researchers [5] focused their efforts on solving their models with a different solver than that used by previous authors, while others used a few different solvers common in literature to investigate the performance of each solver for the same problem. Commercial software constructed from mathematical programming algorithms have been used by most of the early authors, however, the use of meta-heuristic techniques has gradually increased in present day.

Table 1 Review of model parameters

Authors	**a**	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**	**n**	**o**	**p**	**q**	**r**	**s**	**t**	**u**	**v**
This paper	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Saxena & Jain [1]		√	√	√	√	√		√	√	√		√	√	√	√	√	√	√	√	√	√	√
Defersha & Chen [6]		√	√	√	√	√		√	√			√	√	√	√	√	√			√	√	√
Niakan *et al.* [7]		√	√	√			√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Safaei & Tavakkoli-Moghaddam [8]		√	√	√	√			√	√	√		√		√		√	√	√	√	√	√	√
Safaei *et al.* [9]	√		√	√	√	√		√	√	√		√	√	√	√	√	√	√	√	√	√	√
Ossama *et al.* [10]		√	√	√		√		√	√			√	√	√	√	√	√			√	√	√
Bayram & Sahin [11]		√	√	√		√		√	√	√		√	√	√	√	√	√			√	√	
Kia et al [12]		√	√	√				√	√	√		√	√	√	√	√	√			√	√	
Shahram *et al.* [13]			√	√	√		√	√	√			√	√	√	√	√	√			√	√	√
Aramoon *et al.* [14]		√	√	√		√		√	√			√	√	√	√	√	√	√		√	√	√

a: Selecting the best route; **b**: Allowing alternative routing; **c**: Deterministic demand fluctuation; **d**: Dynamic cell reconfiguration; **e**: Intercell workload balancing; **f**: Sequence of operations; **g**: Setup time; **h**: Cell size constraint; **i**: Intercell material movement; **j**: Intracell material movement; **k**: Operator allocation; **l**: Machine capacity; **m**: Identical machines within a cell; **n**: Identical machines in the entire system; **o**: Machine investment cost; **p**: Unit operation time; **q**: Machine operation cost; **r**: Intercell batch size movement; **s**: Intracell batch size; **t**: Multiperiod planning; **u**: Machine relocation; **v**: Process batch size.

In recent publications, DCMS, Genetic Algorithm (GA), and varied multi-objective models have been utilised to identify potential improvements and obstacles within diverse organisations. The work of [15] presents a unique layered GA with a reinitialisation approach to preserving population diversity. They found their model to improve avoiding premature stall in local minima. The work of [16] incorporates a focus on environmental implications of a supply chain and how goods sourced with environmental consideration showed robust resource allocation. Furthermore, [17] used a variable neighbourhood search GA to enhance workload balancing in an extended travelling salesman problem. Their work showed how the workload balancing could be intelligently introduced for better computational efficiency. A NSGA-III was used by [18] to improve downtime in a production environment. They compared its performance to that of a NSGA-II and found that the additional objective function improved the quality of solutions generated. To help present-day manufacturers deal with diverse and volatile demand fluctuations, the work of [19] presents a mathematical model with a GA to show how work from academics can be applied in industry. With the aid of a numerical case study, they successfully showed how literature ideas are effective in industry scenarios. This paper seeks to merge the key elements from these reviewed works. They each promote different ideas for improved solutions; hence, we propose a mathematical model with an environmental objective function, and social considerations modelled in constraint form, including a workload balancing constraint. The proposed NSGA-II is modified to cater for unique solutions and is applied over three industry problems to show how it is generally applicable, and not biased to a single data set.

3. Problem description

We propose a DCMS model where part operations are performed sequentially on any capable machine. The operation capabilities of each machine are known and are deterministic. Each part operation can be performed by more than one machine, creating the possibility of alternate process routings. The operation time of all part operations, on all capable machines, are known and deterministic. Furthermore, we assume that the planning horizon is for a predetermined number of periods, with each machine having a predetermined production capacity, expressed in hours, for that period. These production periods are of equal duration. Duplicating machines within and across cells to meet capacity requirements is permissible without limit to the number of machines procured, so long as the cell size constraints are satisfied.

We assume that parts are transported between cells in batches and the transportation cost per batch is known. The per batch Green House Gas (GHG) emissions are also known. The model assumes that intra cell material handling is manually managed by human porters, or is managed by equipment with a negligible power consumption compared to the inter cell material handling equipment. The demand in each period must be satisfied within that period, no backorders, or inventory is kept between periods. All machine types can be relocated between cells. The cost and emissions generated for moving a machine from one cell to another are known. Relocation impacts include uninstalling, moving and reinstallation efforts altogether.

