Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Applied Soft Computing 5 (2005) 281–299
Applied Soft Computing
www.elsevier.com/locate/asoc

# Evolutionary computing in manufacturing industry: an overview of recent applications

V. Oduguwa*, A. Tiwari, R. Roy

*Enterprise Integration, School of Industrial and Manufacturing Science, Cranfield University, Bedford MK430AL, UK*

**Abstract**

Traditional methods often employed to solve complex real world problems tend to inhibit elaborate exploration of the search space. They can be expensive and often results in sub-optimal solutions. Evolutionary computation (EC) is generating considerable interest for solving real world engineering problems. They are proving robust in delivering global optimal solutions and helping to resolve limitations encountered in traditional methods. EC harnesses the power of natural selection to turn computers into optimisation tools. The core methodologies of EC are genetic algorithms (GA), evolutionary programming (EP), evolution strategies (ES) and genetic programming (GP). This paper attempts to bridge the gap between theory and practice by exploring characteristics of real world problems and by surveying recent EC applications for solving real world problems in the manufacturing industry. The survey outlines the current status and trends of EC applications in manufacturing industry. For each application domain, the paper describes the general domain problem, common issues, current trends, and the improvements generated by adopting the GA strategy. The paper concludes with an outline of inhibitors to industrial applications of optimisation algorithms.


*Keywords:* Evolutionary computing; Genetic algorithms; Manufacturing industry

## 1. Introduction

Real world engineering can be characterised as having chaotic disturbances, randomness and complex non-linear dynamics [1]. Most industrial processes are usually large scale, highly dimensional, non-linear, highly uncertain with complex highly skilled operators to control the process plants. Conventional methods such as trial and error are often used to solve complex optimisation problems. This approach relies on the use of the analyst's qualitative ($Q^L$) knowledge to explore the design space [2,3]. Expensive advanced computational analyses (such as finite element) are also used to understand the behaviour of complex engineering problems. These are often invoked repeatedly during the search process making the optimisation and concept exploration time consuming. This traditional search method often results in sub-optimal solutions due to inherent limitations in incomplete knowledge representation and the fact that elaborate exploration of the design space is inhibited. There is also a tendency to accept local optimal solutions considered to be sufficiently

* Corresponding author. Tel.: +44 1234 754072.
*E-mail address:* v.oduguwa@cranfield.ac.uk (V. Oduguwa).

1568-4946/$ – see front matter 
doi:10.1016/j.asoc.2004.08.003

good for the chosen objective due to the following reasons: subjective judgment, similarity to historical result or simply constraint on time to deliver workable solutions. Multiple global optimal solutions are desirable for these classes of problems to give alternative solutions in the presence of increasing dynamic and ill-defined problem space. EC techniques are receiving an increasing interest for solving real world engineering problems. These techniques are proving robust in delivering global optimal quality solutions and are helping to resolve some of the complexity issues encountered in real world problems.

EC is a method of harnessing the power of natural selection to turn computers into optimisation tools. EC is one of the main constituents of soft computing (SC), where SC is a collection of methodologies including neural computing (NC) and fuzzy computation (FC), EC and their various combinations [4]. EC techniques have been successfully applied in many areas including: engineering design optimisation, manufacturing system, process control, medical-diagnosis, and simulation and communication systems.

This paper attempts to bridge the gap between theory and practice by exploring the features of classical optimisation algorithms, EC techniques and characteristics of real world problems. The paper surveys recent innovative EC applications in the manufacturing industry in the following areas: metal forming, chemical industry, paper industry, scheduling and process planning, CAD/CAM environment and manufacturing related industries. For each application area, the paper describes the general domain problem, common issues, current trends, and the improvements generated by adopting the EC strategy. All the application problems that are discussed in this paper are used/piloted in the specific industry. The paper concludes with an outline of inhibitors to industrial applications of optimisation algorithms.

Papers from related IEEE Transactions using IEEE Xplore and other related journals over the last 5 years are reviewed here to show reported industrial innovations using EC. The paper is organised as follows: Section 2 describes the algorithmic approaches to optimisation. EC and its components are described in Section 3. The characteristics of real world problems are explored in Section 4. Industrial applications of EC techniques are discussed in Section 5, Section 6 outlines the inhibitors to industrial applications of EC-based techniques, and the concluding remark is given in Section 7.

## 2. Algorithmic approaches to optimisation

This section explores the features of classical optimisation algorithms in order to appreciate the difficulties recent EC-based approaches have to overcome to be successful. Literature suggests a number of optimisation techniques for solving optimisation problems. These techniques can be classified into two broad categories: classical and evolutionary.

Most classical algorithms use a point-by-point deterministic procedure for approaching the optimum solution. Such algorithms start from a random guess solution. Thereafter, based on a pre-specified transition rule, the algorithm suggests a search direction, which is often arrived at by considering local information. A uni-directional search is then performed along the search direction to find the best solution. This best solution becomes the new solution and the above procedure is continued for a number of times. Fig. 1 illustrates this procedure. Algorithms vary mostly in the way the search directions are defined at each intermediate solution [5].

Classical optimisation methods can be classified into two distinct groups: direct methods and gradient-based methods. In direct search methods, only the objective functions and the constraint values are used to guide the search strategy. Some examples of these methods are the simplex search method [7], Hooke–Jeeves pattern search method [8] and Powell's

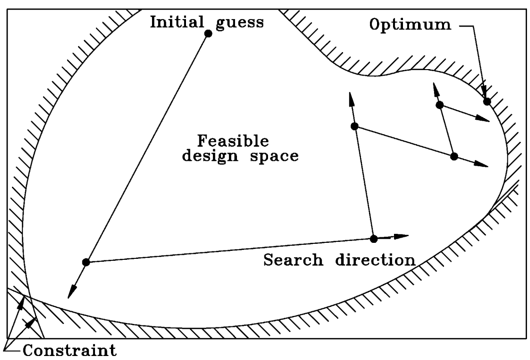


Fig. 1. Classical optimisation algorithms (adapted from Deb [6]).

conjugate direction method [9]. On the other hand, the gradient-based methods use the first- and/or second-order derivatives of the objective functions and/or constraints to guide the search process. Some examples of these methods are Cauchy's (steepest descent) method [10], Marquardt's method [11] and conjugate gradient method [12]. Since derivative information is not used, the direct search methods are usually slow, requiring many function evaluations for convergence. For the same reason, they can also be applied to many problems without a major change in the algorithm. On the other hand, gradient-based methods quickly converge near an optimal solution, but are not efficient in non-differentiable or discontinuous problems. In addition, there are some common difficulties with most classical direct and gradient-based techniques, as mentioned below [5]:

- The convergence to an optimal solution depends on the chosen initial solution.
- Most algorithms tend to get stuck to a sub-optimal solution.
- An algorithm that is efficient in solving one optimisation problem may not be efficient in solving a different optimisation problem.
- Algorithms are not efficient in handling problems having a discrete search space.
- Algorithms cannot be efficiently used on a parallel machine.

The above-mentioned drawbacks of classical optimisation techniques have led to the growth of research in the field of EC. The EC techniques can handle most of the drawbacks of classical algorithms, and the characteristics of the EC techniques, especially their robustness, make them suitable for dealing with the features of a variety of real-life optimisation problems [13]. However, all these advantages of EC techniques come at the cost of their high computational expense.

## 3. Evolutionary computation

Evolutionary computing techniques are robust and offer exceptional adaptive capabilities to handle non-linear, highly dimensional and complex engineering problems. They do not require explicit knowledge of the problem structure or differentiability, and have ability to provide multiple near-optimal solutions to even ill-defined problems from the expert. The origins of EC can be traced back to 1950s [14,15], since then several evolutionary algorithms have been proposed, some of which includes: GAs [16,17], GP [18], EP [19] and ES [20,21]. Fig. 2 illustrates the taxonomy of EC technique and their main components. Many of the algorithms share the same basic template but follow different paradigms [22]. Each approach starts with a random or semi-random population of candidate solutions for the problem at hand, and evolves it iteratively applying a set of stochastic operators known as mutation, recombination and selection. The resulting process tends to find global optimal solutions to the problem given sufficient time. A brief description of these EC components is given as follows.

