Throughput estimation of small flow lines using hybrid evolutionary techniques

Konstantinos S. BOULAS, Georgios D. DOUNIAS
Department of Financial and Management Engineering, School of Engineering, University of the Aegean, Chios Island, Greece , City, State, Country, kboulas@aegean.gr,g.dounias@aegean.gr

Chrissoleon T. PAPADOPOULOS
Master of Engineering Management, School of Engineering, Nazarbayev University,Astana, Kazakhstan
chrysoleon.papadopoulos@nu.edu.kz

This work develops a method for obtaining analytical formulas for estimating the throughput of short reliable approximately balanced production lines with exponential service times, without intermediate buffers. Machine processing times are very close to one another, but their values may vary within an interval. The method used to obtain the throughput is based on a hybrid intelligent scheme combining two known evolutionary techniques, (a) genetic programming, which is an evolutionary computation paradigm for the automatic induction of syntactic expressions, and (b) genetic algorithms, an approach which is used for searching the space of possible program syntaxes, combined with a symbolic regression task. The complete algorithmic scheme and its variants aim to construct formulas that express the line throughput as a ratio of two polynomials. Similar formulas have been previously presented in literature in a theoretical work where exact formulas for two and three stations were given via analytical calculation. In this work, the training set for the intelligent algorithmic scheme used is produced through Markovian analysis of the system, in order to obtain high-quality accurate data for generalization. The obtained formulas for three and four station lines are fairly accurate and seem to generalize well. The proposed method is believed that can be used to produce approximate (or even accurate) throughput formulas in polynomial form for longer production lines.

Key words: Throughput; Serial production lines; Symbolic regression; Genetic Programming; Hybrid intelligent systems

1. Introduction

Since the mid 50's numerous methods and techniques have been elaborated for solving the problem of throughput estimation of a serial production line. All these methods have been applied in special cases of the general problem, which is very complicated due to the combinatorial explosion. Due to the complexity of the problem, it is generally accepted that there is a lack of analytical formulas for direct estimation of throughput. Related research articles have been published by Hunt (1956), Muth (1984) and Blumenfeld (1990) among others that deal with the expression of throughput with the use of a mathematical formula (approximate or exact).

The complexity of a system increases with the number of workstations in the production line, the capacity of the intermediate buffers and the variability of processing times, among other factors. Methods based on decomposition proposed by Gershwin (1987) are used to deal with this complexity. Based on data obtained by decomposition technique in order to train an algorithm of Genetic Programming (GP) Papadopoulos et al. (2002) extract some formulas elaborating the idea of using intelligent techniques from the domain of Artificial Intelligent to deal with that kind of problems, see also Boulas et al. (2017) for similar attempts.

GP as first introduced by Koza (1992) is an automated method for creating a solution, suboptimal in most of the cases, but good enough to be accepted if the algorithm has the proper preparation and modulation. It is necessary to state the problem at a high level and let the population based algorithm generate the solution. GP is the extension of Genetic Algorithms (GA) (Holland (1992)) and for this reason, shares many common elements and features with them.

In the present work, a GP scheme is gradually transformed into a GA, conserving all its characteristics. We aim to accelerate the evolution procedure of extraction approximate formulas for throughput estimation of a short serial production line that consists of a small number of stages with no intermediate buffers. In this work we use the model of a production line handling discrete worked pieces, with asynchronous movement of parts among workstations, see Papadopoulos et al. (2009) for details of the model.

2. Solution Methodology

We aim to create an approximate formula for the throughput using symbolic regression techniques and ultimately understand the accurate solution structure. In order to train the algorithms, we use data obtained by the MARKOV algorithm given in Papadopoulos et al. (2009) for K stages, $K = 2, 3, 4$ with no intermediate buffers, one machine per workstation and exponential mean service rates in the interval $\mu_i \in [0.9, 1.1]$, where $i \in \{1, 2, \ldots, K\}$. We create the necessary matrix of N fitness cases, $NX(K+1)$, using the μ_i of each machine and the corresponding throughput X_K obtained by the MARKOV algorithm. We use that matrix for the training of the proposed algorithm aiming at expressing the throughput as a ratio of two polynomials obtained directly from GP. That means, the terminals of GP should be monomials in the form $m_i = c_i \mu_1^{r_{i1}} \mu_2^{r_{i2}} \cdots \mu_K^{r_{iK}}$ where m_i is the monomial i in the terminal set T, $c_i, r_{i1}, r_{i2}, \ldots, r_{iK}$ are integers in an interval, which is parameter of the algorithm, and represent the ith monomial coefficient c_i where $c \in \{n, d\}$ of formula (1), i.e., take value n if the monomial stands in numerator and d if the monomial stands in denominator. The exponents r_{ij} with $j \in \{1, 2, \ldots, K\}$ are for each mean service rate, respectively, and the $r \in \{p, e\}$ of formula (1), i.e., take value p if the monomial stands in numerator and e if the monomial stands in denominator. Finally, the $\mu_1, \mu_2, \ldots, \mu_K$ are the mean service rates of the workstations of the production line, and they are arguments in the MARKOV algorithm and train the algorithm through the fitness cases matrix. By constructing the trees of GP, we have the following two observations:

1. The root of the tree is always the division operator and the other internal nodes are always addition operators. Thus, the function set F contains only the two symbols $F = \{+, /\}$.

