Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

An Artificial Immune Systems (AIS)-based Unified Framework for General Job Shop Scheduling

Henry Y. K. Lau, X. Qiu

Department of Industrial and Manufacturing Systems Engineering,
The University of Hong Kong, Pokfulam Road, Hong Kong
(Tel: +852-28578255; e-mail: hyklau@hku.hk)

Abstract: Process scheduling is a classical problem in the field of production planning and control; in particular, effective job shop scheduling remains an essential component in today's highly dynamic and agile production environment. This paper presents unified framework for solving generic job shop scheduling problems based on the formulation of a job shop into three main classes of problem, namely, static, semi-dynamic and dynamic scheduling problems. Algorithms based on artificial immune systems, an engineering analogy of the human immune system, are developed to solve the respective classes of job shop scheduling problems. A high level decision support model is presented for the effective deployment of the scheduling strategies whereby a unified approach to solving real job shop problems is achieved.

1. INTRODUCTION

Job shop scheduling problem (JSSP) is a traditional NP-hard problem in complex production systems (Zobolas *et al*., 2008). In general, the problem involves the allocation of scarce resources to task over a period of time under certain constraint of utilization, restricted time windows, etc. (Pinedo, 2012). In production engineering, the automatic generation of optimal job shop schedules is important to the overall production operation with best resource utilization and efficient management of the production facilities. Our previous research modelled a generic job shop scheduling problem as three distinct but loosely related problems, namely, static JSSP that operates under a static scheduling environment with known information about the jobs and machines without unexpected events; semi-dynamic JSSP – a static JSSP with the presence of uncertainties occurred in the dynamic scheduling process; and dynamic JSSP that operates under a dynamic operating environment with unpredictable disruptions.

As an NP-hard problem of resource allocation, meta-heuristics and artificial intelligence methods have been developed over the years to produce optimal solutions. Some of these include taboo search (Watson *et al*., 2003), simulated annealing (Ponnambalam *et al.*, 1999), genetic algorithm (Dellacroce *et al*., 1995; Wang and Tang, 2011), particle swarm optimization (Niu *et al*., 2008), ant colony optimization (Puris *et al*., 2007), bee colony optimization (Wong *et al*., 2010), and artificial bee colony (Yao *et al*., 2010). These methods are well studied and are successfully applied to generate solution in different domains.

In our previous research, a number of AIS-based optimization algorithms have been developed to solve the three different classes of job shop scheduling problems (Qiu and Lau, 2012; 2013). Current research suggests that in practice, these three classes of job shop scheduling problems are not completely independent. During the epoch of a typical job shop, the problem often transits from one class to another and there exists some correspondence between them. In this paper, we introduce a unified framework to integrate the three classes of the problem and the corresponding AIS-based algorithm for solving a job shop scheduling problem in a holistic manner. By building the relationship among the three classes of job shop scheduling problems, a unified scheduling strategy can be formulated with a view to producing a more complete solution that is amenable for practical deployment

2. IMMUNE-BASED SCHEDULING ALGORITHMS

Artificial immune systems (AIS) is an artificial intelligent method with appealing features of the human immune system including self-learning, long lasting memory, cross reactive response, self-identity, discrimination of self from non-self, fault tolerance and autonomy (de Castro and Timmis, 2002). With these properties, AIS has been successfully been applied to tackle problems in optimization, clustering, pattern recognition, anomaly detection, computer security, machine learning, scheduling, robotics, and control (Hart and Timmis, 2008). In the domain of scheduling, AIS-based algorithms have been developed for flow shop scheduling (Kumar, et al., 2006), job shop scheduling (Chandrasekaran *et al*., 2006), flexible job shop scheduling (Bagheri *et al*., 2010;), resource constraint project scheduling (Peteghem and Vanhoucke, 2009), multiprocessor scheduling (Wojtyla *et al*., 2006).

In addition, the computation paradigms of AIS-based algorithms also shows strong learning and adaptability to its surroundings, these specific features are considered highly suitable and applicable in dealing with system involving constantly changing dynamics. In particular, a new AIS algorithm known as Dendritic Cell Algorithm (DCA) has gained great success in solving computer security problems (Greensmith *et al.*, 2005) for detecting intrusion and attacks in computer network. The following section briefly summarizes the AIS-based algorithms we developed for solving the three classes of job shop scheduling problems.