To simplify the model, it is assumed that machine relocation time is negligible and does not affect the available time on the machine for that period. Meaning that plant shutdown time between periods is not included in the total time over the planning horizon. Machines are independently grouped, meaning that there are no machine colocation or separation constraints. We further assume that one worker/operator is assigned to each machine. When a machine is purchased, an experienced worker is hired to operate the machine. When a machine is retired, its operator is retrenched. When a machine is relocated, the assigned operator is relocated with the machine. Machines are added at the beginning of each period and retired/relocated at the end of each period. All machines will be retired at the end of the last demand period to prepare the plant for the next project. Lot splitting is not permissible and job setup time is included in the part processing time.

We present the model notation as:

Sets

h – time index, $h = 1,2,3,\dots,H$ (H is total number of periods)
p – part index, $p = 1,2,3,\dots,P_h$ (P_h is total number of parts in period h)
j – index of operations on parts, $j = 1,2,3,\dots,O_p$ (O_p is total number of operations needed for part p)
m – machine index, $m = 1,2,3,\dots,M$ (M is total number of machines)
c – cell index, $c = 1,2,3,\dots,C$ (C is total number of cells)

Decision variables

x_{jpmch} – 1 if operation j of part p by machine m is done in cell c during time h, otherwise 0
N_{mch} – number of type m machines placed in cell c at time h
${K^{+}}_{mch}$ – number of type m machines added to cell c at the beginning of time h
${K^{-}}_{mch}$ – number of type m machines removed from cell c at the end of time h
a_{jpm} – 1 if operation j of part p can be done on machine m; otherwise 0

Parameters

LB – Cell size lower bound
UB – Cell size upper bound
D_{ph} – Demand of part p during period h
B_p^{inter} – Intercell batch size for part p
B_p^{intra} – Intracell batch size for part p
T_m – Available time on machine m (h)

t_{jpm} – time for operation j of part p on machine m (h)
α_m – Overhead cost of machine type m
β_m– Variable operating cost for each unit time on machine m (R/h)
γ^{inter} – Batch intercell material handling cost
γ^{intra} – Batch intracell material handling cost
δ_m – Relocation cost for machine type m
φ^{inter} – Batch intercell carbon emissions (kgCO2)
τ_m – Carbon emissions from adding and removing machine type m (kgCO2)
μ_m – Carbon emissions from idle time of machine type m (kgCO2/h)
ε_m – Variable carbon emissions for each operating unit time, on machine type m (kgCO2/h)
σ_m – Carbon emissions from relocating machine type m (kgCO2)
q – Workload balancing factor (taken as ± 0.75 for 75 %)
η – maximum number of different operations an operator can be assigned (taken as 3)

We applied the model to three different problems from literature and solved each problem with the addition of supplementing data to incorporate the second objective. The environmental data for each problem is contained in Table 2, while the reader is referred to the literature [20, 9] for the rest of the data. A linear weighting was observed, such that low-cost machines had a higher environmental impact, while costly machines had a lower environmental impact.

Table 2 Environmental objective input data

Problem	Machines	M1	M2	M3	M4	M5	M6	M7	M8	M9
1	Sourcing emissions	5087	3149	4313	4408	6200	4509	2480	3815	n/a
2		5425	6200	5710	4822	5425	4931	4340	4520	4822
3		6200	5320	4960	5320	6200	4650	n/a	n/a	n/a
1	Relocation emissions	656	406	556	568	800	581	320	492	n/a
2		500	600	550	350	500	400	250	300	350
3		310	266	150	266	310	233	n/a	n/a	n/a
1	Idle time carbon emissions	11	7	9	9.5	13.6	10	5.4	8.4	n/a
2		6.5	8	7	5	6.5	5.5	3	4	5
3		8	6	4	6	8	2	n/a	n/a	n/a

Problem	Parts	P1	P2	P3	P4	P5	P6	P7	P8
1	Unit intercell emissions	2	2	2	2	n/a	n/a	n/a	n/a
2		1.5	2.1	1.8	1.1	0.95	2.2	2	n/a
3		1.8	1.8	2.25	1.2	2.25	1.13	1.13	1.5