2. There is complexity in the case of raise power operations that causes long runs. Since the T has a finite number of elements, we save execution time storing all results in computer memory using a matrix $M \times N$ where M is the number of monomials in the T. The matrix computation takes place once during the initialization stage, monomials values for each fitness case stored in memory where is accessed from the algorithm every time the specific monomial exists in a solution chromosome.

The above two observations combined with the fact that by executing the algorithm we try to construct an approximate solution for calculating throughput which means that we investigate various combinations of numerator and denominator, enables us to replace the trees of GP with respective chromosomes of a GA. The evaluation of each chromosome of the GA gives the same result as the corresponding tree of GP. That means there is a one-to-one correspondence between the GP trees to the chromosomes of the GA and one of them is known then the other can be obtained too. This

exact match can be used in future experiments in more complicated schemes like ensembles. The final output of the algorithm is the best solution at the time that the termination criterion of the algorithm is activated and is a ratio of two polynomials in the form of formula (1).

$$X_K = \frac{n_1(\mu_1^{p_{11}}\mu_2^{p_{12}}\cdots\mu_K^{p_{1K}}) + n_2(\mu_1^{p_{21}}\mu_2^{p_{22}}\cdots\mu_K^{p_{2K}}) + \cdots}{d_1(\mu_1^{e_{11}}\mu_2^{e_{12}}\cdots\mu_K^{e_{1K}}) + d_2(\mu_1^{e_{21}}\mu_2^{e_{22}}\cdots\mu_K^{e_{2K}}) + \cdots} \quad (1)$$

3. Conclusions

The results so far show that the algorithm, encoded in C++, approximates MARKOV's results with an average error in the order 10^{-4} and it can manage solution populations consisting of 10^6 individuals with an average size of a chromosome consisting of about 60 genes in a reasonable time. The proposed approach generalizes well in out of sampling data as well as in out of the training area experimentation. Nevertheless, some problems arise that have to do with the vast search space, with the trade-off between speed and memory and with the convergence in local minima from a point in the evolutionary procedure and after that. Experimentation continues, for performance improvement as well as for understanding the structure of the exact solution,if that is possible, using the GP and GA paradigm .

Acknowledgments

The authors thank the Municipality of Chios for granting sabbatical permission to the principal researcher of this work K. Boulas, (ΑΔΑ: 783ΧΩΗΝ-ΛΤΤ).

References

Blumenfeld, Dennis E. 1990. A simple formula for estimating throughput of serial production lines with variable processing times and limited buffer capacity. *International Journal of Production Research* **28**(6) 1163–1182.

Boulas, Konstantinos, Georgios Dounias, Chrissoleon Papadopoulos. 2017. Approximating Throughput of Small Production Lines Using Genetic Programming. Evangelos Grigoroudis, Michael Doumpos, eds., *Operational Research in Business and Economics: 4th International Symposium and 26th National Conference on Operational Research, Chania, Greece, June 2015*. Springer International Publishing, 185–204.

Gershwin, Stanley B. 1987. An Efficient Decomposition Method for the Approximate Evaluation of Tandem Queues with Finite Storage Space and Blocking. *Operations Research* **35**(2) 291–305.

Holland, John Henry. 1992. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

Hunt, Gordon C. 1956. Sequential Arrays of Waiting Lines. *Operations Research* **4**(6) 674–683.

Koza, John R. 1992. *Genetic programming: on the programming of computers by means of natural selection*. MIT press, Cambridge, Massachusetts London, England.

Muth, Eginhard J. 1984. Stochastic processes and their network representations associated with a production line queuing model. *European Journal of Operational Research* **15**(1) 63–83.

Papadopoulos, Chrissoleon, Athanasios Tsakonas, George Dounias. 2002. Combined Use Of Genetic Programming And Decomposition Techniques For The Induction Of Generalized Approximate Throughput Formulas In Short Exponential Production Lines With Buffers. *Proceedings of the 30th International Conference on Computers & Industrial Engineering*. University of the Aegean.

Papadopoulos, Chrissoleon T., Michael J. Vidalis, Michael E. J. O'Kelly, Diomidis Spinellis. 2009. *Analysis and Design of Discrete Part Production Lines*. Springer New York.