3. AIS ALGORITHMS FOR JSSP

For the static problem, a hybrid AIS based algorithm that incorporates the mechanisms of clonal selection theory (Timmis, 2007) and immune network theory (Jerne, 1974; Forrest and Beauchemin, 2007), and particle swarm optimization (PSO). The clonal selection theory establishes the framework of the hybrid algorithm, while the immune network theory is applied to increase the diversity of antibody set which represents the solution candidates. The algorithm involves the processes of selection, cloning, hypermutation, memory, and receptor editing. The PSO is designed to optimize the hypermutation process of the antibodies to accelerate the search procedure.

The hybrid algorithm is tested with benchmark problems of different sizes and is compared with greedy randomized adaptive search procedure, modified genetic algorithm, multi-modal immune algorithm, and other AIS based or hybrid algorithms discussed in the references (Banharnsakun *et al.*, 2011; Binato *et al.*, 2002; Chandrasekaran *et al.*, 2006; Coello *et al.*, 2003; Ge *et al.*, 2008; Luh and Chueh, 2009). The algorithm performs competitively in all cases as it achieves the global optimal solution even for large-size problems. The results show improvements in solution accuracy and computational efficiency. For example, the average computation time for the problem sizes of 6×6, 10×5, 10×10, 15×5, 15×10, 20×5, 20×10, 20×15 require about 20 seconds, 30 seconds, 5 minutes, 2.5 minutes, 20 minutes, 4 minutes, 52 and 55 minutes respectively when these problems are running on a Intel Core 2 Quad Q9400 2.66 GHz PC.

For the class of semi-dynamic JSSP, a DCA-based algorithm is developed to handle unexpected events occurred in the job shop with these events being modelled as intrusions in a computer system. With regard to semi-dynamic JSSP, the process of rescheduling that updates the on-going processing schedule in response to the changes, is used to minimize the negative influence when disruptions causes significant deterioration to the overall working system. While finding a rescheduling process to maintain job shop stability and efficiency is one the most difficult tasks, we develop an extended DCA algorithm to arrange a dynamic rescheduling process to keep the job shop working in a stable and efficient manner. An outline of the algorithm is shown in Fig. 1.

