J Intell Manuf (2007) 18:505–511
DOI 10.1007/s10845-007-0054-4

Combining data mining and Game Theory in manufacturing strategy analysis

Yi Wang

Published online: July 2007

Abstract The work presented in this paper is result of a rapid increase of interest in game theoretical analysis and a huge growth of game related databases. It is likely that useful knowledge can be extracted from these databases. This paper argues that applying data mining algorithms together with *Game Theory* poses a significant potential as a new way to analyze complex engineering systems, such as strategy selection in manufacturing analysis. Recent research shows that combining data mining and *Game Theory* has not yet come up with reasonable solutions for the representation and structuring of the knowledge in a game. In order to examine the idea, a novel approach of fusing these two techniques has been developed in this paper and tested on real-world manufacturing datasets. The obtained results have been indicated the superiority of the proposed approach. Some fruitful directions for future research are outlined as well.

Keywords Data mining · Game Theory · Manufacturing strategy analysis · Nash equilibrium · Formal concept

Introduction

Game Theory is becoming more and more important and widely used as a tool to select strategies. The complex behaviour of the agents and a huge amount of data and information generated from manufacturing environment will made the agents very difficult to deicide which strategy they should select correctly. *Game Theory* is not easy to apply to manufacturing engineering due to the use of mathematical optimization, which is not sufficient to dealing with the vast data. Thus new techniques are needed to improve the over all performance of the game and relief the load of data and information from the agents.

This paper proposes an intelligent approach applying data mining techniques on the game theoretical data to extracting a few but important information and knowledge in order to assist in a better decision-making. This paper further suggests a new methodology to structure the knowledge of a game.

The paper is organized as follows: Section "Literature review" gives a brief literature review. In Section "Drawbacks of Game Theory", drawbacks of the fundamental assumption of the game theory, is discussed. A new framework for developing game mining process is presented in Section "Game mining". Section "Mining algorithm" describes a new way to structure the game knowledge. Section "An illustrative example" gives an illustrative example. Finally, concluding remarks are drawn and future research directions are highlighted in Section "Conclusion".

Literature review

In manufacturing engineering, *Game Theory* has been used in many areas. Golden and Dollinger (1993) discussed the uses of *Game Theory* in cooperative alliances and competitive strategies in small manufacturing firms. Lygeros, Godbole, and Sastry (1996) used *Game Theory* to design hybrid controllers for complex systems, which are ideally suited to a multi-agent setting. A *Game Theory* approach for co-op advertising models in manufacturer-retailer supply chains was introduced (Huang & Li, 2001). Guan (1995) and Shen (2002) realized the game-like nature of independent scheduling decisions and tried to use *Game Theory* to make their agents smarter. The approach and the theory have been

Y. Wang (✉)
Manufacturing Engineering Centre,
Cardiff University,
Queen's Buildings, The Parade, Newport Road,
Cardiff, Wales CF24 3AA, UK
e-mail: sceyw2@cf.ac.uk

applied to manufacturing scheduling as well. These are only a few of examples among quite a number of research papers published.

Drawbacks of Game Theory

Fail of rationality

An essential assumption in classical game theoretic analysis is *rationality*. "*The capacity of the human mind for formulating and solving complex problems is very small compared with the size of the problem whose solution is required for objectively rational behaviour in the real world or even for a reasonable approximation to such objective rationality*". (Simon, 1957)

This assumption does not guarantee a uniqueness and well-understood solution. Often the solution is irrational and indicates that a player might pick one of the several strategy combinations.

The assumption of rationality is also rather ambitious. It implies that decision-makers are capable of having complete knowledge of relevant information, perfect anticipation of future consequences, and the cognitive capacity and time available in order to do so. In reality, none of these assumptions are sufficiently met (Sterman, 2000).

A rational behaviour is difficult to be addressed because people might not always behave rationally. Irrational behaviour may follow from the following factors (Meyer & Booker, 1991):

- Failure to update their action of new information that becomes available.
- Acting on perceptions that are in fact false.
- Acting on personal agendas.
- Poor understanding of interdependencies between events.

The challenge is to capture the relevant causal behaviour between players without making the process too complex. Selecting the important relationships rather than the less important ones can only be done based on experience or learning from historical data.

Although *Game Theory* is fairly well developed, there are many aspects that still need to be explored. For example, it is common knowledge that the solution to the prisoner's dilemma is that both prisoners should confess. However in real life it is not always true and many similar situations yield different results. The following section illustrates a special case of the problem.