4. Mathematical model

Minimise:

$$\begin{aligned} Z_1 = & \sum_{h=1}^{H}\sum_{m=1}^{M}\sum_{c=1}^{C} N_{mch}\alpha_m \\ & + \sum_{h=1}^{H}\sum_{c=1}^{C}\sum_{p=1}^{P}\sum_{j=1}^{Op}\sum_{m=1}^{M} \beta_m D_{ph} t_{jpm} x_{jpmch} \\ & + \frac{1}{2}\sum_{h=1}^{H}\sum_{p=1}^{P}\sum_{j=1}^{OP-1}\sum_{c=1}^{C}\left\lceil\frac{D_{ph}}{B_p^{inter}}\right\rceil \gamma^{inter}\left|\sum_{m=1}^{M} x_{(j+1)pmch} - \sum_{m=1}^{M} x_{jpmch}\right| \\ & + \frac{1}{2}\sum_{h=1}^{H}\sum_{p=1}^{P}\sum_{j=1}^{OP-1}\sum_{c=1}^{C}\left\lceil\frac{D_{ph}}{B_p^{intra}}\right\rceil \gamma^{intra}\left(\sum_{m=1}^{M}\left|x_{(j+1)pmch} - x_{jpmch}\right| \right. \\ & \left. - \left|\sum_{m=1}^{M} x_{(j+1)pch} - \sum_{m=1}^{M} x_{jpch}\right|\right) + \frac{1}{2}\sum_{h=1}^{H}\sum_{m=1}^{M}\sum_{c=1}^{C}\delta_m\left(K^{+}{}_{mch} + K^{-}{}_{mch}\right) \end{aligned} \tag{1}$$

$$Z_2 = \sum_{h=1}^{H}\sum_{c=1}^{C}\sum_{m=1}^{M}\tau_m\left|N_{mc(h+1)} - N_{mch}\right| + \sum_{h=1}^{H}\sum_{m=1}^{M}\sum_{c=1}^{C}\sigma_m({K^+}_{mch} + {K^-}_{mch}) + \sum_{h=1}^{H}\sum_{c=1}^{C}\sum_{m=1}^{M}\mu_m\left\{\sum_{p=1}^{P}\sum_{j=1}^{OP}T_mN_{mch} - D_{ph}t_{jpm}x_{jpmch}\right\} + \frac{1}{2}\sum_{h=1}^{H}\sum_{p=1}^{P}\sum_{j=1}^{OP-1}\sum_{c=1}^{C}\left[\frac{D_{ph}}{B_p^{inter}}\right]\varphi^{inter}\left|\sum_{m=1}^{M}x_{(j+1)pmch} - \sum_{m=1}^{M}x_{jpmch}\right| \tag{2}$$

Subject to:

$$\sum_{c=1}^{C}\sum_{m=1}^{M}a_{jpm}x_{jpmch} = 1 \quad \forall\, j,p\,,h \tag{3}$$

$$\sum_{p=1}^{P}\sum_{j=1}^{OP}D_{ph}t_{jpm}x_{jpmch} \leq T_mN_{mch} \quad \forall\, m,c,h \tag{4}$$

$$LB \leq \sum_{m=1}^{M}N_{mch} \leq UB \quad \forall c,h \tag{5}$$

$$\sum_{m=1}^{M}\sum_{p=1}^{P}\sum_{j=1}^{OP}x_{jpmch} \geq \frac{q}{C}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{p=1}^{P}\sum_{j=1}^{OP}x_{jpmch} \quad \forall\, c,h \tag{6}$$

$$\sum_{p=1}^{P}\sum_{j=1}^{OP}x_{jpmch} \leq \eta N_{mch} \quad \forall\, m,c,h \tag{7}$$

$$N_{mc(h-1)} + {K^+}_{mch} - {K^-}_{mch} = N_{mch} \;\; \forall\, m,c,h \tag{8}$$

$$x_{jpmch} \in \{0,1\}\,, N_{mch}\,, {K^+}_{mch}\,, {K^-}_{mch} \geq 0 \in \mathbb{Z} \tag{9}$$

Eq. 1 is a cost minimisation objective, consisting of five terms. The first term is the machine overhead/rental cost. The second term represents the machine operational cost per unit time. The third and fourth terms calculate the intercell and intracell material handling costs respectfully. The machine relocation cost is calculated in the fifth term. Eq. 2 represents an environmental impact minimisation objective, consisting of four terms measured in $kgCO_2$. The first term represents the carbon emissions from the machine manufacturing, transportation to and from the plant on purchase/retirement, and from the recycling/disposal of its components. The second term represents emissions from machine relocation within the system. The third term measures the emissions from the machine while idling. The fourth term measures the emissions from intercell material handling. Eq. 3 ensures that each part operation is assigned to a single machine and is performed only once. Eq. 4 bounds the model from violating the capacity of the machines. Eq. 5 introduces the lower and upper bounds for the number of machines permissible in a cell. Eq. 6 is the social constraint for intercell workload balancing. Eq. 7 is the social constraint on the number of different part operations a single operator can be assigned to perform. Eq. 8 ensures that the machine placement and relocation is calculated accurately across demand periods.

