International Journal of Research in Engineering and Innovation (IJREI)
journal home page: http://www.ijrei.com
ISSN (Online): 2456-6934

Performance measurement of flexible manufacturing system in context to Indian industries using soft computing techniques

R.S. Mishra, Himani Saxena, A.K. Madan

Department of Mechanical, Production Industrial and Automobiles Engineering, Delhi Technological University, Delhi, India

Abstract

This paper focuses use of soft computing techniques for complete scheduling of FMS. The optimization of various components of FMS was analyzed by using various techniques i.e. (fuzzy logic, genetic algorithm and neural network). It was observed that Fuzzy logic technique gives better results.

Keywords: Soft computing techniques, FMS, Optimization techniques

1. Introduction

Soft computing referred to as computational intelligence, by which the inexact solution with unknown algorithm can be computed to get an exact solution in polynomial time Soft computing works on uncertainty, partial truth, and approximation. The role model for soft computing is the human mind.The principal constituents of Soft Computing are Fuzzy Logic, Evolutionary Computation, Machine Learning and Probabilistic Reasoning.

Soft computing encourages the integration of soft computing techniques and tools into both day to-day and advanced practical applications. By merging novel notions and techniques of soft computing with other diversified multidiscipline, a unifying platform that fosters comparisons, extensions, and new applications can be obtained in order to establish better outputs.

Soft computing aims to overcome NP (nondeterministic polynomial time) complete problems and well suited for solving real world problems where ideal models are not available with reasonably lesser time. Soft computing yields flexible knowledge acquisition (by machine learning from data and by interviewing experts), and flexible knowledge processing (inference by interfacing between symbolic and pattern knowledge), which enable intelligent systems to be fabricated at less cost.

FMS Scheduling problem is one of the most difficult NP-hard combinational optimization problems. Therefore, determining an optimal schedule and controlling an FMS is considered a difficult task. To achieve high performance for an FMS, a good scheduling system should make a right decision at a right time according to system conditions. This paper focuses uses soft computing techniques for complete scheduling of FMS

The complete scheduling of FMS includes two independent processes: sequencing of jobs and scheduling those prioritized jobs. FMS is actually an automated set of numerically controlled machine tools and material handling systems, capable of performing a wide range manufacturing operations with quick tooling and instruction changeovers.

FMS differs from the conventional systems in terms of flexibility in the flow of materials from one tool to another and performing the operations as per the required sequence. Flexibility in material handling, in combination with multipurpose tools, makes it possible for a flexible manufacturing system to process a great diversity of parts. FMS is essentially a computer-controlled production system, which brings together different standalone machines and control equipment capable of processing a variety of part types or jobs. FMS ensures quality product at lowest cost while maintaining small lead-time.

Main purpose of FMS is to achieve efficiency of well-balanced transfer line while retaining the flexibility of the job shop. A FMS has four or more processing workstations connected mechanically by a common part handling system and electronically by a distributed computer system. It covers a wide spectrum of manufacturing activities such as machining, sheet metal working, welding, fabricating, scheduling and assembly.