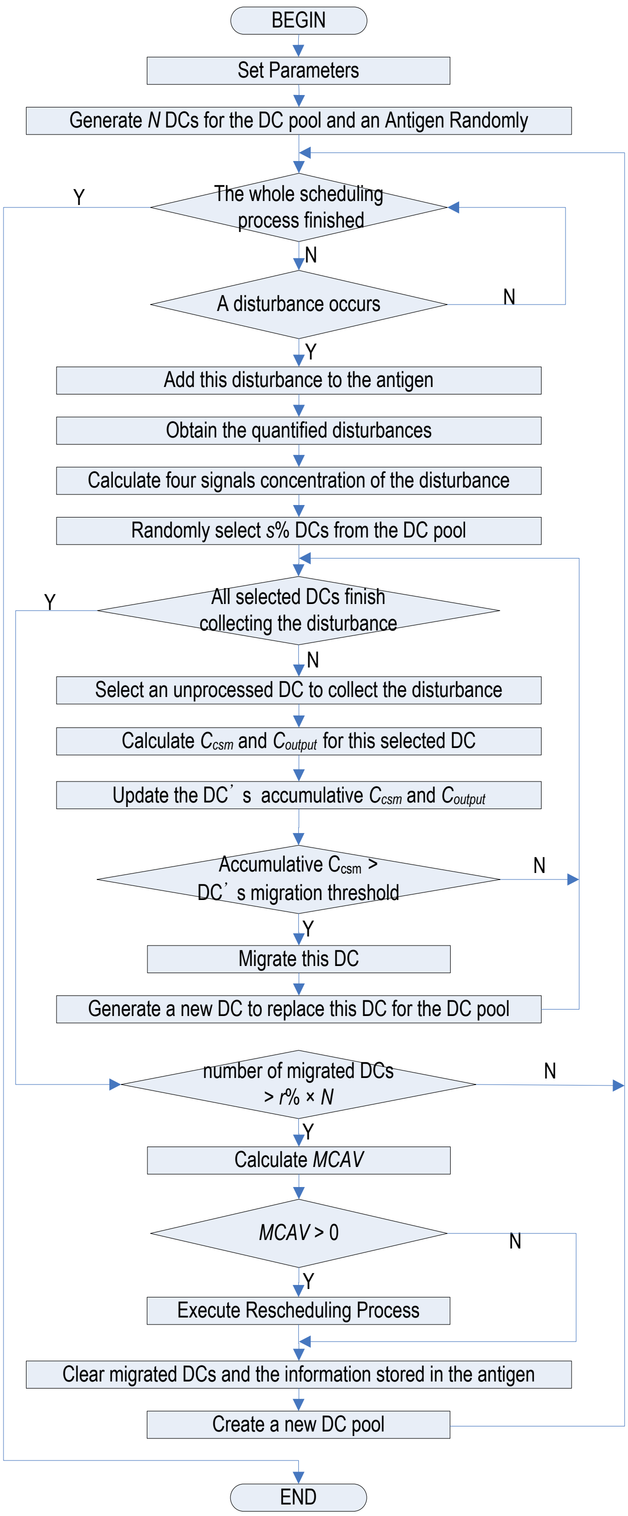

Fig. 1. An DCA algorithm for semi-dynamic JSSP. In the algorithm, based on the states of the migrated dendritic cells (DCs), each antigen type is assigned a context called the mature context antigen value (MCAV) which is used to assess its degrees of anomaly.

Experiments of 21 problems of different dimensions benchmark problem used in the static cases are selected with randomly generated disturbances introduced. The experiment results demonstrate its ability of timely triggering the rescheduling process.

For the dynamic job shop scheduling, we formulate the problem based on the classical combinatorial optimization problem with the assumption that new jobs continuously arrive in a stochastic manner with the existence of unpredictable disturbances. We develop a hybrid algorithm based on the integration of immune network theory with Priority Dispatching Rules (PDRs). Based on the immune network theory, the idiotypic network model (de Castro and Timmis, 2002) is incorporated for the priority dispatching rules to dynamically select the most appropriate priority dispatching rules for the jobs waiting to be processed by an available machine under a dynamic environment.

Numerical experiments are performed to benchmark the performance of the algorithm with other 155 popular dispatching rules for solving multi-objective problems that are formed by a random weighted combination of the eleven representative objectives. For each experiment, the test data set contains 1000 randomly generated test samples and each sample contains 10000 continuously arriving jobs. According to Fig. 2, results show that in a dynamic job shop environment, our algorithm outperforms 762 times out of 1000 test with the other 155 priority dispatching rules, and rank second for 136 times etc. With this result, the effectiveness of the idiotypic network model for solving multi-objective problem is established

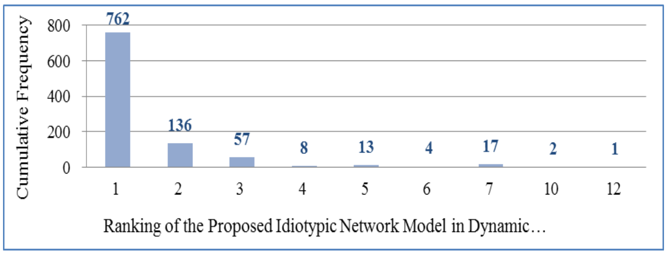

Fig. 2. Results for solving multi-objective dynamic JSSP

4. A STRATEGY FOR SOLVING GENERAL JOB SHOP

We adopt the switching strategy to integrate the three algorithms to be deployed in different job shop situations. Switching is defined as a strategy that changes the sequencing approach to the dispatching approach by taking into account the system status (Matsuura *et al.*, 1993). The sequencing approach defines the scheduling plan for all available jobs, such as the proposed AIS-based hybrid algorithm for static JSSP in this research, while the dispatching approach provides a solution according to the local rules to pick up a certain job from a number of waiting jobs to be operated on the just available machine, like the priority dispatching rules. The switching strategy attempts to make the best use of these two methods. Theoretically, the sequencing approach is more efficient than the dispatching approach in the purely static environment (Matsuura and Kanezashi, 1996). Therefore, the sequencing approach is firstly adopted to deal with the problem with the existing jobs. The operations of each job can be processed in accordance with the predetermined sequences. But when the environment is judged to have deviated from the original plan to an unendurable state caused by the unexpected disturbances, the switching approach is adopted to switch to the dispatching approach.

