Information Extraction from Solution Set of Simulation-based Multi-objective Optimisation using Data Mining

Catarina Dudas and Amos Ng
Centre for Intelligent Automation
University of Skövde
Box 408, 541 28 Skövde
Sweden

Henrik Boström
Informatics Research Centre
University of Skövde
Box 408, 541 28 Skövde
Sweden

KEYWORDS
Output analysis, Data mining, Information extraction.

ABSTRACT

In this work, we investigate ways of extracting information from simulations, in particular from simulation-based multi-objective optimisation, in order to acquire information that can support human decision makers that aim for optimising manufacturing processes.
Applying data mining for analyzing data generated using simulation is a fairly unexplored area. With the observation that the obtained solutions from a simulation-based multi-objective optimisation are all optimal (or close to the optimal Pareto front) so that they are bound to follow and exhibit certain relationships among variables vis-à-vis objectives, it is argued that using data mining to discover these relationships could be a promising procedure. The aim of this paper is to provide the empirical results from two simulation case studies to support such a hypothesis.

RELATED WORK

Data mining is a technique which has been used in both private and public sectors and clearly with different objectives. Companies within banking, insurance and retailing use data mining to reduce cost, detect frauds and to advertise in more effective ways. Homeland security is yet another application area of growing interest, in which data mining also has been used.
One of the first uses of artificial intelligence in manufacturing applications was accomplished in the 1980's according to (Kusiak 2006). In the beginning of the 1990's, the use of data mining techniques was introduced for production, something which has been growing since then. A comprehensive review of papers considering data mining applications within manufacturing is presented in (Kusiak 2006). Manufacturing operations, fault detection, design engineering and decision support systems have been in focus as research topics, but there is still an enormous potential for further research in other application areas, such as maintenance, layout design, resource planning and shop floor control.
The combination of multi-objective optimisation solutions and data mining techniques is a fairly unexplored area. Therefore the literature reveals quite few reports. (Chiba et al. 2006) and (Jeong et al. 2005) apply the use of analysis of variance (ANOVA) and Self-Organizing Maps (SOM) in the design process for aerodynamic optimisations problems. It is found that the ANOVA obtains the quantitative correlation between objective function and design variable. The result from SOM is qualitative and subjective and can be used for understanding of the design variable influence. Furthermore, SOM explains the trade-off between the competing objectives.

DATA MINING

Data mining is an automated or semi-automated technique used to discover and interpret hidden relationships, patterns or trends in large data sources. A blend of concepts and algorithms from machine learning, statistics, artificial intelligence, and data management are borrowed to the field of data mining.
Figure 1 shows the data mining process as an iterative procedure (Fayyad et al. 1996). The process can be divided into three parts: selection/pre-processing, mining and presentation.

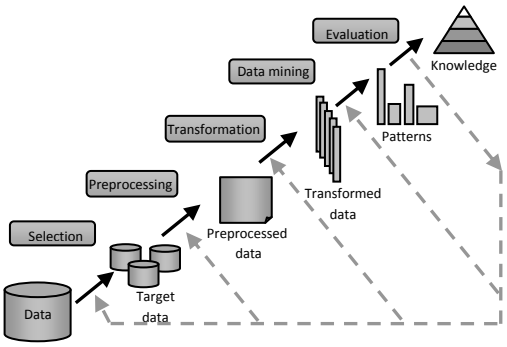

Figure 1: Data Mining Process

The first step involves gathering, organising and cleaning data before it can be used. The mining process involves choosing appropriate method(s) to be used for searching patterns in data. The final step is about how to present the results of the prior processes in a suitable way. After evaluation of the presented results, the entire process, or parts of it, may be re-iterated.

Data mining is a rapidly expanding field with growing interests and importance. Simulation based optimisation is certainly an application area where the use of this technology can provide a significant advantage.