### 3.1. Genetic algorithms

GAs are adaptive methods used to solve search and optimisation problems, based on genetic processes of biological organisms. GA was developed in the 1970s by John Holland and students at the University of Michigan. Their aim was to simulate adaptive process of natural systems and to develop artificial systems that retain features of natural systems. The canonical

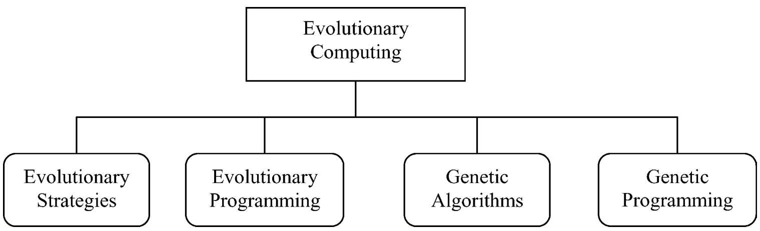


Fig. 2. Taxonomy of EC components.

form of the GA encodes each candidate solution to a given problem as a binary, integer, or real-valued string, referred to as chromosome. GAs simulate the genetic state (chromosomes) of a population of individuals using recombination operators (crossover and mutation). Crossover exchanges genetic material between two parents. Mutation flips a bit in a chromosome. Mutation is conducted to prevent the premature convergence of the design variables that is all the bit structures of strings in the mating pool become identical in an early stage of evolution. However, mutation may also play a role detrimental to achieve fast convergence [23]. Each individual is evaluated and fitness assigned in proportion to the value of the objective function for the individual. New individuals created by these operators are selected on the basis of their fitness for the next generation.

### 3.2. Genetic programming

Genetic programming (GP) uses the principles of genetics and Darwinian natural selection to evolve computer programs [18]. GP is largely similar to GA. The notions of mutation (crossover) and fitness are essentially the same. The main difference is the representation of the solution. GA creates string of numbers that represent the solution usually associated with variables of a function while in the GP the individuals are tree structure programs and the genetic operators are applied to the branches or the tree nodes. GP uses LISP programs as the search space but other programming codes can also be adopted [24].

### 3.3. Evolutionary programming

EP was originally introduced to operate on finite state machines and the corresponding discrete representations. However, recent versions have focused on continuous parameter optimisation problems [25]. As a result EP is similar to ES. In EP paradigm, each parent individual in the population generates an offspring by mutation (asexual reproduction). Mutations are typically performed with uniform probability and are often adaptive. After evaluating the offspring a variant of stochastic tournament selection selects $\mu$ best individuals from the union of the parent and the offsprings, i.e. a randomised $(\mu + \mu)$ selection is adopted. The best individual is always retained ensuring that once an optimum is found it cannot be lost. EP is a mutation–selection algorithm that uses no recombination.

### 3.4. Evolutionary strategies

Evolutionary strategies (ES) were developed by Rechenberg [20] and Schwefel [21] and were originally applied in hydrodynamic optimisation problems. There are two main ES strategies: $(\mu, \beta)$ ES and the $(\mu + \beta)$ ES [21], where $\mu$ parents produce $\beta$ offsprings using recombination and mutation operators. In the $(\mu, \beta)$ ES, the best $\beta$ offspring are allowed to survive and the parents are replaced with the offspring. As a result the parents do not take part in the next generation. In contrast, the $(\mu + \beta)$ ES allows both the parent and offsprings to survive. The $(\mu + \beta)$ ES is based on an elitist strategy while the $(\mu, \beta)$ ES is not. The recombination operator plays an important role in the ES. Each individual has a mutation operator with adaptive mutation rate.

The different algorithms as described above are required to tackle the vast range of real world problems. According to the no free lunch (NFL) theorem for optimisation [26], no single method can outperform all the other methods on all problems, and since no single method is suitable to solve all classes of problems it follows that different methods are required to solve all classes of problems. In the section that follows, characteristics of real world problems are explored to show the challenges faced by different algorithms in solving real world problems.

## 4. Characteristics of real world problems

Most real world manufacturing problems are usually large scale, highly dimensional, non-linear and highly uncertain involving interaction with engineers and highly skilled operators that control the process plants. These problems are also characterised by chaotic disturbances, randomness and complex non-linear dynamics [1]. An understanding of the features of real world problems supports algorithmic development to include wider applications of EC-based algorithms in industry. Therefore some of the main characteristics of real world

Table 1
Features of real-life engineering optimisation problems [13]

| Classification schemes | Features |
|---|---|
| Number of parameters | Highly dimensional |
| Existence of constraints | Constrained |
| Number of objective functions | Multi-objective |
| Nature of objective functions | Hybrid |
| Separability of functions (for quantitative and hybrid problems) | Inseparable |
| Dependence among variables | Independent- and dependent-variable |
| Nature of quantitative search space | Unknown search space<br>Multi-modal |
| Nature of equations involved (for quantitative and hybrid problems) | Linear, non-linear, geometric and quadratic |
| Nature of design variables | Static and dynamic |
| Permissible values of design variables | Hybrid |
| Expert knowledge | Imprecise, uncertain and stochastic |
| Nature of qualitative search space | Discontinuous and highly multi-modal |

optimisation problems are outlined below and are summarised in Table 1 [13].

### 4.1. Number of variables and the ‘curse of dimensionality’

Most real world problems quite often have very large number of variables. Deb [12] explored a real world problem with over 500,000 variables. Large number of variables also implies exponential increase in interactions [27] among those variables. This effect was described by Bellman [28] as the ‘curse of dimensionality’. It is a non-trivial task to model the problems with such effect. These variables can also be either discrete, continuous or a combination of the two [29].

### 4.2. Complex search space

The complexity of the search space is due to the size and nature of objectives and constraints. Objectives and constraints can be single or multiple, however most real world problems tend to be multiple. The multi-objective nature results in conflicting objectives. The presence of multiple constraints significantly affects the performance of any optimisation algorithm including evolutionary search methods [30], making it hard to find optimal solutions. The objective and constraint functions can be rough, severely non-linear, discontinuous or even ill-defined / undefined in some regions of the variable space [31]. The existence of several local extrema (multi-modality) is also quite common. Search space can be known or unknown but most real-life search spaces are unknown due to the combinatorial explosion of the variable interactions. The search space can be quantitative ($Q^T$) or qualitative ($Q^L$) in nature. $Q^T$ describes the behaviour of the problem using suitable numerical variables. However, this can be incomplete, and there are certain behaviours (e.g. colour) that are not easily described using analytical means. In such cases human based reasoning can be formulated into rules of $Q^L$ models to express such behaviours. $Q^L$ models can also be combined with $Q^T$ models.

### 4.3. Adaptability

Real-life problems can be dynamic in nature. The problem faced at any moment may have diverged significantly from the problem anticipated originally. Traditional approaches often have difficulty coping with this problem as calculations must be restarted from the beginning if any variable in the problem changes [1]. The aim in optimising real world problem is to generate good solutions, ‘satisficing’ [32] quickly and attainment of optimum is much less important.

### 4.4. Hierarchical

Hierarchical nature of real-life problems implies the ‘pecking order’ of the subsystem’s problem space. Many real-life optimisation problems are hierarchical in nature with multi-level problem space [33]. A two level problem is described here where the subsystem is considered as the low-level problem and the system level (which acts as the coordinator) is considered as the high-level problem. The low-level search provides insight into the intricacies of subsystem while the high-level search aggregates the overall search behaviour with respect to the system requirement. Optimisation search between both levels could be fully cooperative, fully conflicting or cooperative/conflicting [34]. For example, at the managerial level, managers need to decide which products to process and which operating route to adopt while at the process level, the operators have to determine the detailed operating conditions of each process units.

### 4.5. Human interaction

Traditional optimisation method relies on trial and error as the main search and sort method. Engineers, designers and in general, practitioners all influence the knowledge that is brought to solve complex real-life problems. This influence is qualitative in nature [3] and forms an integral part of the engineering problem solving process. Design decisions are still being made using subjective judgements based on the design experience and the 'rule of thumb' derived by accumulating years of experience [35]. Despite the high level of scientific knowledge that exists for some engineering problems, the human interaction element also leads to a high source of uncertainty and imprecision concerning the application of knowledge to engineering problems. There are challenges in developing near-closed form approaches that incorporate both the qualitative and quantitative knowledge required to solve complex engineering problems.