Designer's dilemma

The *Designer's Dilemma* (Pham, Wang, & Dimov, 2005,2006) referred in this section is adapted from the "*Prisoner's Dilemma*" (Axelrod, 1984) that is used as a classical example to illustrate the fundamentals of *Game Theory.*

Table 1 Designer's dilemma

Designer 1	Designer2	
	Accept	Refuse
Accept	(500, 500)	(1000, 0)
Refuse	(0, 1000)	(100, 100)

Imagine that two designers are available for a given design task. However, there is a manufacturer who does not have sufficient resources for both designers. Two designers are isolated from each other. The manufacturer visits each of them and offers them a deal for designing a perfect product.

If only one designer accepts the deal, this designer will carry out the whole project and receive all the payment. If neither of them accepts the offer, the manufacturer will keep them for future projects, thus paying them a royalty. However, if one of them refuses the job, that designer takes no further part in the process and receives no payment.

If both are accepted, they will be working together, and each of them will receive less than payment when one had refused the job. The dilemma resides in the fact that each designer has a choice between only two options, but cannot make a good decision without knowing what the other one will do. This is shown in Table 1.

The example shows a one shot strategy profile setup according to *Game Theory*. As in the classical *Prisoner's Dilemma, each* of the players is tempted to deny (refuse). However, due to the uncertainty with the opponent's action, a rational player would realize that the opponent might confess (accept), thus leaving him with the worst payoff. To be on the safe side he will choose to confess (accept). If both players are rational, the solution to the *Prisoner's Dilemma* is that both should confess (accept).

In real life strategy selection is based on a player's knowledge of the opponent. If a player has access to all the knowledge, then the player has a full control of the *Game*. This knowledge may be gathered from past records by applying data mining methods. A *Dynamic Causal Mining* (DCM) (Pham et al., 2005, 2006) is applied in this work to extract knowledge to assist in strategy selection with the form of causal rules among historical data of strategies.

Game mining

Rule-based Game Theory

The *Game Theory* originated from the work of Von Neumann and Morgenstern (1964). There are two ways to classify *Game Theory* either by corporative/non-corporative (Von Neumann & Morgenstern, 1964) or rule/freewheeling based (Kleindl, 1999). In rule-based *games* where players interact according to specified "rules of engagement". A game consists of a set of rules governing a competitive situation. The rules can be represented by variables reflecting individuals or groups of individual's selectable strategies

A rule in a rule-based game is common if it is known to all players, or is hidden otherwise. The hidden rules can be revealed by techniques such as data mining. Different data mining techniques will uncover different kind of rules.

These rules might come from contracts, loan covenants, or trade agreements. In freewheeling *games*, players interact without any external constraints. For example, buyers and sellers may create value by transacting in an unstructured fashion. Business is a complex mix of both types of *games*. For rule-based *games*, *Game Theory* offers the principle, where to every action, there is a reaction.

Game levels

A game can be played at two levels: micro level and macro level.

Micro level is concrete and closely associated with the properties of the game world. Actions are related to smallest entities; therefore the techniques must react to short-term and real-time game play. Because the decisions are made for a short term, less accurate rules does not necessary lead to total failure of the game. For instance, if the first sets of selected strategy sequences showed wrong result, there is still time and resource to alter the sequences.

On the Macro level the decision is made over a long period of time. The amount of data can be large, and, therefore, the main problem is to filter it down to a suitable form. Due to quantization, information can be lost and the problem is to ensure that no vital knowledge kept. Decision-making in macro level is for speculation and the cost of a wrong decision is high.

Because decisions are made more frequently on micro level than macro level, the requirement of quality at the micro level cannot be as high as on the macro level. The game is a generator that produces events and states. Modelling the game means recognizing the underlying dependencies between the players. Previous work in game pattern recognition and game mining (Tveit et al., 2002) focuses on the dependencies between the players in the sequence and claim that it is sufficient to consider short term modelling.

Game mining stages

This section provides a general framework for *game mining.* It is an iterative and continual process of discovering game rules, formulating game policies, testing the results and revising the models.

Stage 1: Problem definition. In this phase, the business issues of a problem are identified based on previous game play. The game does not limit to computer or board game. The game could include conflict between buyers and seller or disagreement between agents or bargaining. This process defines and states and the key variables. The time horizon for the problem is also defined so that the cause and effects can be identified. This stage describes the details about players and decides which of their individual's strategies should be included.