Specifically, a typical practical job shop environment always begins with the static environment. Initially, the system always has a set of available jobs with previously defined processing time and precedence sequence to be processed. The jobs are expected to be processed in accordance with the complete scheduling plan generated by the AIS-PSO hybrid algorithm for the entire planning horizon. This is regarded as a sequencing approach. During the processing period, uncertainties arise. Events occur unexpectedly at unpredictable moments. Despite these disruptions, the whole system is still relatively static as relatively few disturbances occur over time. Now the job shop turns to a semi-dynamic situation. To reduce the negative effects coming from the disruptions, the extended DCA algorithm developed for semi-dynamic JSSP is applied to test whether rescheduling is necessary and to determine when to trigger such rescheduling operation. From the view point of shop management and administrative cost, it is desirable to keep the initial scheduling plan unchanged as much as possible. Therefore, if the rescheduling process is triggered too often, the current scheduling environment as a result will then deviate from the original scheduling plan and this cannot be neglected. When this happens, it is time to switch from the sequencing approach to the dispatching approach to deal with such more frequently occurred disturbances. The algorithm with priority dispatching rules for dynamic online JSSP is then activated. Once the job shop environment returns to a relatively stable state as indicated by the level of disturbances detected, the scheduling method will be switched back to the sequencing approach. This scheduling strategy is illustrated in Fig. 3.

The means to measure the frequency that the disturbances occur and the relatively stability of a job shop are determined according to the specific operating conditions in the actual situation, customer demand and operator's subjective judgment. For example, such level of disturbances can be computed by the frequency of the disturbances arrivals over a certain period, or the time between the arrivals of two successive disruptions.

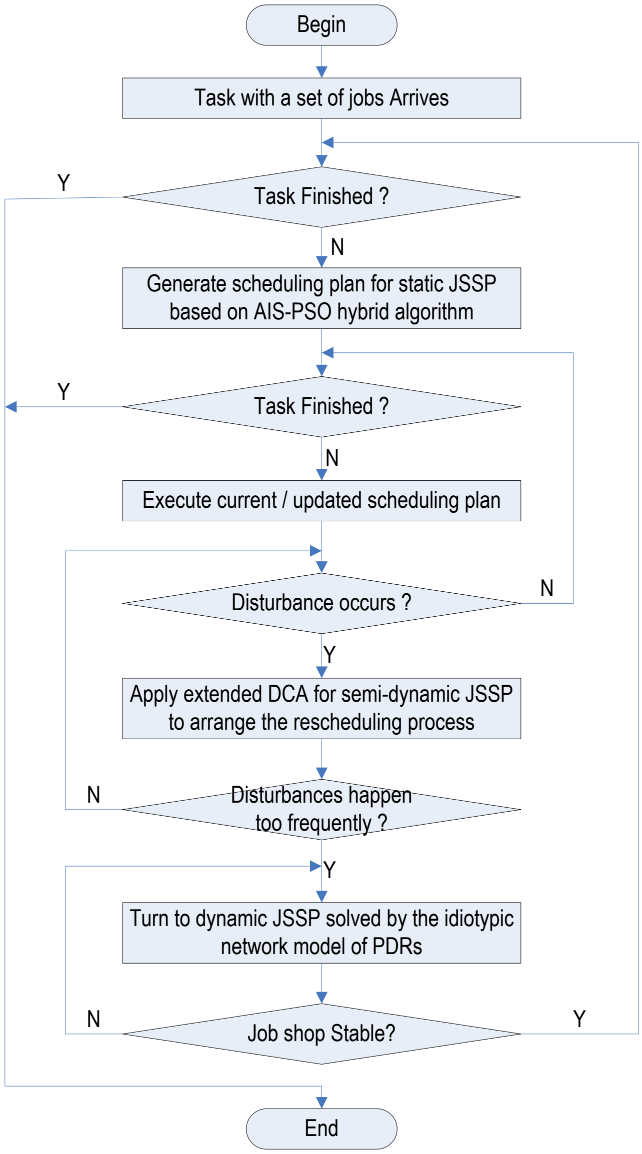

Fig. 3. The strategy for solving generic JSSP

5. A UNIFIED JOB SHOP SCHEDULING FRAMEWORK

To implement this scheduling strategy for a typical job shop system, a unified scheduling framework of a job shop is designed that consists of four main modules: job shop database module, algorithm execution module, job shop processing & controlling module, and job shop evaluation module. These modules collaborate with each other to keep the job shop scheduling system to operate in an efficient way. The structure of the framework is shown Fig. 4 and the details of the four modules are described in the following paragraphs.

