Computers & Operations Research 39 (2012) 424–436

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

A comparison of genetic programming and artificial neural networks in metamodeling of discrete-event simulation models

Birkan Can *, Cathal Heavey

Enterprise Research Centre, ERB, ER1030, University of Limerick, Limerick, Ireland

ARTICLE INFO

Available online 7 May 2011

Keywords:
Simulation metamodel
Genetic programming
Symbolic regression
Neural networks
Design of experiments
Decision support tool

ABSTRACT

Genetic programming (GP) and artificial neural networks (ANNs) can be used in the development of surrogate models of complex systems. The purpose of this paper is to provide a comparative analysis of GP and ANNs for metamodeling of discrete-event simulation (DES) models. Three stochastic industrial systems are empirically studied: an automated material handling system (AMHS) in semiconductor manufacturing, an (*s*,*S*) inventory model and a serial production line. The results of the study show that GP provides greater accuracy in validation tests, demonstrating a better generalization capability than ANN. However, GP when compared to ANN requires more computation in metamodel development. Even given this increased computational requirement, the results presented indicate that GP is very competitive in metamodeling of DES models.

1. Introduction

DES models are widely used in the design and analysis of systems. In many instances model execution times of DES models can be computationally expensive. In such cases, its use in operational tasks such as design, sensitivity analysis and optimization can be significantly undermined. This shortcoming of DES has motivated development of methods that allow the creation of approximate models, i.e., metamodels of systems which sacrifice accuracy for computational gain. A metamodel refers to an approximate predictive model of system performance which is dependent on decision variables.

Wang and Shan [1] outline the techniques that can be used to build approximate response models for engineering design problems. Jin et al. [2] perform a comparative analysis of different techniques: polynomial regression (PR), Kriging (KG), multivariate adaptive regression splines (MARS), radial basis functions (RBF) on a range of test problems. The results of the study showed a dispersion of the observed performance of the techniques in terms of accuracy and problem structures, e.g., size and nonlinearity of the problem. While these articles focus on only deterministic problems, Li et al. [3] present a thorough comparison on stochastic problems considering artificial neural networks (ANNs) and support vector regression (SVR) in addition to KG, MARS and RBF. However, none of these studies considered GP [4] in metamodeling of stochastic problems to evolve symbolic expressions.

In this paper, a comparison between GP and ANNs is presented since ANNs have found frequent application in simulation metamodeling. These approaches can develop models without underlying assumptions or *a priori* knowledge about the relationship between the control factors and the performance. To build approximate models, they only require the information on system configuration and the corresponding performance, i.e., training data. The approximations are progressively improved using the information available from previously generated metamodels throughout the process to stimulate the search to find better metamodels. This can be a means for building highly accurate metamodels as they do not necessitate a simplification on the complexity of the systems studied.

In the evaluation of GP against ANN, DES models of three systems, which are different in the size of decision space, in the degree of variability and in the range of the performance measures, are used. Since there is neither assumptions nor information on the underlying functions of the performance of these systems, the uniform design (UD) [5] is used to sample the decision space of the problems. UDs are space-filling experimental designs which can be used to obtain training data when the underlying model is unknown [6]. Therefore, they are inherently suitable to use with metamodeling approaches, such as ANNs and GP.

In the remainder of this paper, first a literature review of simulation metamodeling is presented. Following, the GP approach is briefly introduced. Subsequently, in Section 4, the methodology of the study is given and lastly the results of the

* Corresponding author.
E-mail address: birkan.can@ul.ie (B. Can).

0305-0548/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2011.05.004

study are presented. The results focus on the accuracy and robustness of the methods across the problems both in the model building (training) and in the validation (test) stages as well as the computational requirements. The results show that GP is a very competitive metamodeling method, showing superior results in developing more generalized metamodels when compared to ANN.

2. Literature review

One of the earliest reported approaches for building approximations is the response surface methodology (RSM) [7]. In general, low order polynomials are used in conjunction with regression analysis to fit the system responses [8–10]. In polynomial regression (PR), a low order analytical function of decision variables with unknown coefficients is used. The model is estimated via regression analysis. The model coefficients are updated with respect to residual errors between the fitted values by the regression model and the target values in training data. To drive the process an error function, such as root mean squared error (*rmse*), is used in assessing the relative performance of the models. Noguera and Watson [11] presented a simulation-based metamodeling study, where PR is applied to attain first- and second-order polynomial models of throughput of a chemical plant as a function of process variables such as demand rate, process rate, transfer rate and quality policy. Similarly, Durieux and Pierreval [12] used second-order polynomials to perform sensitivity analysis of a flexible manufacturing system to outline the effects of design parameters on average system utilization. The study compares cases where the number of decision variables increases from 1 to 8, and the results indicate a loss of accuracy in PR models with an increase in complexity. Although PR is easy to perform, assuming a simple form for the response function may not always be sufficient to capture nonlinear behavior of the system, which is its major drawback [2].