5. Solution approach

GAs are an evolutionary metaheuristic technique that progressively iterates from a randomly generated initial population, towards an optimal solution set through crossover and mutation operations [18]. The unique GA developed to solve the mathematical model will be detailed hereafter in eight sections. Fig. 1 depicts the holistic program flow of the optimisation process.

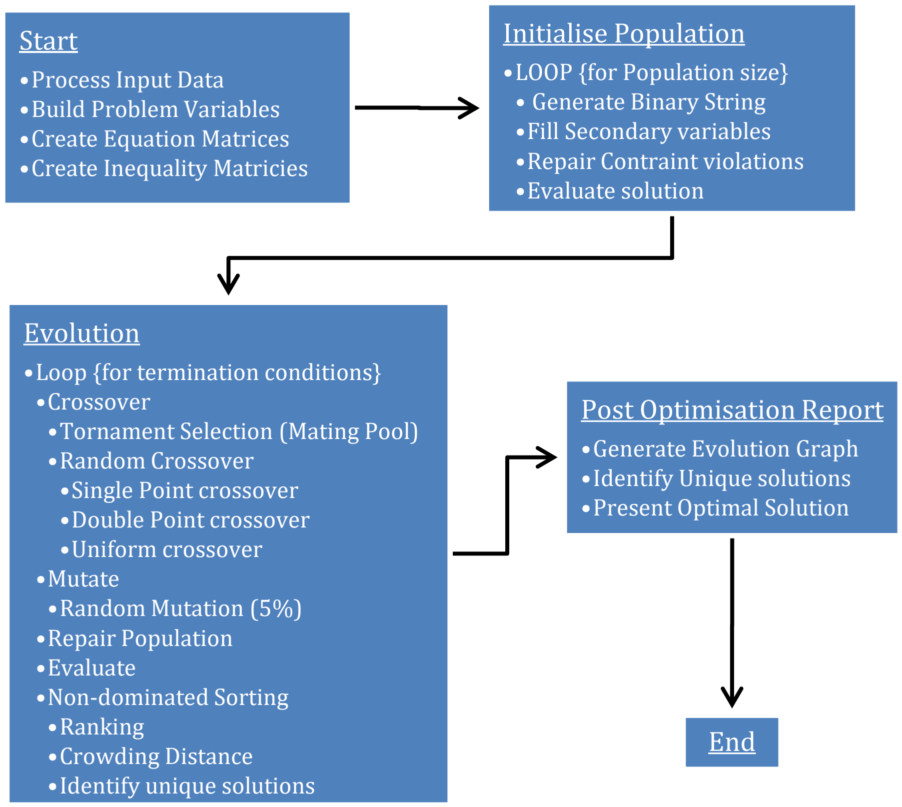


Fig. 1 GA optimisation algorithm flow

5.1 Chromosome encoding

Within the population, each solution's chromosome structure was encoded as a single row array of integers representing all variables. Each array was split into two sections, separating the binary part production decision variables x_{jpmch} from the rest. The part production decision variables were used as the primary variables while the other variables were changed accordingly, to satisfy the problem constraints. Hence, during initialisation, crossover and mutation, only primary variables were changed and/or modified, while the rest were adapted by a custom repair function to suit the new arrangements. An overview of the repair function is given hereafter.

5.2 Initial population

To generate the initial population, a random binary $M \times N$ matrix was constructed, where M was the input population size and N was the number of primary decision variables. Each row, which represented a solution in the population, then had the rest of the variables populated accordingly. The solutions were individually scrutinised against the constraints using the penalty approach, and a penalty value was assigned to each solution based on the number of constraints violated. An attempt to repair solutions that violated constraints was initialised after the chromosomes were evaluated. The initial population was then filled with the solutions that had a penalty value of zero from the lot. This initialisation strategy is similar to that used by Kia [20].

5.3 Tournament selection

Instead of a roulette wheel approach [19], a tournament selection [21] process was held for each generation to select the prime solutions for the mating pool. A random bi-contestant structure was observed for parent selection. Better ranking solutions won the contest and were added to the mating pool, however, in cases where contestants were of equal rank, solutions with a higher crowding distance were selected. In cases where contestants had the same rank and an equal crowding distance, a random solution was selected between the two. Each solution in the population participated in two tournaments, such that there would be two copies of the best solution in the mating pool, and the worst solution of the population would never enter the mating pool. Tournaments were conducted until the mating pool was the same size as the original population.