Predictive and Descriptive Data Mining

Data mining techniques have become standard tools to develop predictive and descriptive models in situations where one wants to exploit data collected from earlier observations in order optimise future decision making (Witten and Miller 2005). In the case of predictive modelling, one typically tries to estimate the expected value of a particular variable (called the dependent variable), given the values of a set of other (independent) variables. In the case of a nominal dependent variable (i.e., the possible values are not given any particular order), the prediction task is usually referred to as classification, while the corresponding task when having a numerical dependent variable is referred to as regression. One usually wants the model to be as correct as possible when evaluated on independent test data, and several suggestions for how to measure this have been proposed. For classification, such measures include accuracy, i.e., the percentage of correctly classified test examples, and the area under the ROC curve (AUC), i.e., the probability that a test example belonging to a class is ranked as being more likely belonging to the class than a test example not belonging to the class (Provost et al. 1998). Besides the ability to make correct predictions, one is also often interested in obtaining a comprehensible (descriptive) model, so that the reasons behind a particular classification can be understood, and also that one may gain insights into what factors are important for the classification in general. Examples of such comprehensible models are decision trees and rules, e.g. (Quinlan 1993), while examples of models not belonging to this group, often called black-box, or opaque, models, include artificial neural networks and support vector machines; see e.g. (Hastie et al. 2001).

Decision Trees and Ensembles

Techniques for generating decision trees are perhaps among the most well-known methods for predictive data mining. Early systems for generating decision trees include CART (Breiman et al. 1984) and ID3 (Quinlan 1986), the latter being followed by the later versions C4.5 (Quinlan 1993) and C5.0 (Quinlan 1997). The basic strategy that is employed when generating decision trees is called recursive partitioning, or divide-and-conquer. It works by partitioning the examples by choosing a set of conditions on an independent variable (e.g., the variable has a value less than a particular threshold, or a value greater or equal to this threshold), and the choice is usually made such that the error on the dependent variable is minimised within each group. The process continues recursively with each subgroup until certain conditions are met, such as that the error cannot be further reduced (e.g., all examples in a group belong to the same class). The resulting decision tree is a graph that contains one node for each subgroup considered, where the node corresponding to the initial set of examples is called the root, and for all nodes there is an edge to each subgroup generated from it, labelled with the chosen condition for that subgroup.

Decision trees have many attractive features, such as allowing for human interpretation and hence making it possible for a decision maker to gain insights into what factors are important for particular classifications. However, recent research has shown that significant improvements in predictive performance can be achieved by generating large sets of models, or ensembles, which are used to form a collective vote on the value for the dependent variable (Bauer and Kohavi 1999). It can be shown that as long as each single model performs better than random, and the models make independent errors, the resulting error can in theory be made arbitrarily small by increasing the size of the ensemble. However, in practice it is not possible to completely fulfil these conditions, but several methods have been proposed that try to approximate independence, and still maintain sufficient accuracy of each model, by introducing randomness in the process of selecting examples and conditions when building each individual model. One popular method of introducing randomness in the selection of training examples is bootstrap aggregating, or bagging, as introduced by (Breiman 1996). It works by randomly selecting n examples with replacement from the initial set of n examples, leading to that some examples are duplicated while others are excluded. Typically a large number (at least 25-50) of such sets are sampled from which each individual model is generated. Yet another popular method of introducing randomness when generating decision trees is to consider only a small subset of all available independent variables at each node when forming the tree. When combined with bagging, the resulting models are referred to as random forests (Breiman 2001), and these are widely considered to be among the most competitive and robust of current methods for predictive data mining. The

drawback of ensemble models are however that they can no longer be easily interpreted and hence provide less guidance into how classifications are made.
The Rule Discovery System™ (RDS) addresses this problem by providing some insight into what factors are of importance in an ensemble of decision trees by presenting the variable importance of each independent variable, i.e., how much the variable, relative to all other variables, contributes to reducing the squared error of the dependent variable.

TWO DATA MINING APPLICATIONS

The use of data mining in manufacturing applications can have different aims and purposes. In this paper a specific approach is presented: data mining for identifying patterns in data sets generated by multi-objective optimisation. The first data set handles a buffer allocation problem and the second is for identifying dispatching rules setting in a production line. The data mining software used is RDS for both studies.