Corresponding author: R.S. Mishra
Email Id:rsmishradtu@gmail.com

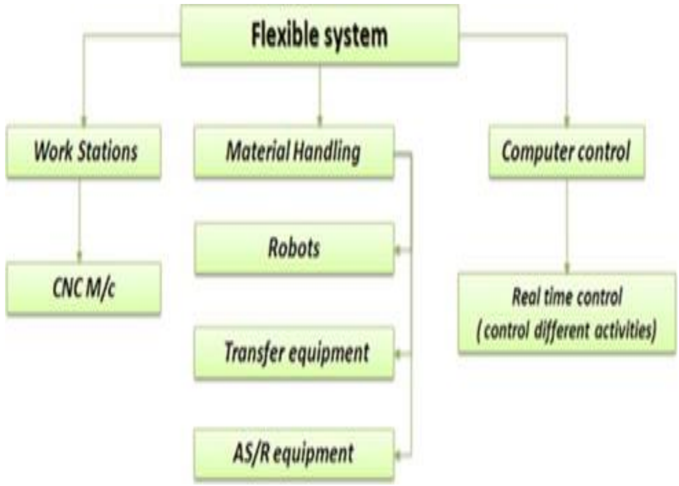

Figure1: Components of FMS

2. Literature Review

The papers [1, 2] represent two authoritative and extensive surveys of this research area, which has received considerable attention in the recent literature. In these contexts, dispatching systems take decisions on part flow in real time and represent a form of low level control based on algorithms using a limited amount of local information. Adequate coordination/cooperation mechanisms are required to achieve the desired global behavior when the decision task is distributed across concurrently operating agents. In fact, many researches on focus on the development of decision strategies and protocols of interaction between agents, based on metaphors of negotiation in micro-economic environments.

Since the first appearance of the earliest approaches, the main motivation of this type of research is that fictitious currency and pricing policies represent ‘‘invisible hands to guide the negotiation to improve system performance’’ [3]. Hence, these invisible hands must also compensate for the lack of a general view in local information. In other words, a dominating opinion in multi-agent manufacturing research is that by using market or auction based task contracting, the network of interacting autonomous agents can lead to the requested characteristics of flexibility and reactivity of the manufacturing system [4–6]. On the other hand, many authors [7] also point out the inherent difficulty in designing effective forms of cooperation between local autonomous controllers, without contradicting in some way the multi-agent design principles, which suggest the absence of hierarchical forms of supervision. In fact, more than 30 years of research in the field of dispatching systems confirm that there is no explicit way to relate the local objectives of a decision entity to global system performance [8]. As a consequence, global performance becomes extremely sensitive to the definition and the fine tuning of the task-contracting rules [7]. Furthermore, in general, negotiation algorithms have many free parameters to tune, and there are few general criteria to set up agents that can effectively emulate human decision-making. The inspiration from communities of intelligent decision makers in uncertain and extremely dynamic environments, and that fuzzy techniques are suited to model human decision-making. Therefore, this paper discusses the potentialities of the challenging combination of soft computing techniques and multi-agent paradigms in task-contracting problems for manufacturing control. In particular, the paper examines if and how much agents’ decision schemes benefit from the application of fuzzy methodologies.

In principle, these approaches are more effective in finding qualitative and approximate rewarding tradeoffs between conflicting decision objectives, in obtaining more context-independent reactive behaviors, and in solving the coordination/cooperation problems. Other tools from soft computing lend themselves to tackle multi-agent research problems. In particular, evolutionary algorithms (EA) [9,10] are a set of versatile stochastic search techniques inspired to natural evolution and genetics that have been successfully applied to a variety of manufacturing problems, including also task negotiation and scheduling in distributed environments. To enlighten the potentiality of soft computing methodologies, this paper describes a new multi-objective task-contracting mechanism based on fuzzy decision-making and compares it with other conventional negotiation schemes for real-time part flow control. For comparison purposes, the paper considers a recently proposed evolutionary strategy [11] to adapt agent’s decision parameters to the changing conditions of the manufacturing floor. All the proposed approaches are compared on a detailed simulation model of a hypothetical manufacturing system. [12]. Referring to a common benchmark is necessary. In fact, as recently pointed out by Cavalieri et al. [13], many authors often do not provide sufficient detail on their design hypotheses and on the structural characteristics of the manufacturing system. Thus, an objective view of the applicability and the performance bounds of their approaches is lacking. Hence, the adoption of this case study as common framework to evaluate the performance of MAS is fundamental for an objective understanding of the true potentials of agent technology in real-time control. Our simulation study encompasses both steady-state performance measures, and the response of the controllers to dynamic changes as workstation (machine) and AGV failures and changes in the workload. Encompassing a set of theories and algorithms inspired to human approximate classification and reasoning, fuzzy logic (FL) is a main component of soft computing that has received considerable attention in the area of manufacturing control and optimization. Since their earliest developing stages, fuzzy techniques [14] have been applied to assign priorities in job shop environments. In fact, the inherent turbulence and the lack of precise information about the real-time operating conditions make conventional planning strategies based on analytical models extremely unsuitable for these environments.