Alenezi et al. [13] used SVR to perform real time prediction of the order flow times in three manufacturing systems with different sizes: small, medium and large. The predictions from SVR with a linear kernel and ε-insensitive loss function are compared against traditional time series model and ANNs. The results indicate that SVR and ANNs gave close predictions in the vicinity of 13–22% *rmse* in predicting flow times in manufacturing systems. Nonetheless, both SVR and ANNs outperformed exponential smoothing and moving average models in all of the test problems. The literature presents many available kernel functions to use in SVR. However, SVR consists of a large family of parameters which influence its performance; hence, sophisticated approaches may be needed to obtain optimal settings [14]. Nonetheless, Guoqu and Dagui [15] presented an optimization study where a SVR metamodel is used to assist a genetic algorithm. They compared SVR against PR, KG and ANN in a two-bar structure design problem indicating better predictive performance of SVR.

Another method applied in simulation metamodeling is radial basis functions (RBFs). Hussain et al. [16] compare RBF against PR metamodels on various test functions. They considered full factorial and Latin hypercube sampling (LHS) for generating the training data. For the studied test functions, RBFs coupled with LHS outperformed PR with factorial designs. On the other hand, results with factorial designs generally exhibited better performance characteristics in terms of error and the distance[1] of the optimum of the metamodel from the actual optimum of the test function. Furthermore, as the number of samples increased, the performance of RBF models improved drastically compared to PR models.

Kriging (KG) is another approach which has found application among the recent metamodeling studies. Kleijnen [18] presents a review of the KG method applied to simulation metamodeling. Biles et al. [19] present a simulation-based optimization study of constrained (*s*,*S*) inventory systems using KG. While classical KG has been applied mainly to deterministic problems [18], Ankenman [20] presented a variation, stochastic Kriging, which can be used in metamodeling of stochastic simulation models. KG models have their merits, including an error bound estimation of the prediction which can be useful when the training data is noisy. However, KG exploits the surrounding observations during metamodeling. Therefore, the method renders the estimation local to the limit of the range of training data. This can undermine KG's ability to extrapolate the unseen data, potentially leading to unrealistic estimations in problems with irregular, non-stationary response behavior [21]. On the other hand, metamodels of industrial systems are often expected to generalize outside the training range for usability.

A predominant approach to metamodeling of simulation models is ANNs due to their efficiency [22–26]. In an earlier study, Kilmer et al. [27] studied ANNs on an inventory control problem. Charmbers and Mount-Campbell [28] approximated throughput rate and job flow times of a manufacturing process using ANNs. They developed metamodels of these performance measures dependent on buffer levels. The ANNs were trained with 2300 data points collected from a queuing network simulation. Their results indicate approximations with 6% maximal error. Yildiz and Eski [29] developed simulation metamodels of assembly lines in order to identify optimal design and operational parameters. The study reported improvement levels reaching up to 20% with the improved efficiency in search through the use of ANNs. Similarly, Altiparmak et al. [25] derive throughput rates of assembly lines based on individual buffer levels at work stations in a comparative study between ANNs and PR models. Recently, Yang [30] applied a single layer neural network to derive cycle time (CT)—throughput (TH) curves of manufacturing systems. They first identify the ANN with the least number of hidden nodes sufficient to perform metamodeling by using singular value decomposition [31]. They sequentially collect the design samples by using the best network models. Their results on a semiconductor manufacturing system indicate estimates deviating 5–10% compared to the actual measurements. Despite the efficiency of ANNs, their usability can be prohibitive in practice. ANNs require prudent selection of configuration parameters to protect against the phenomenon of overfitting [32], where the complexity of the model inhibits its performance on unseen data, i.e., generalization capacity. This basically refers to the case where the training error of a model is low; however, high in validation step. Furthermore, the resulting network models can be difficult to interpret meaningful for practitioners.

So far, a review on common approaches applied in metamodeling of DES models industrial systems is provided identifying advantages and shortcomings. In this paper, we compare these shortcomings via symbolic regression (SR) with GP. The use of GP for generating symbolic representations of training data was first proposed by Koza [4]. Through the use of effective convergence algorithms, GP can produce compact expressions which have excellent generalization properties [33]. Such representations are often mathematical expressions, i.e., functions, unlike metamodels generated by ANNs. These explicit functions can provide an immediate alternative platform to DES models in practice when the trade-off between accuracy and the computational requirements is appropriate. As the application of GP to

[1] See Keys [17] for the use of this metric to assess metamodel performance.