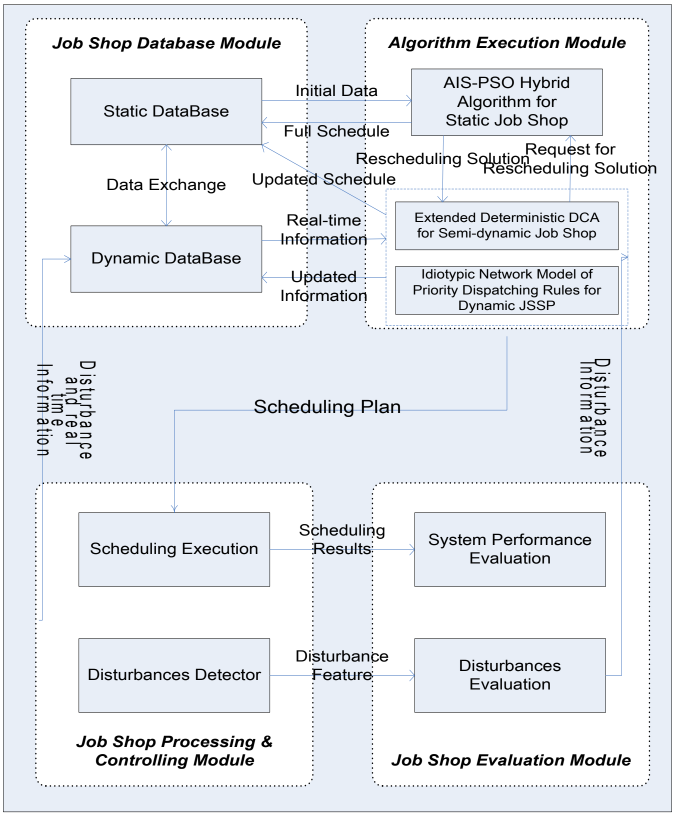

Fig. 4. The unified scheduling framework

5.1 Job Shop Database Module

The module holds key information of the resources. It is divided into two sub-modules: the "Static Database" sub-module and the "Dynamic Database" sub-module. The "Static Database" sub-module stores the initial information such as precedence order and number of operations of the jobs before they are processed. This information is delivered to the "AIS-PSO Hybrid Algorithm for Static Job Shop" sub-module to generate the complete initial scheduling plan. The generated scheduling plan is saved in the "Static Database" sub-module. The "Dynamic Database" sub-module stores the real-time information and traces the system state during the production process, such as the operation of each job currently being processed and the slack time of each job. The "Dynamic Database" sub-module only stores information of the unfinished operations of the existing schedule. All relevant real time information is obtained from the "Scheduling Execution" sub-module. Moreover, when unexpected events arrive, the "Dynamic Database" sub-module receives the disturbance information from the "Disturbances Detector" sub-module and the updated scheduling information from the "Algorithm Execute" sub-module, the updated scheduling plan will then be stored in the "Static Database" sub-module.

5.2 Algorithm Execution Module

The three classes of job shop scheduling problems are solved by the three different AIS-based algorithms developed in the research. The AIS-based hybrid algorithm with PSO receives the input from the "Static Database" sub-module with the completely static information of all jobs and machines, and outputs the full scheduling plan for the scheduling system. When this module receives the disturbances information from the "Disturbances Evaluation" sub-module in the "Job Shop Evaluation Module", the corresponding algorithms for the dynamic situations are activated. Both algorithms need to acquire real-time job shop information from the "Dynamic Database" sub-module. If the "Disturbance Evaluation" sub-module shows that the system is still in a stable state and this disruption is only an exception, the extended DCA is triggered to initiate the rescheduling process. To reschedule the job shop, the AIS-based hybrid algorithm with PSO for static JSSP is requested to generate a new scheduling plan. If the "Disturbance Evaluation" sub-module considers that the system is changing frequently, the idiotypic network model of priority dispatching rules for dynamic JSSP is deployed to generate the scheduling plan. All newly generated or updated scheduling plans are sent to "Scheduling Execution" sub-module to inform the production operation of the system and subsequently stored in the "Static Database" sub-module, while the currently updated system status with the disturbances is stored in the "Dynamic Database" sub-module.