Fuzzy systems, on the other hand, attempt to mimic human expertise in the form of rules as ‘‘IF work in progress of part type A is high AND average service time of part type B is low AND . . . THEN priority part A is low AND priority part type

B is medium AND . . .''. Therefore, in such applications the fuzzy controller works as a numerical model of the heuristic knowledge of a human planner, and in general yields performances comparable to those of plants directly controlled by human experts. Among the others, Ben- Arieh and Lee [15], Nahavandi and Solomon [16] and Erkmen et al. [17] have proposed this typical application of fuzzy control in manufacturing environments.

These approaches not only lead to higher performance with respect to conventional strategies, but rather provide a transparent evaluation and decision logic, which enlightens the mechanisms underlying the input–output behavior of the controllers. This is an extremely important feature, since many manufacturing environments still need human supervision to operate correctly. Angsana and Passino [18] proposed a scheduling approach that is one of the earliest attempts to design a distributed network of fuzzy controllers in manufacturing.

The authors underline that their scheduling system, a fuzzified version of optimal buffer clearing policies, could be viewed as a ''distributed intelligent system'', since each machine is separately controlled by a local dispatcher. Each dispatcher is a fuzzy controller working on rules having buffer contents as inputs and next part type to be processed as output. The system is distributed, since each controller uses only local information to perform its decisions. However, from the multi-agent viewpoint it lacks of explicit interaction between the autonomous controllers, since the authors acknowledge that the number of parts travelling over the track from one machine to the other is the only form of communication between controllers. An interesting aspect that is considered in the paper is the capability of the controllers to self-tune themselves when machine parameters change. More precisely, the authors devise an adaptation law that changes a single parameter of the control on the basis of the performance of the policy over a shifting time window. The parameter under tuning is the amplitude of the definition universe of the fuzzy sets, and the objective of the adaptation is to have rules always coherent with the actual operating conditions. A recurrent idea to improve the effectiveness of dispatching strategies is to develop specific heuristics to switch in real time between different decision rules. To implement such schema in a distributed decision environment, it is necessary to understand how intelligent agents modify their actions taking into account the system operating conditions. There are recent centralized solutions to this problem. For instance, Park et al. [8] define a mapping ''system-state-to dispatching- rule'' to implement an adaptive sequencing algorithm in production lines. The mapping is based on IF–THEN statements, e.g. ''IF system utilization is greater than ''a'' AND buffer size is ''b'' AND . . . THEN the dispatching law is SPT''. The rule base is also optimized by an ''inductive learning approach'' using a set of training examples. Analogously, Yu et al. [19] propose fuzzy rules to associate a set of ''environmental variables'' describing the manufacturing conditions (e.g. the current workload or the remaining production time for parts at hand) to a set of dispatching rules. Recently, also Kazerooni et al. [20], Naso and Turchiano [21], Gao et al. [22], Wan and Yen [23] suggested similar approaches. However, associating a specific dispatching rule to each state of the plant is not an easy task, especially when multiple dispatchers share the part flow control duties. Namely, in most manufacturing systems, there is no evident relationship between system performance and the single decisions taken by the autonomous agents. Real-time task scheduling methodologies for MAS employ intelligent negotiation strategies to increase system fault tolerance and robustness to unforeseen circumstances (e.g. bottlenecks and variable product demand). In these approaches, agents have to effectively balance their autonomy against simplicity of regulation of the overall system behavior. However, task-contracting schemes inherit the inherent myopia and unpredictability of performance bounds typical of dispatching systems. These limitations are widely acknowledged among the main reasons against the widespread use of agents in industry. In this specific context, the cross-fertilization of soft computing and MAS tools offers a promising research direction that has been recently investigated. For instance, to ensure that agents act coherently from a global point of view, compensating the non-local effects of local decisions,