## 5. Evolutionary computation in industry

EC techniques are gaining increased interest in industry. Literature reveals a number of real-life applications of optimisation algorithms, especially EC. Fig. 3 presents a pie chart showing the proportion for different EC techniques in the sample of publications reviewed in this paper. The sampling is not exhaustive and is just intended to give a snap shot of the research efforts for various EC techniques.

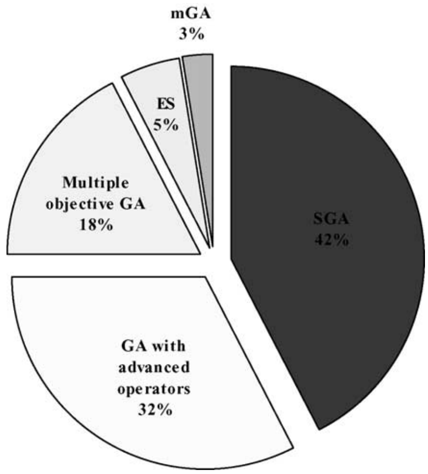


Fig. 3. Proportion of EC techniques used in publications surveyed in this paper.

The chart in Fig. 3 is fairly intuitive, with each sector representing the percentage proportion of different EC techniques used in the publications reviewed in this paper. For example, out of the total application papers reviewed 42% fall in the category of SGA applications. The chart also shows that there is a comparable research effort in the area of GA with advanced operators (advanced GA) and the SGA applications. Also 18% of surveyed papers contain multi-objective problems. These statistics indicate that 50% (18 + 32) of the GA applications reported in this paper solves some complex real world problems. However the chart highlights that there is a significant research effort to develop more advanced EC operators to deal with some of the limitations of the SGAs when dealing with real-life problems.

This is increasing the research interest to develop more advanced operators for obtaining better solutions at reduced computational cost. Since real world problems are multi-objective in nature, more applications are also emerging in this area. Fig. 3 also indicates EC applications using ES and mGAs.

This section reviews EC applications in the following areas: metal forming industry, paper industry, chemical industry, scheduling and process planning, engineering design optimisation and related manufacturing applications. The problems described in this review are real world application problems having the characteristics described in Section 4. The survey describes the engineering problem under study, the EC technique adopted in solving the problem and the improvements achieved by adopting the EC technique.

### 5.1. Metal forming industry

The metal forming (MF) process is a complex operation requiring a simple geometry to be transformed into a complex one. The main goal of optimisation in MF is to produce sound products through optimal process design, since the process material and die variables significantly influence the process. Classical approaches such as trial and error are tedious, ill-structured, time consuming and costly. Dynamic programming can handle continuous and discrete variables, but is limited since the MF process normally involves large amount of process variables with wide range of values that may be active in the optimisation problem. Also, derivative based approaches are not suitable since the objective function may possess multiple stationary points. Several authors [36,37] have shown that the GA-based approaches can be used to deal with these complex real world problems. An outline of EC applications in MF industry reported in this paper is given in Table 2.

In rolling system optimisation, a number of studies have been reported using GA for optimal design [44,45], for multi-objective optimisation problems [40,41,46], and recently for dealing with quantitative and qualitative search space [49,50]. These approaches have been shown to offer a more structured approach to process optimisation problems. They also offer the benefit of cataloguing the optimal solutions for future re-use. This can save design time and effort for future problems. However the main problem experienced using GA in this environment is due to the expensive function evaluations. Since objective functions are often analytically unknown, function evaluations can only be achieved through costly computer simulations. The slow convergence criteria to near-optimal solution with very small tolerance accompanied with the large population of solutions required for the evolutionary process result in expensive evaluations. Therefore, several approaches have been developed to address this drawback.

Oduguwa et al. [47,49,51] developed meta-models as surrogate to expensive computational FE simulations. The meta-model were used for function evaluations in the optimisation of rolling design problems. This reduced the computational cost significantly. Several other authors have adopted mGA as embedded optimisers within expensive computational simulation software to reduce the computational cost. The mGA attempts rapid convergence using a small population size which typically consists of five individuals created using selection and crossover. Several authors have also applied mGA for attaining optimal design. Deb et al. [38,39] conducted an experimental study using mGA to identify the optimal setting of the burners in continuous casting. Roy et al. [43] proposed adaptive mGA for optimal design of process variables in multi-pass wire drawing. In spite of the small population, the authors reported expensive computation runs required for convergence to optimal solutions with very small tolerance. Chung and Hwang [23] addressed this problem by developing a modified mGA using gray coding for representing the chromosome. The approach was applied for optimal process design for non-isothermal metal forging. The authors suggested that more effort is required to develop sound convergence.

### 5.2. Paper industry

The paper and pulp mill is a complex continuous production process where shutdowns and disturbances are propagated throughout the plant and influence the whole mill. It represents a highly non-linear system that has many different operating points. The resulting problem is a production scheduling problem with the following features: mixed integers, multiple objectives, multiple constraints and high dimensionality. Several gradient-based techniques including quasi-Newton and sequential quadratic programming [52] have been applied in this area. However, these techniques are known to get stuck in local optimum. Recently EC-based approaches have been applied to several paper processing problems. The main problems experienced using the EC-based techniques are due to the difficulties of achieving feasible solutions and the slow computational speed compared to gradient approaches (because of the large number

Table 2
Types of metal forming problems

| Applications | Characteristics | EC component | Reference |
| --- | --- | --- | --- |
| Continuous casting (CC) | A real coded GA for optimal setting of process parameter setting in CC of steel | Advanced GA | [36,37] |
| | Parameter optimisation of re-heat furnaces for bloom delivery (using SGA, differential evolution (DE) and mGA) | SGA, DE, mGA | [38,39] |
| | Pareto converging GA(PCGA) for attaining optimal CC velocity | PCGA | [40] |
| | SGA for optimal setting of process parameter in CC of steel | SGA | [41] |
| Forging | Micro GA is used for optimal process design of non-isothermal metal forging | mGA | [23] |
| | Inverse elitist evolutionary search for optimal design of metal forging | Advanced GA | [42] |
| Extrusion | Inverse elitist evolutionary search for optimal design of extrusion | GA | [42] |
| Wire drawing | Optimisation of process variables in multi-pass wire drawing | mGA | [43] |
| Rolling | SGA is applied for the optimisation of roll profiles in strip rolling | SGA | [44] |
| | SGA is used for optimal design of tandem cold rolling mills | SGA | [45] |
| | Two objective GA for pass optimisation of schedule in cold rolling of silicon steel strips | Advanced GA | [46] |
| | Multi-objective optimisation of rod product design | NSGAII | [47] |
| | A qualitative and quantitative design optimisation approach to minimise both load and shape in rod rolling product design | NSGAII, FL | [48] |
| | Three objective optimisation of elongation, strain distribution and shape in rod rolling | NSGAII, FL | [49] |

Table 3
Types of Problem in the Paper Industry

| Applications | Characteristics | EC component | Reference |
|---|---|---|---|
| Paper making process | A GA multi-criteria approach with constraint-handling techniques for optimisation of energy cost and production rate of a pulp and paper mill | Advanced GA | [53] |
| | Optimisation of a paper making process | Advanced GA | [54,55] |

of solutions required for the evolutionary process). Several approaches have been developed to address this drawback. An outline of EC applications in the paper industry reported in this paper is given in Table 3.

Santos and Dourado [53] developed a genetic algorithm based approach and applied it to optimise the energy cost and production rate of a pulp and paper mill. The authors adopted a GA multi-criteria approach with constraint-handling techniques. A genetic algorithm approach was adopted because optimal solutions to the problem were not considered to be achievable using traditional methods. The solution strategy aims to preserve feasible solutions using specialised crossover and mutation operators for transforming one feasible solution to another. A uniform crossover operator was used and the mutation phase consists of a set of four strategies: uniform, boundary, non-uniform and exchange mutations. A stochastic universal sampling method was adopted as the selection operator, and this was used in the context of generational reproduction (this is where the whole population is replaced in each generation). A fitness-sharing scheme to distribute the population in niches within the search space with restriction in mating was also adopted. Each chromosome was coded as a sequential real multiple real variables. The authors concluded from the obtained results that GAs have strong potential for solving these classes of problems. Borairi et al. [54,55] used GA in conjunction with neural network for optimisation of a paper making process. A common learning feature of the NN is the back-propagation (BP) technique that is used on the multiplayer feed-forward NN. A typical problem with this approach is that as the network complexity increases the performance of the BP decreases rapidly. GA is proposed as an alternative to the training of the NN weights. A simple GA approach was used to find optimal structure of the neural networks as well as to train the weights. GA was able to find the global minimum for the error.