5.3 Job Shop Processing & Controlling Module

This module is responsible for executing the scheduling process according to the scheduling plans provided by "Algorithm Execution Module" and simultaneously monitors the system. All the real-time information including the finished operations, the completed jobs, and existence of unexpected events are collected by this module is delivered to the "Dynamic Database" sub-module. Additionally, "Algorithm Execution Module" sends the scheduling results to "System Performance Evaluation" sub-module, while "Disturbance Detector" sub-module monitors the existence of the disruptions and collects disturbance related information for the "Disturbance Evaluation" sub-module.

5.4 Job Shop Evaluation Module

This module evaluates the system performance and the real-time disturbances. The "System Performance Evaluation" sub-module in this module behaves as an interface between the system and the user. According to its outputs, the users are able to observe real-time system status and obtain the underlying and macroscopic knowledge about the system. The "Disturbance Evaluation" sub-module evaluates the disturbance characteristics, measures the occurring frequency of the disturbances, classifies and quantifies the disturbances for the "Scheduling Execution" sub-module to determine the actions to be taken.

6. CONCLUSIONS

A scheduling strategy is introduced to model a realistic job shop where the states of a job shop changes over time dynamically during operation. Based on this scheduling strategy, a unified framework of a job shop system is designed to provide a holistic approach to solve generic job shop scheduling problems. The framework consists of four modules, namely, job shop database module, algorithm execution module, job shop scheduling processing & controlling module, and job shop evaluation module. These modules collaborate with each other to operate the job shop scheduling system efficiently, and also report the real time system performance to the users via the "System Performance Evaluation" sub-module, which helps the managers monitor and control the whole scheduling system at any time. In summary, this research will not only advance the methodology for JSSP but also expand the application domain and strategies of AIS theories in solving practical industrial problems

REFERENCES

Bagheri, A., M. Zandieh, I. Mahdavi and M. Yazdani (2010). An artificial immune algorithm for the flexible job-shop scheduling problem. *Future Generation Computer Systems*, **26**(4), 533-541.

Banharnsakun, A., B. Sirinaovakul and T. Achalakul (2011). Job shop scheduling with the best-so-far ABC. *Engineering Applications of Artificial Intelligence*, **25**(3), 583-593.

Binato, S., W.J. Hery, D.M. Loewenstern and M.G.C. Resende (2002). A GRASP for job shop scheduling. *Essays and Surveys in Metaheuristics*, **15**, 59-79.

Chandrasekaran, M., P. Asokan, S. Kumanan, T. Balamurugan, and S. Nickolas (2006). Solving job shop scheduling problems using artificial immune system. *International Journal of Advanced Manufacturing Technology*, **31**(5-6), 580-593.

Coello, C.A.C., D. Rivera and N.C. Cortes (2003). Use of an artificial immune system for job shop scheduling. *Proceedings of the 2nd International Conference on Artificial Immune Systems*, **2787**, 1-10.

de Castro, L.N. and J. Timmis (2002). *Artificial immune systems: A new computational intelligence approach.* New York: Springer-Verlag.

Dellacroce, F., R. Tadei and G. Volta (1995). A genetic algorithm for the job-shop problem. *Computers & Operations Research*, **22**(1), 15-24.

Forrest, S. and C. Beauchemin (2007). Computer immunology. *Immunological Reviews*, **216**, 176-197.

Ge, H.W., W.L. Du, Q. Feng and W. Lu (2007). An intelligent hybrid algorithm for job-shop scheduling based on particle swarm optimization and artificial immune system. *Analysis and Design of Intelligent Systems Using Soft Computing Techniques*, **41**, 628-637.