5.4 Crossover

Crossover was performed with 80 % probability. To generate offspring, three crossover operations were utilised, namely, Single Point Crossover, Double Point Crossover and Uniform Crossover [15]. Two solutions were randomly selected from the mating pool and a crossover operator was randomly selected from the three operators. The crossover operations were performed only over the binary primary variables as discussed. After crossover, each offspring was evaluated and if it violated any constraints, it was repaired and then re-evaluated.

5.5 Mutation

A 5 % mutation rate over the primary genes of crossover offspring was observed. A uniform mutation strategy was employed [20], which randomly mutated genes across all the binary decision variables. Since the primary variables were binary, the mutation changed the genes randomly selected for mutation from either 1 to 0 or vice versa. After the mutation, mutant solutions that violated the constraints were repaired and all solutions were evaluated and grouped with the rest of the solutions for ranking.

5.6 Repair function

A repair function was developed to fix solutions that violated any of the constraints during initialisation, crossover, and mutation. When a solution was under repair, initially the primary variables were repaired such that constraint Eq. 3 was observed. Thereafter, machines were placed in the appropriate cells in keeping with constraint Eq. 4. All machines that did not have parts allocated to them were removed from the solution, and lastly, the demand periods were then joined together to determine machine relocation variables. Thereafter, repaired solutions were checked against cell size constraints Eq. 5. The penalty approach was effective at maintaining most of the constraints, however, not all of them. The cell upper bound constraints Eq. 5 were particularly difficult to repair without redefining the primary variables and increasing the computational effort of the algorithm needlessly. Hence, when a solution did not satisfy all the constraints after the first repair attempt, it was discarded. If the cell lower bound was violated, a minimal number of random machines were placed in the cell to satisfy the constraint.

5.7 Penalty function and evaluation

In keeping with the non-dominated sorting method of evaluation, and in conjunction with the penalty approach, each solution was graded in three major categories namely, penalty value, rank, and crowding distance. Other attributes of each solution which were also evaluated were solution uniqueness, number of solutions that dominate that solution, and the number of solutions that were dominated by that solution. For a given solution $\boldsymbol{Z}$, where $\boldsymbol{A}$, $\boldsymbol{A_{eq}}$, and $\boldsymbol{b_{eq}}$ are matrix representations of the problem constraints, and M is a large positive value, the penalty expression used for the inequality constraints was of the form:

$$P_1(Z) = M \cdot \max\,[0, [\,A - b]]^2 \tag{10}$$

while the penalty expression for the equality constraints was modelled:

$$P_2(Z) = M \cdot sum\left\{\left[sum\left[\,A_{eq} - b_{eq}\,\right]\right]^2\right\} \tag{11}$$

5.8 Unique non-dominated sorting

The weighting of the objective functions was enforced using principles from the NSGA-II approach to avoid localised stalling and protracted computational time. We extended on this method, by sorting the final population for each generation with the best unique solutions from the combined populations, i.e., the previous generation population, the offspring population, and the mutant population. During experimentation, this modified approach not only improved the range of solution exploration, but also improved population diversity across generations. The method initially groups solutions according to pareto front rankings; thereafter, the solutions are sorted according to their individual crowding distances within each rank. Solutions are then compared to each other and the top unique 200 (population size) solutions were passed on to the next generation. When there were less than 200 unique solutions the algorithm added duplicate solutions from the top-ranking solution going down, to fill the required number.

For a single minimisation objective function, solution $Z(x_1)$ dominates solution $Z(x_2)$ if the value of x_1 is less than that of x_2. When a second minimisation objective is introduced, $Z(x_1, y_1)$ dominates $Z(x_2, y_2)$ if and only if $(x_1 \leq x_2$ and $y_1 \leq y_2)$ and $(x_1 < x_2$ or $y_1 < y_2)$. The Boolean result from this expression was used to identify non-dominated solutions across populations and assign rank accordingly. Where N is the total number of fronts/ranks for each generation, F is the pareto front, and M is the total number of objectives, the crowding distance (d_i) for each solution was calculated using the equation:

$$d_i = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{F_n(Z_m)(i+1) - F_n(Z_m)(i-1)}{F_n^{\,max} - F_n^{\,min}} \quad \forall\, i \tag{12}$$