Maione and Naso [11] propose an evolutionary supervisor coordinating two types of agents, namely part agents (PA) and workstation agents (WA), which control part flow and service sequence, respectively. The objective of PA is to obtain the requested service in the shortest time. On their part, WA must minimize idle and setup times. Both type of agents use a fuzzy multi-criteria decision algorithm combining several decision rules in a unique criterion. All agents have a limited life-cycle, after which a discrete-event simulation evaluates their fitness (i.e. the ability to fulfill their objective). Then, an iterative evolutionary strategy employs simulation results to compute new agents with improved fitness. Hence, the evolutionary strategy acts as an implicit supervisor continuously adapting the population of agents controlling the plant. Evolutionary algorithms, another fundamental component of soft computing, are robust search strategies, inspired to natural evolution laws and genetics, which have been extensively applied to manufacturing problems. Most of this research focuses on off-line optimization of hard combinatorial problems (e.g. production planning, scheduling). References [24, 25] are some recent examples of this research, and paper [26] offers a comprehensive survey of the research in this area in recent years.

Due to their common inspiring metaphor, centered on communities of biological entities ''adapting'' their behavior to changing environments, and due to the versatility and robustness of their performance, evolutionary algorithms have received an increasing attention in the area of MAS. Also in this case, the lack of adequate models, and the inherently complex, turbulent and unpredictable nature of the operating environment impede the application of conventional optimization approaches, and constitute the main motivations to use surviving-of-the-fittest-driven heuristics. A common example is the optimization of the decision strategies of a network of distributed controllers. In principle, whenever the strategy underlying agents' decision, interaction or negotiation tactics can be parameterized (i.e. encoded in a string of binary or real parameters), it is possible to use an\ evolutionary algorithm to optimize the entire network of agents. Applications of this point of view include communities of cooperative robots [27, 28] or artificial benchmarks [29–32]. Although size and complexity of these agent benchmarks are considerably smaller than those of manufacturing system problems, many authors conclude that the successful results obtained with intelligent agents in these simplified virtual domains will be soon extended to real-world problems. Recent literature offers some preliminary applications in manufacturing domains. For instance, McDonnell and Joshi [33] use a reinforcement learning algorithm to determine the optimal negotiation strategy between agents controlling the machine setup timing in a flexible manufacturing system (FMS).

3. Conclusion from Literature Review

From the above literature review it is concluded that all the approaches simulate the hypothetical manufacturing system using multi-agent control systems. Also a the task-contracting scheme for multi-agent manufacturing control based on soft computing that applies fuzzy techniques which are implemented on a real-time task-contracting mechanism for a part flow control in manufacturing floor. For comparison purposes, the evolutionary strategies are adapted to optimize decision parameters with the changing conditions of the manufacturing floor.

4. Research Gap Identified

The literature review reflects the work on hypothetical manufacturing system using multi agent systems which simulated the part flow control in manufacturing floor using fuzzy technique and evolutionary strategy. By optimizing the various components of FMS by measuring the performance of FMS which includes quality control, degree of flexibility, CNC machine tools, AS-RS, AGV's and robots in context to industrial approach by using different soft computing techniques. The different techniques will produce different results to each component among them the exact value is evaluated. This paper mainly deals with following objectives

- Optimization of CNC machine tools.
- Optimization of Robots, Automated guided vehicle (AGV), Automated Storage and Retrieval System (AS-RS).
- Optimization of Computer aided quality control.
- Optimization of degree of flexibility if FMS using soft computing techniques.
- Comparison of results of performance measurement of FMS using various techniques.