### *5.3. Scheduling and process planning problems in production industry*

Scheduling is an important aspect of the economy defined as the allocation of resources over time to perform a collection of activities [63]. The scheduling problems are normally difficult to solve due to various constraints, complex product structures with many levels of assembly and large number of potential sequences. Mathematical programming techniques such as mixed integer programming techniques, mixed integer non-linear techniques provide near-optimal solutions, but they tend to be suitable to small scheduling problems [56]. In most GA applications [64] to the scheduling problem, the sequence of the entire schedule is encoded in the chromosomes. Such a method can render the SGA unusable since a large proportion of offspring can become infeasible after the genetic operations. To overcome these problems, a large part of GA research in this area has focused on developing approaches that improve the feasibility of chromosomes in the population. An outline of EC applications for the scheduling and process planning problems reported in this paper is given in Table 4.

Engell and coworkers [57] applied an augmented GA to a real world scheduling problem in the polymer industry. The author applied problem specific knowledge to generate feasible individuals for initialising the population. The chromosome consists of integer and real parts, and two mutation methods (low and high) with crossover were used for the two parts to maintain diversity. Infeasible offsprings detected after the genetic operation are simply deleted from the population. Hicks and coworkers [56] proposed a repair strategy in their GA application for scheduling complex product in Capital Goods Company. The

Table 4
Scheduling and process planning problems in production industry

| Applications | Characteristics | EA component | Reference |
|---|---|---|---|
| Scheduling | SGA for scheduling complex products (using repair strategy to obtain feasible solutions) | Advanced GA | [56] |
| Polymer | Optimal scheduling of polymer batch plant using augmented GA | Advanced GA | [57] |
| Assembly planning | Optimisation of assembly sequence planning using wave of assembly principle to maintain feasible individuals | Advanced GA | [58] |
| District heating and cooling | Optimal operational planning of district heating and cooling using GA with linear programming | Advanced GA | [59] |
| Production planning | Optimal production planning to meet time-varying stochastic demand (using GA with dynamic programming) | Advanced GA | [60] |
| Network design | Studies in network design, multi-stage planning and minimum spanning tree problems using permutation encoding method for feasible solutions | Advanced GA | [61] |
| | Global topological design of hierarchical network | Advanced GA | [62] |

repair procedure was a four-stage process to identify and rectify infeasible schedules. Marian et al. [58] developed a more elaborate approach (based on Huygens' wave propagation) [65] for an assembly sequence planning problem. The principle is based on determining the components to be added to the partial assembly at the next stage. The wave model of assembly was used to form a guided search algorithm which in turn served to generate feasible assembly sequences. Assembly sequence was encoded in the chromosome where the chromosome is a permutation of product components. The authors reported that the size and space required to store the representation were greatly reduced.

Sakawa et al. [59] proposed a GA for district heating and cooling (DHC) plant using chromosome encoded with integer genes and fitness using linear programming problem to decide feasible solutions. In the proposed approach, continuous variable are represented in the individual by linear programming problem while integer strings corresponding integer variables are determined by the GA. An initialisation strategy based on preserving feasible solutions for the initial population was adopted, since random evolution quite often results in few feasible individuals. The branch and bound (BB) method was used to find optimal solution to the sub-individual problems at initialisation. The fitness of each individual is calculated by solving the linear programming problem using the simplex method and linear scaling. The reproduction adopts a combination of expected value selection and elitist preserving selection. A one-point crossover operator is applied, and the mutation operator is Gaussian distribution based on linear programming relaxation [66] and uniform distribution. Yokoyama and Lewis III [60] also proposed an approach to preserve feasible solutions. Their approach was a two level solution method using GA and dynamic programming (DP) for production planning problem. At the upper level GA was used to determine decision variables that minimise set-up cost and the production cost while at the lower level, the DP was used to determine the decision variables for the production quantities.

Similar encoding efforts are also reported for multistage process problems and network design problems. Gen et al. [61] conducted experimental studies on these problems and adopted various permutation encoding methods. In the coding scheme, the position of a gene is used to indicate the stage and the value of the gene is used to indicate the state at that stage. The author reported that the coding scheme was able to generate feasible individuals through genetic operators with the initial population generated randomly. In telecommunications network design problem, Cortes et al. [62] guaranteed feasible solutions in GA using problem specific heuristics. The network problem was augmented with fictitious node rules to establish feasible communication links. Constraints were used to impose communication over the hubs.

Most of the approaches outlined above attempt to solve the feasibility problem caused by concatenating the decision variables into the chromosomes. It is arguable if these approaches could address the increasing need to deal with more problem features in the process optimisation problem. This results in the need to develop more advanced techniques to preserve feasible solutions. Alternative approaches can be explored by adopting some of the decomposition strategies employed in mathematical programming for exploiting the natural decomposition of individual units. This could have a higher probability of generating feasible individuals. However, approaches would have to be developed to coordinate the search for various independent optimisation processes.

### 5.4. Chemical industry

Chemical plant contains multiple chemical engineering devices such as pumps, distillation columns and chemical reactors. This forms a complex web of interconnecting material, heat and information streams designed to perform a chemical process during which raw materials are converted into desired products. Optimisation of the chemical plant aims to modify the process structure of the operation parameters in order to find the global optimum to perform a certain chemical task. The chemical engineering problem is a complex real world optimisation problem with most of the features described in Section 4. The problem is quite difficult to solve due to the non-linear system dynamics and constraints on the control and state variables [73,74]. Also global optimum is difficult to determine because of insensitivity of the performance index to the control profiles. Some classical approaches for solving the chemical problem include mixed integer non-linear programming techniques [75], heuristical branch and bound techniques [76], hierarchical decomposition approaches [77] and dynamic programming [74]. These general purpose algorithms do not generate results within reasonable times [57]. Gradient-based techniques are known to converge to local optima, and they are not suitable for non-differentiable problems [78]. Application of EC techniques to this class of problem is growing, but has found limited application in chemical engineering. In this paper, the application of evolutionary algorithms in this area is classified into two categories: the first is based on the standard evolutionary algorithms (EAs) and the second is based on EA efforts using advanced operators. An outline of EC applications for the chemical industry reported in this paper is given in Table 5.

Several authors have applied classical GA in chemical industry [67–69]. Harris et al. [67] presented a simple GA-based inversion procedure to determine the optimum rate coefficient for chemical schemes based upon limited net species production data. Grujicic et al. [69] coupled a simple GA with chemical vapour deposition model to determine the process parameters which maximise the carbon nanotubes yield while minimising the amount of deposited amorphous carbon. Rajesh et al. [68] performed a multi-objective optimisation on the unit performance of an industrial hydrogen plant. The authors encoded seven variables in the chromosomes to determine the optimal operating condition for improved unit performance. The main reason for classical GA's success in such complex environments is the size of the problem. Only the necessary part of the problem is encoded in the chromosome. The problem size

Table 5
Types of problem in the chemical industry

| Applications | Characteristics | EC component | Reference |
|---|---|---|---|
| Chemical kinetics | Optimisation of production rates for various conditions | SGA | [67] |
| Hydrogen plants | Optimal operating conditions for improved unit performance | NSGA | [68] |
| Chemical vapour deposition | Optimisation of process parameters for carbon nanotubes | SGA | [69] |
| Calorimetric reactor | GA combined with a local convergent method to improve productivity of reaction mixture in a calorimetric reactor | Hybrid GA plus local convergent method | [70] |
| Chemical plant design | A non-standard evolutionary strategy for the parameter optimisation of a well documented hydrodealkylation (HDA) process in chemical plant design | Advanced ES | [71] |
| | A graph-based EA for structural optimisation of chemical engineering plants | | [72] |

encoded in the chromosomes range from 5 to about 19 [67]. Since the number of parameters is relatively small, there is a higher chance of obtaining feasible solutions. This approach can be considered appropriate with respect to the sparsity effects principle [79], where it is assumed that the process problems are dominated by the chosen parameters and the interactions are negligible. It is interesting to note that most authors suggested the need to include more parameters for a complete optimisation analysis. Motivated by the need for the synthesis of a larger scale process problem, several other research efforts have focused on developing advanced EA operators to work in such complex environments.