6. Results and discussion

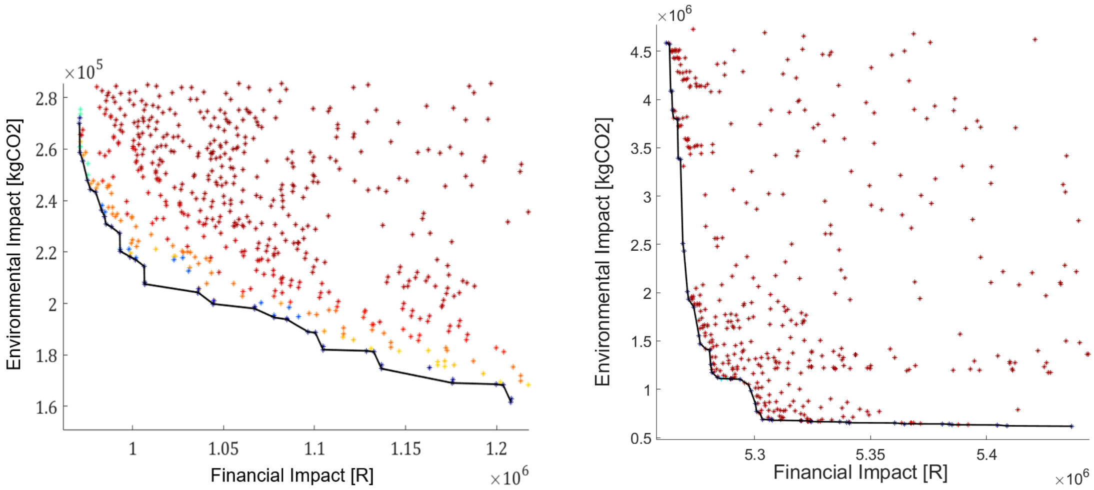
The problems were solved on MATLAB R2021a running on a personal computer with a 2.70 GHz, 7th generation Intel i7 processor and 8 GB RAM. The GA was run for 1000 generations to ensure complete evolution stall for each problem. The results are detailed in Table 3.

Table 3 Post optimisation data

No. of parts	No. of machines	*LB* / UB	No. of constraints	No. of non-zero variables	No. of unique solutions	Min. cycle time (h)	Average Machine utilisation (%)	Lowest obj. 1 val. R(ZAR)	Lowest obj. 2 val. $kgCO_2$
4	8	1/5	1251	1104	31	7059	56-74	970,910	161,682.5
7	9	1/7	2223	1782	73	66	52-67	5,236,400	445,827.8
8	6	1/5	1773	1908	128	689	61-70	342,518	105,210.1

Valuable insights can be extracted from the consistency of the concave shape of the pareto fronts resulting from the optimisation, depicted in Fig. 2 and Fig. 3. It suggests that the model does not favour a particular data set, and that the results are not problem biased. Furthermore, it suggests a conflicting relationship between the objective functions, implying a non-linear trade-off between the two. This is expected since the objective functions have a complex non-linear relation to each other.

The pareto front's concavity is owed to the different aspects each term in each objective function seeks to optimise. In Eq. 1, the first term seeks to minimise the number of machines held in each period, while the first term in Eq. 2 seeks to minimise the changes in the machines held. These terms compete for dominance over machine allocations. Although the fifth term in Eq. 1 and the second term in Eq. 2 both seek to reduce machine relocation, they too will conflict over the particular machines to be moved as one objective favours moving cost effective machines and the other environmentally friendly machines. This conflict stems from the difference in weighting for the machines for each objective. Another dynamic relationship observed is between term 2 of Eq. 1 and term 3 of Eq. 2, as one seeks to minimize the utilization of machines while the other seeks to maximise machine utilisation. Some linearity is shared by term 3 of Eq. 1 and term 4 of Eq. 2 as they both seek to minimise intercell material handling. However, they are limited by the social objective imposed by constraint Eq. 7 which prohibits the overloading of machine operators with multiple different operations. This inclines the model to create larger cells with a higher number of machines and less intercell movement, or fewer machines with higher intercell material handling. The cell upper bound constraint Eq. 5 also influences the model on the machine placements by adding a ceiling to the permissible cell sizes.



(a) (b)