Buffer Allocation

The simulation model used for this study was developed by a simulation optimisation system called FACTS Analyser (Ng et al. 2007). The model of the production line consists of 5 stations and 5 buffers and is controlled by a Critical Work-In-Process (CWIP) strategy. A Pareto Front was found after 40 generations of multi-objective optimisation (MOO) with MA-NSGA-II.
There is a constraint for each individual buffer size ($0 \leq$ Buffer Size ≤ 50) and no constraint on the total buffer size. The CWIP level varies between [0-100] in terms of percent. For the MOO the objectives are to minimise lead time (LT) and maximise throughput (TP).
The data set in RDS™ consists of 6 input variables (Buffer capacity 1-5 and CWIP) and two output variables, LT and TP. For each generation there are 200 observations and in this study the set for generation 39 (G39) is explored. As validation method 10-fold cross validation is used.
Since the data mining software make predictions with one output variable at time, LT and TP have to be studied separately. The algorithms used in these experiments are trees and ensembles of trees. Trees are useful for interpretation of the important input variables and the benefits of an ensemble of trees are that you receive a model with higher correlation and lower error rate.

Results for the buffer allocation case

The optimal solution data set for G39 is as close as possible to an optimal Pareto front. In order to find the key information in the data set the importance score and the decision tree has to be examined. The importance score plot enlightens the variables that are most informative, i.e. contribute to the model primarily. On the other hand, the decision tree can be used to illuminate more detailed information about the settings of the variables.
The importance score for G39 can be found in Figure 2. It reveals that for both LT and TP the most informative variables are buffer capacity 1 and 2 (B1, B2), where B1 is dominating.

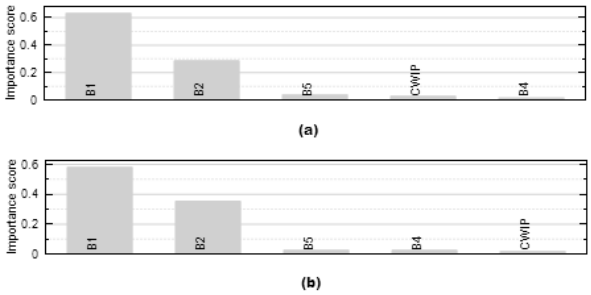

Figure 2: Importance Score for G39 (a) LT, (b) TP

Figure 3 and 4 show the decision tree with TP and LT as output variable where the detailed information can be found.

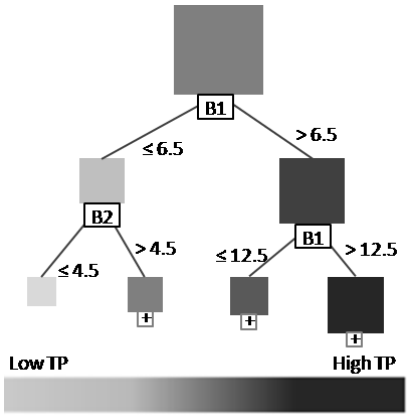

Figure 3: Decision Tree with TP as Output Variable

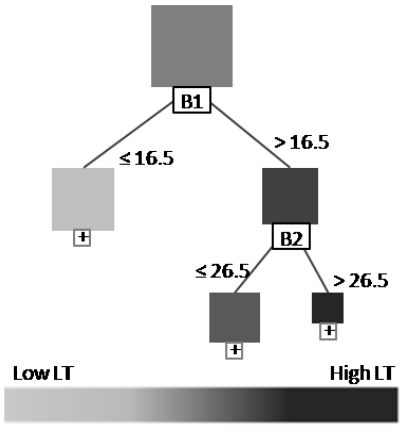

Figure 4: Decision Tree with LT as Output Variable

To get an acceptable TP in this production line the buffer capacity for B1 must be greater than 6. In order to receive as high TP as possible the capacity in this buffer should be greater than 12.
It is desirable to have as low LT as possible and Figure 4 reveals the settings in order to accomplish this. Notable is that the buffer capacity for buffer 1 is the main divider and if the capacity is less than 16 the lead time will be satisfying.
The capacity of the first buffer has the most impact on the throughput and the lead time in this study. The conclusion of this is that the station after the first buffer is the bottleneck in this production line. Letting the capacity for buffer 1 vary between 12 and 16 will result in low lead time and simultaneously provide the highest throughput.

