On Throughput Approximation of Resource Allocation Systems by Bottleneck Regrowing

Ricardo J. Rodríguez[1] and Javier Campos[2]

[1] Centro Universitario de la Defensa, Carr. de Huesca, s/n 50090, Zaragoza, Spain
Università degli Studi della Campania “Luigi Vanvitelli”, Italy
`rjrodriguez@unizar.es`
[2] Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza, María de Luna 1, 50018 Zaragoza, Spain
`jcampos@unizar.es`

1 Extended Abstract

Complex systems such as manufacturing, logistics, or web services, are commonly modeled as Discrete Event Systems dealing with the resource-allocation problem, also called Resource Allocation Systems (RAS). A RAS is a DES in which a set of concurrent processes coexist, which must compete in order to allocate some shared resources [3]. In particular, Petri nets are a widely used formalism to model these systems. Although their functional properties have been extensively studied in the literature, their non-functional properties (such as throughput) have usually been ignored. In this paper, we focus on a Petri net subclass useful for modeling concurrent sequential processes with shared resources, termed as S^4PR nets.

In these nets, workpieces undergo successive transformations, which may also have independent processing steps, until reaching their final state. A production plan is represented by means of a strongly connected state machine (with no internal cycles), in which availability of different routings in the system may be restricted to the use of non-consumable, reusable resources.

Figure 1 depicts a particular example of S^4PR. It represents three sequential processes without any routing decisions and three resources: the initial marking of p_8 and p_9 is the number of idle resources of each type shared by the left-hand side and the central process, while p_{19} represents the resource shared by the central and the right-hand side process.

In this paper, we propose an iterative strategy to approximate the throughput of a RAS, modeled by S^4PR nets. The strategy makes use of mathematical programming problems for which polynomial complexity algorithms exist – thus offering a good trade-off between accuracy and computational complexity – and of the exact solution of underlying Continuous Time Markov Chain (CTMC) models of subsystems much smaller than the whole original model.

Based on previous works regarding the computation of upper throughput bounds [1, 2] and similar to ideas already introduced to calculate upper throughput bounds in certain subtypes of Petri nets such as Marked Graphs (MGs) [5] and process Petri nets (PPNs) [6], our strategy works as follows. For a given

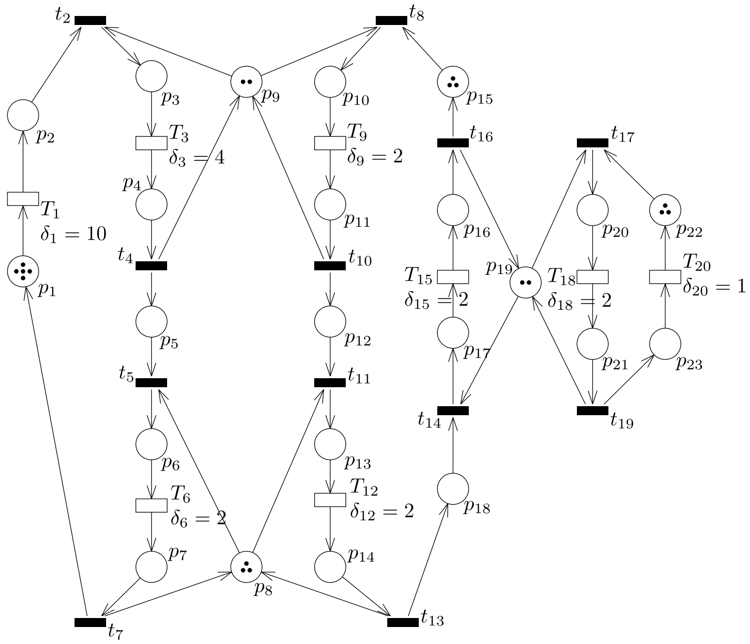

Fig. 1: A particular example of a S^4PR net.

S^4PR net and tolerance, our iterative strategy computes as a first step the slowest P-semiflow of the system (i.e., the *bottleneck*), by computing the upper throughput bound of a subset of transitions. Then, in each iteration step the next P-semiflow most likely to be constraining the current bottleneck is computed. This P-semiflow is taken as the new bottleneck of the system, and the exact throughput of the subnet generated by the new and the previous bottleneck is calculated by solving the underlying CTMC. When tolerance is not achieved, another iteration step will be performed. Note that in each iteration step, the bottleneck of the system is regrown. Note also that the throughput of the transition in the subnet is lower or higher than its real throughput, considering the full system. Hence, the values of throughput computed in each iteration step approximate to the real throughput.

The iterative strategy to approximate throughput values in S^4PR nets is based on first computing the slowest P-semiflow which is then iteratively regrown considering the next P-semiflow most likely to be constraining the system. Although we have shown that our iterative approximation technique converges, we cannot characterize its convergence speed nor its accuracy. Thus, we have evaluated our strategy in a set of randomly generated S^4PR nets to give an insight into its usefulness.

To evaluate the effectiveness of our strategy, we developed a tool[3] to randomly create S^4PR nets taking into account various parameters such as the production plan size, transition rates, number of transitions, resources and available resource copies, and resource-sharing between production plans. Furthermore, our iterative strategy was implemented as a plug-in of PeabraiN [4] (using GLPK v4.55 as an LP solver),

From the extensive experiments performed, we concluded that: (i) the throughput obtained with our approach approximates better to the real throughput than (classical) upper throughput bounds, reaching an improvement on average close to 20%; and (ii) small portions of the net are representative enough to approximate to the real throughput of a transition in big S^4PR nets. In particular, the experiments showed that roughly 20% of the original net size is enough. Roughly speaking, our results indicate that the bigger the S^4PR net is, the smaller the net proportion which is sufficiently representative.

The full version of this paper was published in [7].

References

1. Bernardi, S., Campos, J.: A Min-Max Problem for the Computation of the Cycle Time Lower Bound in Interval-Based Time Petri Nets. IEEE T Syst Man Cy-S 43(5), 1167–1181 (2013)
2. Campos, J., Silva, M.: Structural Techniques and Performance Bounds of Stochastic Petri Net Models. In: Rozenberg, G. (ed.) Advances in Petri Nets 1992. LNCS, vol. 609, pp. 352–391. Springer (1992)
3. Colom, J.: The Resource Allocation Problem in Flexible Manufacturing Systems. In: van der Aalst, W., Best, E. (eds.) Applications and Theory of Petri Nets, LNCS, vol. 2679, pp. 23–35. Springer (2003)
4. Rodríguez, R.J.: A Petri Net Tool for Software Performance Estimation Based on Upper Throughput Bounds. Automat Softw Eng 24(1), 73–99 (Jan 2017)
5. Rodríguez, R.J., Júlvez, J.: Accurate Performance Estimation for Stochastic Marked Graphs by Bottleneck Regrowing. In: Proc. 7th European Performance Engineering Workshop. LNCS, vol. 6342, pp. 175–190. Springer (Sep 2010)
6. Rodríguez, R.J., Júlvez, J., Merseguer, J.: On the Performance Estimation and Resource Optimisation in Process Petri Nets. IEEE T Syst Man Cy-S 43(6), 1385–1398 (2013)
7. Rodríguez, R.J., Campos, J.: On Throughput Approximation of Resource Allocation Systems by Bottleneck Regrowing. IEEE Transactions on Control Systems Technology (2017), `http://webdiis.unizar.es/~ricardo/files/papers/RC-TCST-17.pdf`, accepted for publication. To appear.

---

[3] Released under GPLv3 license and freely available at `http://webdiis.unizar.es/~ricardo/?page_id=527`.