Stage 2: Data preparation. Necessary data are collected from various sources based on the defined problem. A homogeneous data source should be created by resolving the representation and encoding the differences.

Stage 3: Data mining. This stage involves transforming the game data into useful rules. Different types of tools are tested for extracting the rules.

Stage 4: Policy formulation. Formulated policies are created by the rules that have been extracted by the use of mining techniques. These policies should be combined together if strategies are relevant for each player.

Stage 5: Model representation. This stage validates the accuracy of the model by testing the policies for a new real world problem. The model is able to predict results for new cases and then the results of the prediction are used to alter/improve the result of the game. It might be necessary to go back to gather more data so that better decisions can be made faster and accurately. This stage also involves the comparison of the output of the model with the real behaviour of the system. Models must also be tested for some extreme conditions that might not even occur in the real system.

Game behaviour

There are three fundamental modes of behaviour and three derived modes of behaviour. Each of these modes is generated from a particular type of relation between players, which includes a sympathetic and an antipathetic.

Sympathetic behaviour

Sympathetic behaviour is caused by an increase or decrease in game outcome. It reinforces a change made by players with more change in the same direction. This can lead to rapid growth, e.g. in a virus population, which could be difficult to stop.

Examples of sympathetic behaviour can be found in manufacturing. For instance, as funds are invested to increase the capacity of a plant, more products will be manufactured which will generate more funds which can be again invested to create more capacity.

Antipathetic behaviour

Antipathetic behaviour represents an adjustment to achieve a certain goal or objective. It indicates entities attempting to change from its current state to a goal state.

This implies that if the current state is above the goal state, then the system forces it down. If the current state is below a goal state, the system pushes it up. Antipathetic behaviour provides useful stability but resists external changes.

When there is a difference between the goal state and the current state, a gap is created. A feedback signal is generated that tends to reduce that difference, the larger the gap the larger the feedback signal. The signal will continue to exist as long as the difference is non zero.

Oscillatory behaviour arises when significant time delays exist in a *antipathetic* behaviour. Time delays cause feedback to continue after the goal has been attained, which leads to over correction.

Mixed behaviours

There are many basic modes of behaviour and combining *sympathetic* and *antipathetic* behaviour creates even more modes. One of the most commonly observed behaviour patterns in complex and dynamic systems is S-shaped growth. S-shaped growth is the result of interaction between an *exponential growth* and a *goal seeking* behaviour. After a start-up period, the growth is rapid but it gradually slows down.

Game representation

The essential for game mining is the development of a suitable structure of game concepts, so that the discovered knowledge can be formulated properly. The previous works on the chess database has illustrated the need for an appropriate structure for learning (Kleindl, 1999; Tveite et al., 2002). A knowledge representation for game concepts could also contribute significantly to other research related to *Game Theory*, such as collaboration or conflict analysis.

The characteristic of such a representation has to be sufficiently expressive for formulating abstract strategic concepts. It should be easily understood and efficiently implemented by a user.

Classical *game* uses game trees and matrices as solution form. Game trees and matrices provide indeed a very detailed description of the *game*. It is however rather non-practical since the size of the tree becomes very huge, even for a simple game, while the matrix form is too simplified. An attempt to provide a complete description of a complex *game* like *Bridge*, would lead to a combinatorial explosion in tree nodes and thousands of game matrix.

In this paper the concept of game is represented differently than the classical game. The new representation still has normal and extensive form; however the table and tree are replaced with table and lattice.

Table 2 The normal representation of the game mining

P2	P1		
	Strategy 1	Strategy 2	Strategy 3
Strategy 1	a		
Strategy 2		s	a
Strategy 3	a	s	a

A game concept can be defined as a combination of strategies by players. Given a set of strategies $S_{p1} = (s_{1,p1}, s_{2,p1}, \ldots, s_{m,p1})$ of player 1, where some strategies related to some strategies in $S_{p2} = (s_{1,p2}, s_{2,p2}, \ldots, s_{m,p2})$ of player 2. If $s_{i,p1}$ is dynamic causally related to $s_{i,p2}$, then such a pair $(s_{i,p1}, s_{i,p2})$ is called a game concept.