Dispatching rules in a production line

The aim of this experiment is to understand how dispatching rules affect the outcome of a production line. The result of a multi-objective optimisation study is used to discover patterns in dispatching rule settings in order to maximise throughput (TP) and minimise all delayed products, i.e. the total tardiness (TT).
Input to the data mining experiment is output from a Discrete Event Simulation (DES) model. This simulation model is a representation of the H-factory at Volvo Cars in Skövde. The H-factory is committed to camshaft processing and 15 variants are handled on the production line. The H-factory consists of thirteen different groups of operations with one to seven machines in each group. All machines within a group of operations have the same capability and in front of every group of machines is a buffer.
When a product enters to a buffer it checks if a machine is free, if so the product is directly moved there. But, if there is not a machine available then the product is placed on a free spot in the buffer. The buffer is checked every time a machine has finished a product. If there is only one product there; move that one to the machine. The dispatching rules are considered each time there is more than one product in the buffer. The product to pick is dependent on the current dispatching rule assigned to that specific buffer.
There are eight different dispatching rules: shortest processing time (SPT), longest processing time (LPT), earliest due date (EDD), total working remaining (TWR), least work remaining (LWKR), most work remaining (MWKR), minimum slack time (MST) and operation due date (OPNDD).
There are 13 groups of operations and 8 dispatching rules. Due to the great number of different dispatching rule settings, simulation based optimisation (SBO) is used to generate an optimal configuration of the production line. The optimisation objective parameters are TP and TT. The output of the simulation based optimisation is the dispatching rules used for each operation with its resulting TP and TT. The number of different settings is 8^{13}, approximately $5.5 \cdot 10^{11}$. The optimisation generates about 400 solutions which are all non-dominated and on the Pareto front. These are used in the post processing step for investigation of better understanding of the solutions.

Results for the dispatching rules case

The data set with 13 input variables (applied dispatching rule for each buffer) and the output variables are TP and TT was used to generate decision trees. These are simple to interpret and they have therefore high usefulness.
It can easily be seen in Figure 5 that the bottlenecks, i.e. the most important variables, are op20 and op90. This is true for both TP and TT as output variables. In order to generate a small and more interpretable model all other input variables are excluded and new models are built.

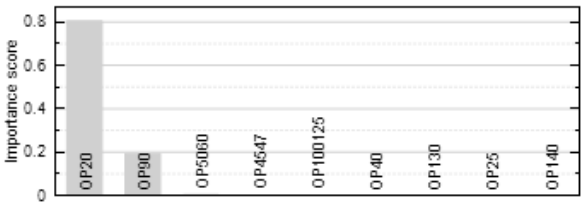

Figure 5: Important Variables in the H-factory.

The decision tree from which detailed information can be found is shown in Figure 6.

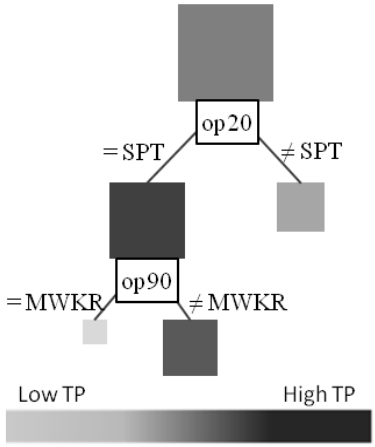

Figure 6: The Tree Structure for Maximising TP.

The information which can be extracted from the tree structure is that for a higher average TP use dispatching rule SPT in op20 and use any of the others but MWKR for the buffer before op90.
The experiment for the TT is performed in a similar way. As an initial study all input variables are used to identify the variables with most importance and the most important variables were identified to be

op20 and op90. Its tree structure can be seen in Figure 7.

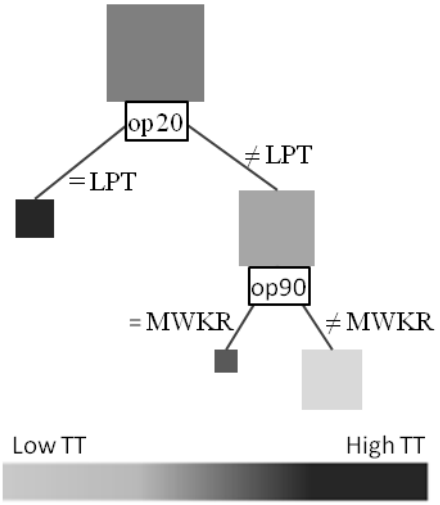

Figure 7: The Tree Structure for Minimising TT.