Fig. 2 Pareto fronts of: (a) problem 1, (b) problem 2

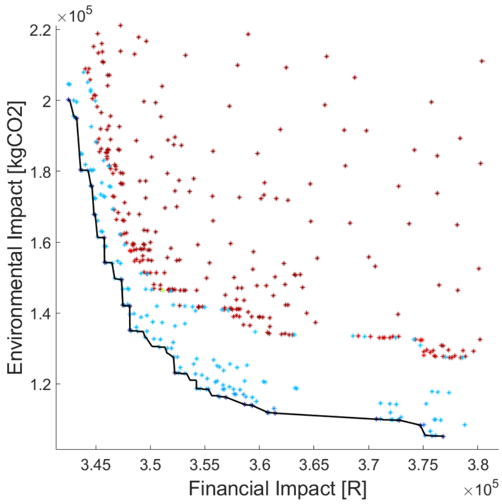


Fig. 3 Pareto front of problem 3

It is worth noting that when these problems were solved by previous researchers, their work did not consider environmental and social objectives, hence, their work is not directly relatable to ours on a one-to-one comparison. However, from a holistic perspective, the pareto front shows that the most cost-effective solutions are not the most environmentally friendly ones. Rather, solutions are a compromise between the two objectives. Since the social objective is embedded in the model constraints, it is observed in each solution. Although we have suggested selecting each problem's solution based on minimum cycle time, or highest machine utilisation, the existence of multiple unique solutions presents opportunity for additional post optimisation processing that could reveal more insights and improved practicality. Some post optimisation processing could include a sensitivity analysis on the different parameters of each problem, or a clustering analysis to identify solution groups with similar trade-offs. However, the most practical post optimisation assessment would be developing quality metrics to narrow down preferred solutions. These metrics could analyse the volume and/or direction of material flow per solution, and/or the traffic of machine relocations. These metrics are not considered in the model but could greatly affect the safety in a production environment, and the human efforts needed to calibrate equipment once it has been moved. Ultimately, the pareto front of unique solutions to choose from provides decision makers options for informed decision making. Once a system has been optimised, a project budget together with an environmental assessment of acceptable emissions can be used to select the best layout according to project specifications on a case-by-case basis.

7. Conclusion

It is known that DCMS are NP hard problems. We proposed a multi-objective mathematical model to solve a novel multiperiod DCMS model which was NP-complete. The problems were formulated as binary integer programming models and solved heuristically by a NSGA-II approach. The model has two objective functions that aim to minimise economic cost, and environmental impact of production. A social aspect of the model was introduced through a set of constraints. The penalty method was used to enforce the model constraints. Three numerical example problems from literature were solved to demonstrate the effectiveness of the genetic algorithm, and a pareto plot was generated for each problem. From post experiment analysis, it was observed that all the pareto fronts were concaved and that the generated plots suggested solutions which were a weighted compromise between the two objectives. The results indicate that incorporating an environmental objective produces alternatives that are not only economically viable but also socially and environmentally responsible. Future enhancements to the model could include factors such as intracell workload balancing, lot splitting, inventory holding, and outsourcing.

Acknowledgements

We acknowledge the Reino Stegen foundation for funding this research under their bursary program at UKZN, and DES GROUP (PTY) LTD 2000/008589/07 for supporting this work.

References

[1] Saxena, L.K., Jain, P.K. (2011). Dynamic cellular manufacturing systems- a comprehensive model, *The International Journal of Advanced Manufacturing Technology,* Vol. 53, 11-34, doi: 10.1007/s00170-010-2842-9.

[2] Defersha, F.M., Chen, M. (2008). A parallel multiple Markov chain simulated annealing for multi-period manufacturing cell formation problems, *The International Journal of Advanced Manufacturing Technology,* Vol. 37, 140-156, doi: 10.1007/s00170-007-0947-6.

[3] Rasshif, M.N., Wicaksono, P.A., Hartini, S. (2023). Desain sustainable value stream mapping untuk evaluasi kinerja keberlanjutan perusahaan obat, *Industrial Engineering Online Journal,* Vol.12, No. 4, 1-9.

[4] Rheault, M., Drolet, J.R., Abdulnour, G. (1996). Dynamic cellular manufacturing system (DCMS), *Computers & Industrial Engineering,* Vol. 31, No. 1-2, 143-146, doi: 10.1016/0360-8352(96)00098-8.

[5] Khettabi, I., Benyoucef, L., Boutiche, M.A. (2022). Sustainable multi-objective process planning in reconfigurable manufacturing environment: Adapted new dynamic NSGA-II vs New NSGA-III, *International Journal of Production Research*, Vol. 20, No. 60, 6329-6349, doi: 10.1080/00207543.2022.2044537.

[6] Defersha, F.M., Chen, M. (2006). Machine cell formation using a mathematical model and a genetic-algorithm-based heuristic, *International Journal of Production Research,* Vol. 12, No. 44, 2421-2444, doi: 10.1080/00207540500337833.

[7] Niakan, F., Baboli, A., Moyaux, T., Botta-Genoulaz, V. (2015). A bi-objective model in sustainable dynamic cell formation problem with skill-based worker assignment, *Journal of Manufacturing Systems,* Vol. 38, 46-62, doi: 10.1016/j.jmsy.2015.11.001.