The normal representation of the game concept

A game context of the form $K = (S_{\mathrm{A}}; S_{\mathrm{B}}; C)$ where S_{A} and S_{B} are strategies that belong to player A and B, respectively; Moreover, $C(S_{\mathrm{A}}, S_{\mathrm{B}})$ is a relation between the sets of strategies that belong to player A and B, respectively.

A formal game context in its normal form can be considered as a table, which relates a record in S_{A} and S_{B}. The entries in the table indicate by a polarity combination of S_{A}, the name of which precedes the corresponding row, is causally related to S_{B}, the name of which is at the top of the corresponding column.

Table 2 shows the normal form representation of a game in which player's causal relationship is already retrieved through DCM; s stands for sympathetic and a stands for antipathetic. In this particular example, player 1's strategy 2 is sympathetically related to player 2's strategy 2 and 3. If player 1 considers changing strategy 2, he has to concern about the causal effect and weather they are desirable.

The extensive representation of the game concept

An extensive form described in Fig. 1 is a graphical visualization of the normal representation. The extensive form consists of nodes and lines. Each node represents one strategy of the given context.

The relations and the information of the context can be read from the extensive form. It allows interpretation of relationships between strategies selected by players. This includes object hierarchies, if they exist in the given context.

An extensive form contains the relationships between strategies selected by different players and thus it is an equivalent representation of a context, i.e. it contains exactly the same information as the cross table. Dependencies and relation-

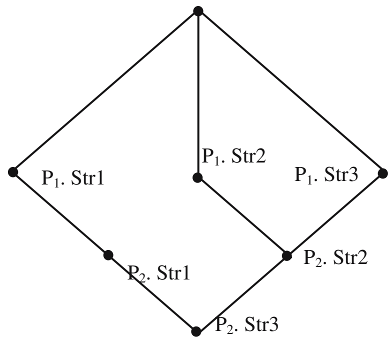


Fig. 1 The extensive representation of the game mining

ships between attributes can be easily detected in an extensive form.

The extension of concepts is retrieved by tracing all paths that lead down from the node to collect the formal objects. In this example, the formal object is strategy 3 of player 2. To retrieve the intension of a formal concept one needs to trace all paths which lead up in order to collect all the formal attributes. In this example, there is the empty node. Thus the concept with the extension "Strategy 3 player 2" covers all the strategies used by player 1.

"Strategy 2 player 2" is a sub-concept of "Strategy 3 player 2" because it only covers 2 strategies from player 1, thus the extension of "Strategy 2 player 2" is a subset of the extension of "Strategy 3 player 2" and the intension of "Strategy 2 player 2" is a superset of the intension of "Strategy 3 player 2". All edges in the extensive form of a concept lattice represent this sub-concept-super-concept relation.

The top and bottom concepts in a concept lattice are special. The top concept has all formal objects in its extension. Its intension is often empty but does not need to be empty. The bottom concept has all formal attributes in its intension. If any of the formal attributes mutually exclude each other then the extension of the bottom concept must be empty. The top concept can be thought of as representing the "universal" concept and the bottom concept the "null" or "contradictory" concept of a formal context.

Mining algorithm

The algorithm is based on the work of association and dynamic mining (Agrewal et al., 1995; Pham, Wang, & Dimov, 2005,2006) and the detail of the algorithm is shown in Appendices 1 and 2. It supports the following operations:

(1) To add and read new attributes.
(2) To maintain a counter for each polarity with respect to every dynamic value set. While making a pass, one dynamic set is read at a time and the polarity count of candidates supported by the dynamic sets is incremented. The counting process must be very fast as it is the bottleneck of the whole process.

Let D denote a database which contains a set of n records with attributes $\{A_1, A_2, A_3, \ldots A_m.\}$, where each attribute is of a unique type (sale price, production quantity, inventory volume, etc). Each attribute is linked to a time stamp t. The records are arranged in a temporal sequence ($t = 1, 2, \ldots, n$). The causality between attributes in D can be identified by examining the polarities of corresponding changes in attribute values. Let D_{new} be a new dataset constructed from D. A generalized dynamic association rule is an implication of the form $A_1 \rightarrow^p A_2$, where $A_1 \subset D$, $A_1 \subset D$, $A_1 \cap A_2 = \phi$ and p is the polarity.