In contrast to the TP experiment the TT case focus on discovering the settings for a low prediction values. Letting the most severe bottleneck have a dispatching rule that is not LPT and the second one having any but MWKR will result in a low TT.
Combining the results will lead the decision maker to draw the conclusions that the dispatching rule for the most influencing buffer before op20 should be shortest processing time (SPT) and for op90 most work remaining (MWKR) should be chosen. The 11 other buffers are not significantly influencing the total tardiness or throughput.

SUMMARY AND FUTURE WORK

In this work, we have shown how information can be extracted from simulation data by means of data mining, providing support for a human operator aiming for optimising manufacturing processes. For example, the operator may learn how various process parameters affect different optimisation criteria.
One main question for future research concerns how to most effectively exploit the information that has been acquired by analysing simulation data, with other sources of information in actual decision situations. Another question for future work is to determine whether data mining can outperform various experimental design methods. While these methods, ranging from orthogonal arrays to stratified Latin Hypercube design, can be used to explore the input variables space uniformly and effectively for generating the required data sets, it is questioned whether these techniques are sufficient enough to unravel the relationships between input decision variables and the output parameters, which is an important purpose of many data mining processes.

ACKNOWLEDGEMENTS

This work was supported by the Information Fusion Research Program (www.infofusion.se) at the University of Skövde, Sweden, in partnership with the Swedish Knowledge Foundation under grant 2003/0104.

REFERENCES

Bauer E. and Kohavi R., 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants, *Machine Learning*, Vol. 36, Issue 1-2, 105-139

Breiman L., 1996. Bagging Predictors, *Machine Learning*, Vol. 24, Issue 2, 123-140

Breiman L., 2001. Random Forests, *Machine Learning*, Vol. 45, Issue 1, 5-32

Breiman L., Friedman J.H., Olshen R.A. and Stone C.J., 1984. *Classification and Regression Trees*, Wadsworth, Belmont

Chiba, K., Jeong S., and Yamamoto K., 2006. "Efficient Data Mining for Multi-Objective Design Exploration regarding Aerospace Vehicle", *ICNPAA-2006: Mathematical Problems in Engineering and Aerospace Sciences*

Fayyad, U. M., Piatetsky-Shapiro, G. and Smyth, P., 1996. From Data Mining to Knowledge Discovery in Databases, *AI Magazine*, Vol. 17, Issue 3, 37-54.

Hastie T., Tibshirani R. and Friedman J., 2001. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer-Verlag, London.

Jeong, S., Chiba, K., and Obayashi, S., 2005. "Data Mining for Aerodynamic Design Space," *Journal of Aerospace Computing, Information, and Communication*, Vol. 2, No. 11, 452–469.

Kusiak, A. 2006. Data Mining in Manufacturing: A Review, *Journal of Manufacturing Science and Engineering*, Vol. 128, Issue 4, 969-976.

Ng, A., Urenda, M., Svensson, J., Skoogh, A., and Johansson, B. 2007. "*FACTS Analyser: An innovative tool for factory conceptual design using simulation"*, In Proceedings of Swedish Production Symposium, Gothenburg, 28-30.

Provost F., Fawcett T. and Kohavi R., 1998. *The case against accuracy estimation for comparing induction algorithms*, Proc. Fifteenth Intl. Conf. Machine Learning, 445-553

Quinlan J.R., 1986. Induction of decision trees, *Machine Learning*, Vol. 1, Issue 1, 81-106

Quinlan J.R., 1993. *C4.5: Programs for Machine Learning*, Morgan Kauffman, San Francisco

Quinlan J.R., 1997. Data Mining Tools See5 and C5.0, http://www.rulequest.com/see5-info.html (accessed March. 28, 2009)

Witten I. and Frank E., 2005. Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann Publisher, San Francisco.