In contrast to the GA applications outlined above, representing the chosen parameters in the chromosome for larger scale problems could result in infeasible offsprings after the genetic operations. Therefore, several researchers have focused on developing approaches that preserve feasible solutions. Balland et al. [70] developed a GA-based approach to improve productivity of reaction mixture in a calorimetric reactor. In an attempt to encourage feasible design solutions, the author combined GA with a local convergent method to obtain correct initialisation points for the convergent method. Emmerich et al. [71] in an experimental study presented a non-standard evolutionary strategy for the parameter optimisation of a well documented Hydrodealkylation (HDA) process in chemical plant design. The author developed ES operators based on maximum entropy principle of mutation to guarantee universal applicability. The authors in [72] later developed a graph-based EA for structural optimisation of chemical engineering plants. A graph representation was adopted to closely mimic the original problem formulation. This facilitated the incorporation of expert knowledge and heuristics.

The trend of EC in the chemical industry suggests continuity in the use of SGA to address optimisation problems involving objectives such as profit and production rate of the chemical plant. However, there is increasing interest to adopt advanced ES algorithm to deal with more complex problems. There are also efforts for integrating expert knowledge and heuristics into the genetic operators.

### 5.5. Curve/surface optimisation

Traditionally, design optimisation is an iterative process of creating a model, using a CAD system, carrying out some analyses which give indications of how to improve the design and then modifying the model and repeating the process. This manual process contributes significantly to lengthening the design cycle and depends critically for its success on the skill of the designer. There is thus a need to develop a process, which requires less intervention and can be carried out within the CAD/CAM environment. Much of the relevant research in design optimisation within the CAD environment has focused on the area of surface optimisation. A number of these applications have used classical optimisation techniques, such as Powell's conjugate direction method [9] and conjugate gradient method [12], for the optimisation of surfaces. Some examples of these applications are discussed below. An outline of EC applications for curve/surface optimisation reported in this paper is given in Table 6.

Table 6
Curve/surface optimisation

| Applications | Characteristics | EC component | Reference |
|---|---|---|---|
| Curve/surface optimisation | A study on genetic shape design | Advanced GA | [80] |
| | Optimal surface reconstruction from digitised point data using computational intelligence methods | Computational intelligence methods | [81] |
| | Surface optimisation within a CAD/CAM environment using GAs | Advanced GA | [82,83] |
| Algorithm development for curve/surface optimisation | Development of EC techniques for handling inseparable function interaction in real-life engineering design optimisation problems | GRGA/GAVD | [13,84,85] |
| | Evolutionary computing techniques for handling variable dependence in engineering design optimisation problems | GRGA/GAVD | [13,84,85] |

The FANGA techniques developed at Saab-Scania [86] perform surface optimisation by, first, developing a set of simple measures which allow a surface to be characterised at sample sections in terms of a few angles. These are then used in a standard optimisation technique to compute the necessary changes to the parameters of the surface model to produce the desired properties. The technique is used to optimise and assess the quality of surfaces used in their aircraft and car body design. The approach is an extension of the earlier development of FORMELA at Saab-Scania, given in [87]. This enables a surface to be refined by the response of a set of angles and directions, which are easily computed from the surface model. These are then used in a standard optimisation technique to compute the necessary changes to the parameters of the surface to produce the required qualities. Earlier research in this area was targeted at improving the design process for car body shapes [88]. Here the surfaces are analysed by examining reflection lines, which are first computed. From an analysis of a set of reflection lines a new set of lines are computed, and the surface adjusted to conform to them.

Furthermore, a comparative study [89] of three popular CAD/CAM systems, CATIA, I-DEAS and UNIGRAPHICS, demonstrates that all have some limited built-in optimisation capabilities provided by classical optimisation techniques. For example, CATIA has a generative part optimisation module that allows a constrained optimisation process to be run on a design stored within the system. The system also automatically parameterises splines so that they may be subsequently updated, and thus allows the performance of optimisation on free-form geometry in 2D or on the surfaces of prismatic solids. However, the main limitation of the system is that only one type of optimisation algorithm is available to tackle all problems. Both I-DEAS and UNIGRAPHICS provide similar facilities to define and iteratively modify parts containing complex free-form surfaces, but all share the same limitations as the CATIA system in terms of the available optimisation methods.

These methods are perhaps not as adaptable and robust as others that have been developed using evolutionary computing techniques, examples of these can be found in [80,81]. This latter work attempts the generation of smooth surfaces from digitised point data using several techniques such as, EP, GAs, ESs and GP. The most significant advantage of using evolutionary computing techniques lies in the gain of flexibility and adaptability to the task in hand, in combination with robust performance and global search characteristics [90]. The evolutionary-based optimisation techniques use a population of solutions in each iteration, instead of a single solution. This enables them, in principle, to identify multiple optimal solutions in their final population. The work by Weinert et al. [81] presents three solutions to the problem of reconstructing smooth surfaces using triangular tiles. It is observed that non-deterministic methods represent new ways of optimising triangulations for those surfaces. Examples of other methods that improve quality of surfaces (smoothness and fairness) are those proposed by Higashi et al. [82] for the Toyota Technological Institute [91]. None of these techniques appear to aim to develop an integrated optimisation capability for dealing with a range of designs within existing CAD software.

Roy et al. [89] and Jared et al. [92] specifically addressed the issue of enhancing optimisation capabilities within a CAD/CAM system. The research defines a concept of ‘Flexible Optimisation within a CAD/CAM System’ and develops case studies based around a 3D solid model for multi-objective optimisation and a curve/surface model for constrained optimisation. Mussa et al. [83] has further explored the area of Flexible Optimisation within CATIA CAD/CAM software for surface optimisation. However, all the above-mentioned research activities are limited in their scope since the genetic algorithms that they provide can deal with only simple curve/surface optimisation problems. Real-life curve/surface optimisation problems possess a number of characteristics, such as high dimensionality and high degrees of interaction among the decision variables, which make it difficult for the genetic algorithms to produce satisfactory results. These characteristics highlight the need for sophisticated optimisation techniques that can deal with the complexities of real-life curve/surface optimisation problems. In a recent project FLEXO (Flexible Optimisation within CAD/CAM Environment), Roy et al. [84,85] propose two evolutionary-based optimisation algorithms that are capable of dealing with the primary features of curve/surface optimisation problems: multiple objectives, constraints, high dimensionality and high degrees of interaction among the decision variables. These two algorithms are GRGA (generalised regression genetic algorithm) and GAVD (genetic algorithm for variable dependence) [13]. These evolutionary computing techniques enable the FLEXO framework to deal with a variety of optimisation problems within CAD/CAM environment, with little user intervention.

### 5.6. Related manufacturing applications

This section surveys applications of EC for solving related manufacturing problems. An outline of the application areas surveyed in this paper is given in Table 7.

Carrillo-Ureta et al. [93] implemented a genetic algorithm approach for the optimal control of beer fermentation. Alcoholic fermentation process control is based on the biochemical properties of the fermentation broth and process operating parameters. Traditional trial and error methods have been used in the past to find the optimal process conditions. A more robust genetic algorithm approach was presented to adjust the temperature profile of the mixture during a fixed period of time in order to reach optimal operating process levels. The results achieved promising temperature profiles [93]. In operating parameter optimisation of stereolithography, Cho et al. [94] applied simple genetic algorithm to minimise part dimension error of the process. Stereolithography is a photopolymerisation process for rapid prototyping (RP) that utilises a laser beam to selectively draw or print cross-sections of a model on a photo-curable resin surface. In the current process, the build inaccuracy and distortion inhibits wider process applications. The GA application was shown to obtain smaller part dimensional error with proposed wider applications in other RP areas. Covas et al. [95] applied a simple genetic algorithm to generate better inputs for the single screw extrusion process. A single and multi-objective optimisation of the polymer extrusion process was conducted. The approach was successfully used to set optimal operating windows fulfilling prescribed criteria [95].