The algorithm makes two passes over the data. In the first pass, the *support* of individual attributes is counted and the frequent attributes are determined. The dynamic values are used for generating new potentially frequent sets and the actual *support* of these sets is counted during the pass over the data. In subsequent passes, the algorithm initializes with dynamic value sets based on dynamic values found to be frequent in the previous pass. After the second of the passes, the *causal rules* are determined, and they become the candidates for the dynamic policy. In the DCM process, the main goal is to find the strong dynamic causal rule in order to form a policy. It also represents a filtering process that prunes away static attributes, which reduces the size of the dataset for further mining.

The process time would be $n^2 - n$, where n is the number of attributes. This becomes a huge problem if n becomes too large. It is obvious that the task becomes much simpler if the size of n could be reduced before the search.

An illustrative example

This example is partially taken from the DCM paper (Pham et al., 2006). It illustrates the causal relationship between different players

Table 3 shows the original pre-processed database. ΔP1, …, ΔP7 indicate the names of 'dynamic' attributes, in this case the players. The numbers in the other columns result after the difference calculation by Eq. (1).

Table 4 illustrates the 'pruned' dynamic database. Pruning is carried out to remove columns (attributes) where the level of *support* is below a set minimum value. In this example, the minimum support is set to two. This implies that columns with two or more zeros are removed.

Table 5 shows the different supports for the pairs of attributes in Table 4. The *supports* are calculated according to user-defined measures (Pham et al., 2006). In order to iden-

Table 3 Original database

ΔP1	ΔP2	ΔP3	ΔP4	ΔP5	ΔP6	ΔP7
+8	+1	+1	+6	+4	0	0
−7	+9	−2	−1	−9	+1	−7
−6	+4	+2	−5	0	−1	−4
+3	+8	−1	−8	−3	−8	+8
−1	−6	0	−3	−6	+2	−5
0	0	−9	−7	−1	−1	−1
+4	+3	+1	0	+8	0	+5
+9	−9	0	−4	0	−1	−7
+1	−5	0	+7	−8	+3	+6

Table 4 Pruned database

ΔP1	**ΔP2**	**ΔP4**	**ΔP7**
+8	+1	+6	0
-7	+9	-1	-7
-6	+4	-5	-4
+3	+8	-8	+8
-1	-6	-3	-5
0	0	-7	-1
+4	+3	0	+5
+9	-9	-4	-7
+1	-5	+7	+6

Table 5 Counting results

	(+,+,+)	(−,−,−)	(−,+,+)	(+,−,−)
P1&P2	0	0	0.1	0
P1&P4	0	0.1	0	0.1
P1&P7	0	0	0	0.2
P2&P4	0	0	0	0.1
P2&P7	0	0	0.1	0.1
P4&P7	0	0.2	0	0

tify the delay and feedback relation, the counting is zig-zag as in Table 4.

Suppose that the threshold *support* is 0.1, the lattice diagram can be drawn as in Fig. 2.

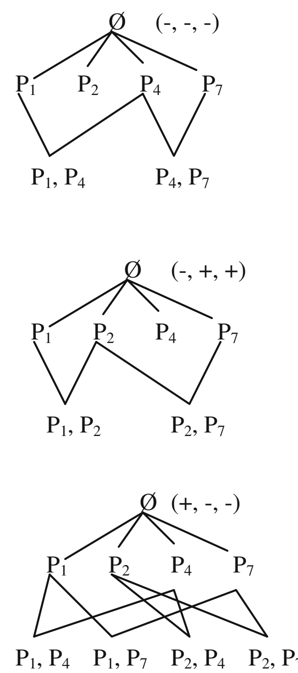
By combining the three lattices, it is clear that (P_2, P_7) is the most frequent set, thus it can be used as a rule. The rule indicates that player 2's strategies are antipathetically causally related to strategies selected by player 7.

Conclusion

The *Game Theory* is itself well developed; however there is still some vagueness and unclearness about the fundamental assumption of rationality. In *Game Theory*, all players have a pattern of behaviour and these patterns are the cause for the individual decision-making. These patterns or rules are often hidden in game data bases and are able to be extracted from the historical datasets by data mining techniques. This paper has developed an intelligent approach integrating data mining techniques and Game Theory to discover the hidden knowledge in game databases in order for players to obtain a better insight to the game involved and the rules developed can enhance their performance and make decision fast, correctly and quickly.

Formal concept lattice is also suggested to structure the knowledge of game for further improving the player understands of the game.