5. Soft-computing methodologies

Soft computing referred to as computational intelligence, by which the inexact solution with unknown algorithm can be computed to get an exact solution in polynomial time. Soft computing works on uncertainty, partial truth, and approximation. The role model for soft computing is the human mind. Soft computing encourages the integration of soft computing techniques and tools into both day to-day and advanced practical applications.

Various methodologies are:

- Fuzzy Logic (FL),
- Genetic algorithm (GA),
- Artificial Neural Networks (ANN),
- Additionally Some Machine Learning (ML) and Probabilistic Reasoning (PR) areas.

5.1 Fuzzy Logic (FL)

- Fuzzy set theory is proposed in 1965, by A. Zadeh as generalization of classical set theory.
- In classical set theory an element is either belong to or not belong to a set and hence such set are termed as crisp set.
- In fuzzy many membership are allowed.

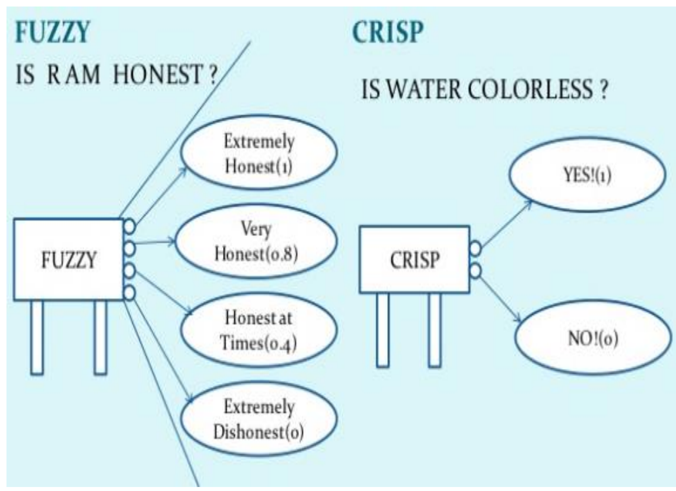

Figure 2: Fuzzy Logic Technique used

5.2 Genetic Algorithm (GA)

It is a process of natural evolution. It is a part of evolutionary computing, which is a rapidly growing area of artificial intelligence. Outline of the Basic Genetic Algorithm

- [Start] Generate random population of *n* chromosomes
- [Fitness] Evaluate the fitness *f(x)* of each chromosome *x* in the population.
- [New population] Create a new population by repeating following steps until the new population is complete
  a. [Selection]
  b. [Crossover]
  c. [Mutation]
  d. [Accepting]

- [Replace] Use new generated population for a further run of algorithm
- [Test] If the end condition is satisfied, stop,
- [Loop] Go to step 2

5.3 Neural Network

It is refer to a network or circuit of biological neurons. The modern term refers to artificial neural networks, which are composed of artificial neurons or nodes. Its capabilities are

- Robustness
- Mapping
- Generalization
- Fault tolerance
- High speed information processing.