[8] Safaei, N., Tavakkoli-Moghaddam, R. (2009). Integrated multi-period cell formation and subcontracting production planning in dynamic cellular manufacturing systems, *International Journal of Production Economics,* Vol. 120, No. 2, 301-314, doi: 10.1016/j.ijpe.2008.12.013.

[9] Safaei, N., Saidi-Mehrabad, M., Tavakkoli-Moghaddam, R., Sassani, F. (2008). A fuzzy programming approach for a cell formation problem with dynamic and uncertain conditions, *Fuzzy Sets and Systems,* Vol. 159, No. 2, 215-236, doi: 10.1016/j.fss.2007.06.014.

[10 Ossama, M., Youssef, A.M.A., Shalaby, M.A. (2014). A multi-period cell formation model for reconfigurable manufacturing systems, *Procedia CIRP,* Vol. 17, 130-135, doi: 10.1016/j.procir.2014.01.120.
[11] Bayram, H., Sahin, R. (2016). A comprehensive mathematical model for dynamic cellular manufacturing system design and linear programming embedded hybrid solution techniques, *Computers & Industrial Engineering,* Vol. 91, 10-29, doi: 10.1016/j.cie.2015.10.014.
[12] Kia, R., Baboli, A., Javadian, N., Tavakkoli-Moghaddam, R., Kazemi, M., Khorrami, J. (2012). Solving a group layout design model of a dynamic cellular manufacturing system with alternative process routings, lot splitting and flexible reconfiguration by simulated annealing, Vol. 39, No. 11, 2642-2658, doi: 10.1016/j.cor.2012.01.012.
[13] Sharifi, S., Chauhan, S.S., Bhuiyan, N. (2014). A dynamic programming approach to ga-based heuristic for multi-period CF problems, *Journal of Manufacturing Systems,* Vol. 33, No. 3, 366-375, doi: 10.1016/j.jmsy.2014.02.004.
[14] Aramon Bajestani, M., Rabbani, M., Rahimi-Vahed, A.R., Baharian Khoshkhou, G. (2009). A multi-objective scatter search for a dynamic cell formation problem, *Computers & Operations Research,* Vol. 36, No. 3, 777-794, doi: 10.1016/j.cor.2007.10.026.
[15] Amjad, M.K., Butt, S.I., Anjum, N., Chaudhry, I.A., Faping, Z., Khan, M. (2020). A layered genetic algorithm with iterative diversification for optimization of flexible job shop scheduling problems, *Advances in Production Engineering & Management,* Vol. 15, No. 4, 377-389, doi: 10.14743/apem2020.4.372.
[16] Fang, I.W., Lin, W.T. (2021). A multi-objective optimal decision model for a green closed-loop supply chain under uncertainty: A real industrial case study, *Advances in Production Engineering & Management,* Vol. 16, No. 2, 161-172, doi: 10.14743/apem2021.2.391.
[17] Wang, Y.D., Lu, X.C., Shen, J.R. (2021). Improved genetic algorithm (VNS-GA) using polar coordinate classification for workload balanced multiple Traveling Salesman Problem (mTSP), *Advances in Production Engineering & Management,* Vol. 16, No. 2, 173-184, doi: 10.14743/apem2021.2.392.
[18] Xu, E.B., Yang, M.S., Li, Y., Gao, X.Q., Wang, Z.Y., Ren, L.J. (2021). A multi-objective selective maintenance optimization method for series-parallel systems using NSGA-III and NSGA-II evolutionary algorithms, *Advances in Production Engineering & Management,* Vol. 16, No. 3, 372-384, doi: 10.14743/apem2021.3.407.
[19] Guzman, E., Poler, R., Andres, B., (2023). A metaheuristic approach combining genetic algorithm and mixed integer linear programming model for production and distribution planning in the supply chain, *Advances in Production Engineering & Management,* Vol. 18, No. 1, 19-31, doi: 10.14743/apem2023.1.454.
[20] Kia, R., Khaksar-Haghani, F., Javadian, N., Tavakkoli-Moghaddam, R. (2013). Solving a multi-floor layout design model of a dynamic cellular manufacturing system by an efficient genetic algorithm, *Journal of Manufacturing Systems,* Vol. 33, No. 1, 218-232, doi: 10.1016/j.jmsy.2013.12.005.
[21] Balakrishnan, J., Cheng, C.H., Conway, D.G., Lau, C.M. (2003). A hybrid genetic algorithm for the dynamic plant layout problem, *International Journal of Production Economics,* Vol. 86, No. 2, 107-120, doi: 10.1016/S0925-5273(03)00027-6.