Fig. 2 The lattice diagram of Table 5



Appendix 1. Mining process

Part 1: – Pre-processing: Removal of the "least" causal data from database

Part 2: – Mining: Formation of a rule set that covers all training examples with minimum number of rules

Part 3: – Checking: Check if an attribute pair is self contradicting (sympathetic and antipathetic at the same time)

Appendix 2. Description of DCM algorithm

Part 1:

Input: The original database, the values of the pruning threshold for the neutral, sympathetic and antipathetic *supports*.

Output: Dynamic sets

Step 1: Check the nature of the attributes in the original database (numerical or categorical).
Initialize a new database with dynamic attributes based on the attributes and time stamps from original database.
Step 2: Initialize a counter for each of the three polarities.

Part 2:

Input: The processed database, the values of the pruning threshold for the supports of the polarity combinations.
Output: Dynamic sets
Step 1: Count single attributes
Prune away those attributes below the user defined threshold.
Step 2: Count attributes from the pruned database pair-wise.
Prune away those attributes pairs below the user defined threshold.

Part 3:

Input: The mined database, the values of the pruning threshold for the supports of the polarity combinations.
Output: Dynamic sets
Step 1: Check weather a rule is self-contradictory (a rule is both sympathetic and antipathetic).
Step 2: If step 1 returns true then retrieve the attribute pair form the preprocessed database.
Step 3: Initialize a counter that includes polarity combination.
Step 4: For the pair of attributes
Count the occurrence of polarity combination with two records each time.
Prune away the pairs if the counted support is below the input threshold.

References

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo, A. I. (1995) *Fast discovery of association rules, Advances in Knowledge and Discovery and Data Mining* Cambridge, MA: AAAI/MIT Press, pp. 307–328.

Axelrod, R. (1984). *The evolution of cooperation*. New York: Basic Books.

Huang, Z., & Li, S. X. (2001). Co-op advertising models in manufacturer-retailer supply chains: A Game Theory approach. *European Journal of Operational Research, 3*(135), 527–544.

Golden, P. A., & Dollinger, M. (1993). Cooperative alliances and competitive strategies in small manufacturing firms. *Journal of Entrepreneurship: Theory and Practice, 17*(4), 266–285.

Guan, Z. (1995). Application of decentralized cooperative problem solving in dynamic flexible scheduling, *Proceedings of SPIE*, Vol. 2620, SPIE, Bellingham, Wash., Aug., pp. 179–183.

Kleindl, B. (1999) Game theoretic perspective on market-oriented versus innovative strategic choice. *Journal of Strategic Marketing, 7*(4), 265–274.

Lygeros, J., Godbole, D. N., & Sastry, S. (1996) Multiagent hybrid system design using Game Theory and optimal control, Decision and Control, 1996. In *Proceedings of the 35th IEEE*, Vol. 2, Kobe, Japan, pp. 1190–1195.

Meyer, M. A., & Booker, J. M. (1991). *Eliciting and analyzing expert judgement: A practical guide, knowledge-based systems*(Vol. 5). London: Academic Press.

Simon, H. (1957). *A behavioral model of rational choice.* Santa Monica, Calif: Rand Corporation.

Shen, W. M. (2002). Distributed manufacturing scheduling using intelligent agents, *Proceedings of IEEE Intelligent Systems and Their Applications*, Vol. 17, Issue: 1, Jan/Feb, London, Ont., Canada.

Sterman, J. D. (2000). *Business dynamics*. Boston, MA: Irwin, McGraw-Hill.

Tveit, A., Tveit, G. B., (2002) Game Usage Mining: Information Gathering for Knowledge Discovery in Massive Multiplayer Games. *International Conference on Internet Computing.* pp. 636–642.

Von Neumann, J., & Morgenstern, O. (1964) Theory of games and economic behavior (3rd ed.). New York: Science Editions, John. Wiley.

Pham, D. T., Wang, Y., & Dimov, S. (2005). Intelligent manufacturing strategy selection. In: *Proceedings of Virtual International Conference on Intelligent Production Machine and Systems, (IPROMS 2005)*, 4–15 July 2005, Cardiff, UK, pp. 246–252, Elsevier Science Ltd, ISBN 10: 0080447309

Pham, D. T., Wang, Y., & Dimov, S. (2006). Incorporating delay and feedback in intelligent manufacturing. *Proceedings of Virtual International Conference on Intelligent Production Machine and Systems, (IPROMS 2005)*, 4–15 July 2006, Cardiff, UK (in publishing).