In truss structure optimisation, several authors have applied genetic algorithm to different categories of the truss problem. Cerrolaza and Annicchiarico [96], and Deb and Gulati [97] applied a classical GA and real coded GA, respectively, to minimise the weight of 3D-truss structures. Chen and Lin [98] applied simple genetic algorithm to find the optimum design spaces for topology optimisation. Hamza et al. [99] applied a classical genetic algorithm to minimise the overall weight of a plane truss design for application of industrial and commercial clear-span buildings. The optimisation problem involved sizing (restricting the choice of truss members to available discrete set), nodal positions of the truss members and topology (connectivity of the truss members). Classical genetic algorithm was adopted in most of the cases. The search was terminated when the given number of objective function evaluations is reached. The author suggested that GA struggles from high dimensionality and it becomes insufficient to achieve enough diversification. Also GA's performance was inhibited due to complex constraints. In spite of the identified weaknesses, GA was still able to achieve better designs with good diversification [99]. Barone et al. [100] applied a multi-objective evolutionary strategy to determine the geometry and optimal operating settings for a crusher circuit for ore processing for

Table 7
Related manufacturing applications

| Applications | Characteristics | EC component | Reference |
|---|---|---|---|
| Beer fermentation | Control of beer fermentation to reach optimal operating process levels | SGA | [93] |
| Stereolithography | Optimal operating parameter optimisation of stereolithography | SGA | [94] |
| Polymer extrusion process | Single and multi-objective optimisation of the polymer extrusion process was conducted | Advanced GA | [95] |
| Truss structure | Optimisation of different categories of truss problem | SGA | [96–99] |
| Ore processing | Geometry and optimal operating settings for a crusher circuit for ore processing for maximising the size of the product | ES | [100] |

maximising the size of the product. The results obtained showed promising financial benefits.

Application of GA's to a variety of real world engineering problems most notably the truss structural optimisation problems suggests a wide use of classical GA for parameter optimisation problems. Despite the limitations of SGA's, results demonstrate robustness and better solutions compared to traditional techniques.

## 6. Inhibitors to industrial applications of EC-based optimisation algorithms

Roy et al. [85] carried out a survey of existing literature in the area of real-life optimisation in order to analyse the state-of-the-art evolutionary-based optimisation algorithms and their real-life applications. This was complimented by an industrial survey in which the designers were interviewed using a semi-formal questionnaire as a support tool. This survey enabled the identification of the features of real-life optimisation problems, current status of optimisation in industry and inhibitors to industrial applications of optimisation algorithms. Since it was observed that there is a lack of algorithmic approach for optimisation in industry, the survey looked at the applications of all optimisation algorithms, rather than just evolutionary-based techniques [85]. Despite the immense potential of EC-based optimisation algorithms, it was observed by Roy et al. [85] that they are not popular in industry as day-to-day design tools. The main inhibitors to the industrial applications of these optimisation algorithms are outlined below [13]:

- The features of real-life optimisation problems, such as the presence of interaction among decision variables, create challenges for EC-based optimisation algorithms that are currently in use in industry. This discourages the industry from adopting these algorithms. This is particularly true for those industry that deal with a wide range of complex designs.
- All EC-based optimisation algorithms work on mathematical models of real-life designs. It is observed that since designers prefer maintaining full control on the design improvement process, they have little faith in the models that are provided to them. This makes them skeptical about the results obtained from the optimisation algorithms. This situation is further worsened by the fact that there is a lack of model development skills among designers in industry. There is also a lack of commercial tools required for carrying out the task of model development.
- Most of the currently available optimisation packages are not integrated within CAD/CAM systems, making their use cumbersome. The designers need to extract the parameters from the CAD/CAM models, feed them to the optimisation packages and bring the optimised parameters back to the CAD system. There are a number of difficulties, associated with this off-line optimisation, which prevent the designers from using the optimisation algorithms. The data transfer often leads to loss of quality and information, which makes the optimisation process inaccurate. This off-line scenario of optimisation also makes designers lose control over the design process. Finally, the inflexible nature of this scenario makes the process iterative and time consuming.
- Another inhibitor to the use of EC-based optimisation algorithms in industry is the important role of designers' skills and experience in the design improvement process. This makes the optimisation task extremely difficult to be modelled and encoded in an algorithmic form. Further, the lack of knowledge of designers in using these algorithms also presents an additional obstacle to their use in industry.
- Each company has its own design improvement process. This process gradually evolves in the company, and hence its people resist the implementation of any new optimisation system and the associated organisational changes. Further, the costs associated with creation, installation and maintenance of EC-based optimisation algorithms discourage their use in industry.

## 7. Concluding remarks

The successful applications of evolutionary computing (EC) suggest that EC will have increasing impact in future. Evolutionary computing is already having an important role on many industrial operations. It provides alternative approaches to traditional analytical problem solving methods (the fundamental limitations of these classical approaches are discussed in Section 2) and it overcomes their shortcomings by its parallel adaptive nature. The main components (GA, GP, EP and ES) of EC were reviewed and a survey some of the successful applications in industry reported.

EC are parallel problem solving algorithms that evolve a random population of solutions iteratively by applying a set of stochastic operators known as mutation, crossover and selection. The primary motivation here is to develop multiple global optimal solutions for the problem at hand. The robustness of EC has been illustrated by surveying recent applications in a variety of sectors within manufacturing industry. In all applications reviewed, the engineering problem involved, the EC technique adopted and the results achieved were described. The survey indicates that there is still a wide use of classical simple GA for solving a broad range of engineering problems, however there is also a realisation that standard canonical form of GA's is not sufficient to tackle the more complex real world problems. For these classes of problems, advanced and quite often 'bespoke' GA operators are alternatives used.

The future of EC appears to be encouraging in terms of its novel application for real world problems. The push for multiple optimal solutions at low cost in an increasingly competitive market environment offers a huge potential for EC applications. In the future an increasing involvement of human interaction within the EC framework is expected, along with the further research interest to deal with high dimensionality and exploit the abundant expert knowledge common in real world problems. This will help the problem solving capability of EC and it will further encourage EC applications for solving industrial problems.

## References


[1] D.B. Fogel, What is Evolutionary computation? IEEE Spectrum (2000) 26–32.

[2] R. Roy, Adaptive search and the preliminary design of gas turbine blade cooling system, Ph.D. Thesis, University of Plymouth, Plymouth, 1997.

[3] V. Oduguwa, R. Roy, Qualitative and quantitative knowledge in engineering system design, Presented at the 17th National Conference on Manufacturing Research, Cardiff University, UK, 2001, pp. 81–86.

[4] A. Tettamanzi, M. Tomassini, Soft Computing: Integrating Evolutionary, Neural and Fuzzy System, Springer-Verlag, New York, 2001.

[5] K. Deb, Optimization for Engineering Design: Algorithms and Examples, Prentice-Hall, New Delhi, India, 1995.

[6] K. Deb, Multi-objective Optimization using Evolutionary Algorithms, Wiley, 2001.

[7] W. Spendley, G.R. Hext, F.R. Himsworth, Sequential application of simplex designs in optimization and evolutionary operation, Technometrics 4 (1962) 441.

[8] R. Hooke, T.A. Jeeves, Direct search solution of numerical and statistical problems, J. ACM 8 (1961) 212–229.

[9] M.J.D. Powell, An efficient method for finding the minimum of a function of several variables without calculating derivatives, Comput. J. 7 (1964) 303–307.

[10] A.L. Cauchy, Methode generale pour la resolution des systemes d'equations simultanees, Comptes rendus de l'academie des sciences 25 (1847) 536–538.

[11] D. Marquardt, An algorithm for least squares estimation of non-linear parameters, SIAM J. Appl. Math. 11 (1963) 431–441.

[12] R. Fletcher, C.M. Reeves, Function minimization by conjugate gradients, Comput. J. 7 (1964) 149–154.

[13] A. Tiwari, Evolutionary computing techniques for handling variable interaction in engineering design optimisation, Ph.D. Thesis, Cranfield University, Cranfield, UK, 2001.