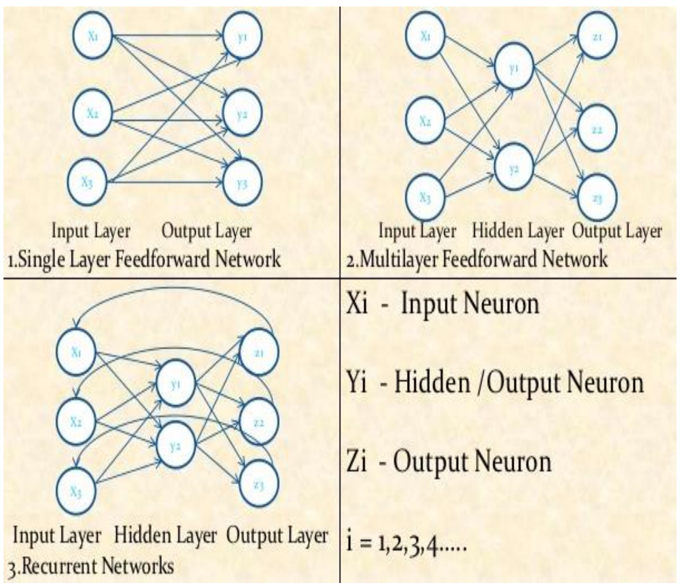

Figure 3: Artificial Neural network technique

6. Conclusion

1. By optimizing machine tools, the tools will more customized and will move in desired locations.
2. By optimizing computer aided quality control, the production plans and the product quality will be much improved.
3. By optimizing the degree of flexibility, using soft computing techniques, in exact and imaginary problems will have accurate solutions.
4. By comparing the results of various components of FMS, the optimized solution can be obtained.

References

[1] A.D. Baker, A survey of factory control algorithms that can be implemented in a multi-agent hierarchy: dispatching scheduling and pull, Journal of Manufacturing Systems 17 (1998) 297–320.

[2] W. Shen, D.H. Norrie, Agent-based systems for intelligent manufacturing: a state-of-the-art survey, Knowledge and Information Systems: An International Journal 1 (2) (1999) 129–156 (extended version available at http://imsg.enme.ucalgary.ca/publication/abm.htm).

[3] G.Y.L. Lin, J.J. Solberg, Integrated shop floor control using autonomous agents, IIE Transactions 24 (3) (1992) 57–71.

[4] K.H. Kim, J.Y. Song, K.H. Wang, A negotiation based scheduling for items with flexible process plans, Computers Industrial Engineering 33 (3–4) (1997) 785–788.

[5] R.J. Rabelo, L.M. Camarinha-Matos, H. Afsarmanesh, Multiagent- based agile scheduling, Robotics and Autonomous Systems 27 (1999) 15–28.

[6] P. Sousa, C. Ramos, A distributed architecture and negotiation protocol for scheduling in manufacturing systems, Computers in Industry 38 (1999) 103–113.

[7] H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, P. Peeters, Reference architecture for holonic manufacturing systems: PROSA, Computers in Industry 37 (3) (1998) 255–274.

[8] S.C. Park, N. Raman, M.J. Shaw, Adaptive scheduling in dynamic flexible manufacturing systems: a dynamic rule approach, IEEE Transactions on Robotics and Automation 13 (4) (1997) 486–502.

[9] D.B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, Wiley–IEEE Press, Piscataway, NJ, 1999.

[10] Z. Michalewicz, Genetic Algorithms þ Data Structures ¼ Evolution Programs, Springer-Verlag, Berlin, 1996.

[11] B. Maione, D. Naso, Evolutionary adaptation of dispatching agents in heterarchical manufacturing systems, International Journal of Production Research 39 (7) (2001) 1481–1503.

[12] S. Cavalieri, L. Bongaerts, M. Taisch, M. Macchi, J. Wyns, A benchmark framework for manufacturing control, in: Proceedings of the Second International Workshop on Intelligent Manufacturing Systems, Leuven, September 1999, pp. 22–24.

[13] S. Cavalieri, M. Garetti, M. Macchi, M. Taisch, An experimental benchmarking of two multi-agent architectures for production scheduling and control, Computers in Industry 43 (2000) 139–152.

[14] H.-J. Zimmermann, Fuzzy Sets Theory and its Applications, Kluwer Academic Publishers, Dordrecht, 1993.

[15] D. Ben-Arieh, E.S. Lee, Fuzzy logic controller for part routing, in: H.R. Parsei, M. Jamshidi (Eds.), Design and Implementation of Intelligent Manufacturing Systems from Expert Systems, Neural Networks, to Fuzzy Logic, Prentice - Hall, New Jersey, 1995.

[16] S. Nahavandi, P. Solomon, Application of fuzzy logic to shop floor scheduling, in: Proceedings of the Second New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems, 1995, pp. 365–368.