Greensmith, J., U. Aickelin and S. Cayzer (2005). Introducing dendritic cells as a novel immune-inspired algorithm for anomaly detection. *Proceedings of the 4th*

International Conference on Artificial Immune Systems, **3627**, 153-167.

Hart, E. and J. Timmis (2008). Application areas of AIS: the past, the present and the Future. *Applied Soft Computing*, **8**(1), 191-201.

Jeme, N. (1974). Towards a network theory of the immune system. *Ann Immunol* (*Paris*), **125**(1-2), 373-389.

Kumar, A., A. Prakash, R. Shankar and M.K. Tiwari (2006). Psycho-Clonal algorithm based approach to solve continuous flow shop scheduling problem. *Expert Systems with Applications*, **31**(3), 504-514.

Luh, G.C. and C.H. Chueh (2009). A multi-modal immune algorithm for the job-shop scheduling problem. *Information Sciences*, **179**(10), 1516-1532.

Matsuura, H. and M. Kanezashi (1996). Makespan comparison between resequencing and switching in a dynamic manufacturing environment. *International Journal of Production Economics*, **44**(1), 137-149.

Matsuura, H., H. Tsubone and M. Kanezashi (1993). Sequencing, dispatching and switching in a dynamic manufacturing environment. *International Journal of Production Research*, **31**(7), 1671-1688.

Niu, Q., B. Jiao and X.S. Gu (2008). Particle swarm optimization combined with genetic operators for job shop scheduling problem with fuzzy processing time. *Applied Mathematics and Computation*, **205**(1), 148-158.

Peteghem, V.V. and M. Vanhoucke (2009). An artificial immune system for the multi-mode resource-constrained project scheduling problem. *Evolutionary Computation in Combinatorial Optimization*, **5482**, 85-96.

Pinedo, M.L. (2012). *Scheduling: theory, algorithms, and systems*: Springer-Verlag.

Ponnambalam, S.G., N. Jawahar and P. Aravindan (1999). A simulated annealing algorithm for job shop scheduling. *Production Planning & Control*, **10**(8), 767-777.

Puris, A., R. Bello, Y. Trujillo, A. Nowe and Y. Martinez (2007). Two-stage ACO to solve the job shop scheduling problem. *Progress in Pattern Recognition, Image Analysis and Applications, Proceedings*, **4756**, 447-456.

Qiu X. and H.Y.K. Lau (2013). An AIS-based hybrid algorithm with PDRs for multi-objective dynamic online job shop scheduling problem. *Journal of Applied Soft Computing*, **13**(3), 1340 – 1351.

Qiu X. and H.Y.K. Lau (2012). An AIS-based hybrid algorithm for static job shop scheduling problem. *International Journal of Intelligent Manufacturing*, (2012).

Timmis, J. (2007). Artificial immune systems: today and tomorrow. *Natural Computing*, **6**(1), 1-18.

Wang, L., and D.B. Tang (2011). An improved adaptive genetic algorithm based on hormone modulation mechanism for job-shop scheduling problem. *Expert Systems with Applications*, **38**(6), 7243-7250.

Watson, J.P., J.C. Beck, A.E. Howe and L.D. Whitley (2003). Problem difficulty for tabu search in job-shop scheduling. *Artificial Intelligence*, **143**(2), 189-217.

Wojtyla, G., K. Rzadca and F. Seredynski (2006). Artificial immune systems applied to multiprocessor scheduling. *Parallel Processing and Applied Mathematics*, **3911**, 904-911.

Wong, L.P., C.Y. Puan, M.Y.H. Low and Y.W. Wong (2010). Bee colony optimisation algorithm with big valley landscape exploitation for job shop scheduling problems. *International Journal of Bio-Inspired Computation*, **2**(2), 85-99.

Yao, B.Z., C.Y. Yang, J.J. Hu, G.D. Yin and B. Yu (2010). An improved artificial bee colony algorithm for job shop problem. *Applied Mechanics and Materials*, **26**, 657-660.

Zobolas, G., C. Tarantilis and G. Ioannou (2008). Exact, heuristic and meta-heuristic algorithms for solving shop scheduling problems. *Metaheuristics for Scheduling in Industrial and Manufacturing Applications*, **128**, 1-40.