[14] G.E.P. Box, Evolutionary operation: a method of increasing industrial productivity, Appl. Stat. 6 (1957) 81–101.

[15] A.S. Fraser, Simulation of genetic systems by automatic digital computers. I. Introduction, Aust. J. Biol. Sci. 10 (1957) 484–491.

[16] J.H. Holland, Outline of a logical theory of adaptive systems, J. ACM 9 (1962) 297–314.

[17] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, 1975.

[18] J.R. Koza, Genetic Programming, MIT Press, 1992.

[19] L.J. Fogel, A.J. Owens, M.J. Walsh, Artificial Intelligence through Simulated Evolution, Wiley, New York, 1966.

[20] I. Rechenberg, Cybernetic solution path of an experimental problem, Royal Aircraft Establishment, Library Translation no. 1122, 1965.

[21] H.-P. Schwefel, Numerical Optimisation of Computer Models, Wiley, 1981.

[22] D.B. Fogel, Evolutionary Computation, IEEE Press, New York, 1995.

[23] J.S. Chung, S.M. Hwang, Application of genetic algorithm to process optimal design in non-isothermal metal forming, J. Mater. Process. Technol. 80–81 (1998) 136–143.

[24] K.E. Kinnear, Advances in Genetic Programming., MIT Press, Cambridge, MA, 1994.

[25] T. Back, H.-P. Schwefel, Evolutionary computation: an overview, Presented at the Proceedings of the Third IEEE Conference on Evolutionary Computation, Piscataway, NJ, 1996, pp. 20–29.

[26] D.H. Wolpert, No Free Lunch Theorems for Optimization, IBM Almaden Research Center, California, 1996.

[27] H.W. Ray, J. Szekely, Process Optimization with Applications in Metallurgy and Chemical Engineering, Wiley, New York, 1973.

[28] R. Bellman, Dynamic Programming, Princeton University Press, 1957.

[29] A. Law, D.W. Kelton, Simulation Modeling and Analysis, McGraw-Hill, 1991.

[30] Z. Michalewicz, A survey of constraint handling techniques in evolutionary computation methods, Presented at the Proceedings of the Fourth Annual Conference on EP, Cambridge, MA, 1995, pp. 135–155.

[31] M.S. Elred, Optimization Strategies for Complex Engineering Applications, Sandia Sand98-0340.UC-705, February 1998.

[32] H.A. Simon, The Sciences of the Artificial, MIT Press, Cambridge, MA, 1969.

[33] V. Oduguwa, A. Tiwari, R. Roy, Genetic algorithm in process optimisation, Presented at CDROM Proceedings of the Eighth Online World Conference on Soft Computing in Industrial Applications (WSC-8), Finland, 2003.

[34] V. Oduguwa, R. Roy, Bilevel optimisation using genetic algorithm, Presented at 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002), Divnomorskoe, Russia, 2002, pp. 322–327.

[35] G.J. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic Theory and Applications, Prentice-Hall, New Jersey, 1995.

[36] B. Filipic, B. Sarler, Evolving parameter settings for continuous casting of steel, Presented at the Proceedings of the Sixth European Congress on Intelligent Techniques and Soft Computing EUFIT'98, Aachen, Germany, 1998, pp. 444–449.

[37] B. Sarler, B. Filipic, M. Raudensky, J. Horsky, An interdisciplinary approach towards optimal continuous casting of steel, Presented at Materials Processing in the Computer Age III, Proceedings of TMS, Nashville, Tennessee, TMS, Warrendale, PA, 2000, pp. 27–36.

[38] K. Deb, N. Chakraborti, A combined heat transfer and genetic algorithm modeling of an integrated steel plant bloom reheating furnace, Presented at the Proceedings of the Sixth European Congress on Intelligent Techniques and Soft Computing EUFIT'98, Aachen, Germany, 1998, pp. 439–443.

[39] N. Chakraborti, K. Deb, A. Jha, A genetic algorithm based heat transfer analysis of a bloom re-heating furnace, Steel Res. 71 (2000) 396–420.

[40] N. Chakraborti, R. Kumar, D. Jain, A study of the continuous casting mold using a pareto-converging genetic algorithm, Appl. Math. Model. 25 (2001) 287–297.

[41] C.A. Santos, J.A. Spim Jr., M.C.F. Ierardi, A. Garcia, The use of artificial intelligence technique for the optimisation of process parameters used in the continuous casting of steel, Appl. Math. Model. 26 (2002) 1077–1092.

[42] C.A. Conceicao Antonio, N. Magalhaes Dourado, Metal-forming process optimisation by inverse evolutionary search, J. Mater. Process. Technol. 121 (2002) 403–413.

[43] S. Roy, S. Ghosh, R. Shivpuri, Optimal design of process variables in multi-pass wire drawing by genetic algorithms, J. Manuf. Sci. Eng. 118 (1996).

[44] L. Nolle, D.A. Armstrong, J.A. Ware, F. Biegler-Konig, Optimization of Roll Profiles in the Hot Rolling of Wide Steel Strip, Genetic Algorithm in Engineering Systems: Innovations and Applications, 1997, pp. 133–138.

[45] D.D. Wang, A.K. Tieu, F.G. de Boer, B. Ma, W.Y.D. Yuen, Towards a heuristic optimum design of rolling schedules for tandem cold rolling mills, Eng. Appl. Artif. Intell. 13 (2000) 397–406.

[46] J.S. Chung, S.M. Byon, H.J. Kim, S.M. Hwang, Process Optimal Design in Metal Forming by Double-objective Genetic Algorithm, in: Transactions of the NAMRI/SME XXVII, 2000, pp. 51–56.

[47] V. Oduguwa, R. Roy, Multi-objective optimisation of rolling rod product design using meta-modelling approach, Presented at the Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002), New York, 2002, pp. 1164–1171.

[48] V. Oduguwa, R. Roy, An integrated design optimisation approach for quantitative and qualitative search space, Presented at the Proceedings of the ASME: 2003 ASME Design Engineering Technical Conference, Chicago, IL, 2003.

[49] R. Roy, V. Oduguwa, Multiobjective optimisation of rod design in long product rolling within a quantitative and qualitative search space, Presented at the 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation, Kobe, Japan, 2003.

[50] V. Oduguwa, R. Roy, D. Farrugia, Fuzzy multi-objective optimisation approach for rod shape design in long product rolling, Presented at IFSA'03, Istanbul, Turkey, 2003.

[51] V. Oduguwa, R. Roy, G. Watts, J. Wadsworth, D. Farrugia, Comparison of low-cost response surface models for rod rolling design problems, Presented at the 18th National Conference on Manufacturing Research, Leeds Metropolitan University, UK, 2002, pp. 209–213.

[52] T.A. Runkler, E. Gerstorfer, M. Schang, E. Junnemann, J. Hollatz, Modelling and optimisation of a refining process for fibre board production, Contr. Eng. Pract. 11 (2003) 1229–1241.

[53] A. Santos, A. Dourado, Global optimization of energy and production in process industries: a genetic algorithm application, Contr. Eng. Pract. 7 (1999) 549–554.

[54] M. Borairi, H. Wang, J.C. Roberts, Dynamic modelling of a paper making process based on bilinear system modelling and genetic neural networks, Presented at UKACC International Conference on Control, 1998, pp. 1277–1282.

[55] H. Wang, M. Borairi, J.C. Roberts, H. Xiao, Modelling of a paper making process via genetic neural networks and first principle approaches, Presented at the IEEE International Conference on Intelligent Processing Systems, ICIPS, Beijing, China, 1997, pp. 584–588.

[56] P. Pongcharoen, C. Hicks, P.M. Braiden, D.J. Stewardson, Determining optimum genetic algorithm parameters for scheduling the manufacturing and assembly of complex products, Int. J. Prod. Econ. 78 (2002) 311–322.

[57] K. Wang, T. Lohi, M. Stobbe, S. Engell, A genetic algorithm for online-scheduling of a multiproduct polymer batch plant, Comput. Chem. Eng. 24 (2000) 393–400.

[58] R.M. Marian, L.H.S. Luong, K. Abhary, Assembly sequence planning and optimisation using genetic algorithm, Applied Soft Computing 2/3F, 2003, pp. 223–253.