[17] A.M. Erkmen, M. Erbudak, O. Anlagan, O. Unver, Genetically tuned fuzzy scheduling for flexible manufacturing systems, in: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 2, 1997, pp. 951–956

[18] A. Angsana, K.M. Passino, Distributed fuzzy control of flexible manufacturing systems, IEEE Transactions on Control Systems Technology 2 (4) (1994) 423–435.

[19] L. Yu, H.M. Shih, T. Sekiguchi, Dynamic selection ofdispatching rules by fuzzy inference, in: Proceedings of the IEEE World Congress on Computational Intelligence, vol. 2, 1998, pp. 979–984.

[20] A. Kazerooni, F.T.S. Chan, K. Abhary, Real-time operation selection in a FMS using simulation—a fuzzy approach, Production Planning and Control 8 (8) (1997) 771–779.

[21] D. Naso, B. Turchiano, A fuzzy multi-criteria algorithm for dynamic routing in FMS, in: Proceedings of the International

Conference on Systems, Man, and Cybernetics (IEEE SMC'98), San Diego, CA, USA, 11–14 October 1998, pp. 457–462.
[22] C. Gao, P. Li, R. Li, A novel approach to intelligent scheduling based on fuzzy feature selection and fuzzy classifier, in: Proceedings of the 38th IEEE Conference on Decision and Control, vol. 5, 1999, pp. 5307–5311.
[23] G. Wan, B.P.-C. Yen, A fuzzy logic system for dynamic job shop scheduling, in: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC'99), vol. 4, 1999, pp. 546–551.
[24] Y. Song, J.G. Hughers, A genetic algorithm with a machine order-based representation scheme for a class of job shop scheduling problem, in: Proceedings of the American Control Conference, San Diego, CA, USA, 1999, pp. 895–899.
[25] S. Hajri, N. Liouane, S. Hammadi, P. Borne, A controlled genetic algorithm by fuzzy logic and belief functions for job shop scheduling, IEEE Transactions on Systems, Man, and Cybernetics 30 (5) (2000) 812–818.
[26] C. Dimopoulos, A.M.S. Zalzala, Recent developments in evolutionary computation for manufacturing optimisation: problems, solutions, and comparisons, IEEE Transactions on Evolutionary Computation 4 (2) (2000) 93–113.
[27] H. Hagras, V. Callaghan, M. Colley, Online learning of fuzzy behaviour co-ordination for autonomous agents using genetic algorithms and real-time interaction with the environment, in: Proceedings of Ninth IEEE International Conference on Fuzzy Systems (FUZZY IEEE 2000), vol. 2, 2000, pp. 853–858.
[28] P.H. Hanh, W. Rinsurongkawong, T. Ujjubandb, Distributed multi-level adaptation of dynamic multi-agent systems, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (IEEE SMC'99), vol. 5, 1999, pp. 126–131.
[29] B. Maskell, M. Wilby, Evolving software agent behaviour, in: Proceedings of the Global Communication Conference (Globecom'96), vol. 1, 1996, pp. 90–94.
[30] M.C. Sinclair, S.H. Shami, Evolving simple software agents: Comparing genetic algorithms and genetic programming performance, in: Proceedings of the IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, 1997, pp. 421–426.
[31] N. Matos, C. Sierra, N.R. Jennings, Determining successful negotiation strategies: an evolutionary approach, in: Proceedings of the IEEE International Conference on Multi-Agent Systems, 1998, pp. 182–189.
[32] K.J. Mackin, E. Tazaki, Emergent agent communication in multiagent systems using automatically defined function genetic programming, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (IEEE SMC'99), vol. 5, 1999, pp. 138–141.
[33] P. McDonnell, S. Joshi, A reinforcement learning approach to support setup decisions in distributed manufacturing systems, in: Proceedings of Sixth International Conference on Emerging Technologies and Factory Automation (ETFA'97), 1997, pp. 221–225.