[59] M. Sakawa, K. Kato, S. Ushiro, M. Inaoka, Operation planning of district heating and cooling plants using genetic algorithms for mixed integer programming, Appl. Soft Comput. (ASC) J. 1 (2001) 139–150.

[60] M. Yokoyama, H.W. Lewis III, Optimization of the stochastic dynamic production cycling problem by a genetic algorithm, Comput. Oper. Res. 30 (2003) 1831–1849.

[61] M. Gen, R. Cheng, S.S. Oren, Network design techniques using genetic algorithms, Adv. Eng. Software 32 (2001) 731–744.

[62] P. Cortes, M.S. Caraballo, J.M. Garcia, J. Larraneta, L. Onieva, GA for planning cable telecommunication networks, Presented at WSC5 Proceedings of the Fifth Online World Conference on Soft Computing Methods in Industrial Applications, 2000, pp. 135–141.

[63] K.R. Baker, Introduction to Sequencing and Scheduling, Wiley, New York, USA, 1974.

[64] Y. Tsujimura, M. Cheng, M. Gen, Improved genetic algorithms for job shop scheduling problems, Eng. Des. Autom. 3 (1997) 133–144.

[65] H.D. Young, University Physics, Reading, MA, 1992.

[66] M. Sakawa, K. Kato, Integer programming through genetic algorithm with double strings based on reference solution updating, Presented at the Proceedings of the 2000 IEEE International Conference on Industrial Electronics, Control and instrumentation, Nagoya, Japan, 2000, pp. 2744–2749.

[67] S.D. Harris, L. Elliot, D.B. Ingham, M. Pourkashanian, C.W. Wilson, The optimisation of reaction rate parameters for chemical kinetic modelling of combustion using genetic algorithms, Comput. Meth. Appl. Mech. Eng. 190 (2000) 1065–1090.

[68] J.K. Rajesh, S.K. Gupta, G.P. Rangaiah, A.K. Ray, Multiobjective optimization of industrial hydrogen plants, Chem. Eng. Sci. 56 (2001) 999–1010.

[69] M. Grujicic, G. Cao, B. Gersten, Optimization of the chemical vapor deposition process for carbon nanotubes fabrication, Appl. Surf. Sci. 199 (2002) 90–106.

[70] L. Balland, N. Mouhab, J.M. Cosmao, L. Estel, Kinetic parameter estimation of solvent-free reactions: application to esterification of acetic anhydride by methanol, Chem. Eng. Process. 41 (2002) 395–402.

[71] M. Emmerich, B. Gro, M. Grotzner, P. Roosen, M. Schutz, A mixed integer evolution strategy for chemical plant optimization, Presented at the Evolutionary Design and Manufacture, ACDM, Plymouth, UK, 2000, pp. 55–67.

[72] M. Emmerich, M. Grötzner, M. Schütz, Design of graph-based evolutionary algorithms: a case study for chemical process networks, Evol. Comput. 9 (2001) 329–354.

[73] J.E. Cuthrell, L.T. Biegler, Simultaneous optimization and solution methods for batch reactor control profiles, Comput. Chem. Eng. 13 (1989) 49–56.

[74] R. Luus, Application of dynamic programming to differential-algebraic process systems, Comput. Chem. Eng. 17 (1993) 373–377.

[75] C. Floudas, A Non-linear and Mixed-integer Optimization: Fundamentals and Applications (Topics in Chemical Engineering), Oxford University Press, New York, 1995.

[76] G. Schernbecker, K.H. Simmrock, Heuristic numeric process synthesis with PROSYN, AIChE Symp. Ser. 92 (1996) 275–278.

[77] J.M. Douglas, Synthesis of separation system flowsheets, AIChE J. 41 (1995) 2522–2536.

[78] M. Schutz, H.-P. Schwefel, Evolutionary approaches to solve three challenging engineering tasks, Computer Methods in Applied Mechanics and Engineering Application of Artificial Intelligence, vol. Special issue on Evolutionary and Genetic Algorithms in Computational Mechanics and Engineering, 1999.

[79] D.C. Montgomery, Design and Analysis of Experiments, 4th ed. Wiley, 1997.

[80] H. Watabe, N. Okino, A study on genetic shape design, Presented at the Proceedings of the Fifth International Conference on Genetic Algorithms, University of Illinois, USA, 1993, pp. 445–450.

[81] K. Weinert, J. Mehnen, Prestifilippo, Optimal surface reconstruction from digitized point data using CI methods, University of Dortmund, Germany, Technical Report CI-5/97, 1997.

[82] M. Higashi, K. Kohji, M. Hosaka, Generation of high-quality curve and surface with smoothly varying curvature, Presented at the Proceedings of EUROGRAPHICS'88 Conference, 1988, pp. 79–92.

[83] R. Mussa, R. Roy, G. Jared, Surface optimisation within a CAD/CAM environment using genetic algorithms, Presented at the Third World Conference on Soft Computing in Engineering Design and Manufacturing (WSC3), Cyberspace, 1998.

[84] R. Roy, A. Tiwari, O. Munaux, G. Jared, Real-life engineering design optimisation: features and techniques, Presented at the CDROM Proceedings of the Fifth Online World Conference on Soft Computing in Industrial Applications (WSC-5), Finland, 2000.

[85] R. Roy, A. Tiwari, A. Braneby, Making evolutionary design optimisation popular in industry: Issues and techniques, Presented at the Sixth Online World Conference on Soft Computing in Industrial Applications (WSC-6), Cyberspace, 2001.

[86] G. Liden, S.K.E. Westberg, Fairing of surfaces with optimization techniques using FANGA curves as quality criterion, Comput. Aided Des. 25 (1993) 411–420.

[87] H. Einar, E. Skappel, FORMELA: a general design and production data system for sculptured products, Computer Aided Des. 5 (1973) 68–76.

[88] E. Kauffman, R. Klass, Smoothing surfaces using reflection lines for families of splines, Comput. Aided Des. 20 (1988) 73–78.

[89] R. Roy, G. Jared, J. Grau, T. Buchannan, Adaptive design decision support within CAD environment, Presented at Design Reuse, 1998, pp. 255–262.

[90] T. Back, U. Hammel, H.P. Schwefel, Evolutionary computation: comments on the history and current state, IEEE Trans. Evol. Comput. 1 (1997) 3–17.

[91] M. Szilvasi-Nagy, Detecting regions with undesirable curvature, Comput. Aided Geom. Des. 14 (1997) 699–706.

[92] G. Jared, R. Roy, J. Grau, T. Buchannan, Flexible optimisation within the CAD/CAM environment, Presented at the Proceedings of the CIRP International Seminar on Intelligent Computation in Manufacturing Engineering, Capri, Italy, 1998, pp. 503–508.

[93] G.E. Carrillo-Ureta, P.D. Roberts, V.M. Becerra, Genetic algorithm for optimal control of beer fermentation, Presented at the Proceedings of IEEE International Symposium on Intelligent Control, Mexico City, 2001, pp. 391–396.

[94] H.S. Cho, W.S. Park, B.W. Choi, M.C. Leu, Determining optimal parameters for stereolithography processes via genetic algorithm, J. Manuf. Syst. 19 (2000) 18–27.

[95] J.A. Covas, A.G. Cunha, P. Oliveira, An optimization approach to practical problems in plasticating single screw extrusion, Polym. Eng. Sci. 39 (1999) 443–456.

[96] M. Cerrolaza, W. Annicchiarico, Optimization of finite element bidimensional models: an approach based on genetic algorithms, Finite Element Anal. Des. 29 (1998) 231–257.

[97] K. Deb, S. Gulati, Design of truss-structures for minimum weight using genetic algorithms, Finite Elements Anal. Des. 37 (2001) 447–465.

[98] T. Chen, C. Lin, Determination of optimum design spaces for topology optimization, Finite Elements Anal. Des. 36 (2000) 1–16.

[99] K. Hamza, H. Mahmoud, K. Saitou, Design optimization of N-shaped trusses, Presented at the Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002), New York, 2002, pp. 1089–1096.

[100] L. Barone, L. White, P. Hingston, Designing crushers with a multi-objective evolutionary algorithm, in: Presented at the Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002), New York, 2002, pp. 995–1002.