MODELING AND PERFORMANCE OPTIMIZATION OF A DISTRIBUTED PRODUCTION SYSTEM

O. Zaikin*, A. Dolgui, and P. Korytkowski***

* *Technical University of Szczecin, Institute of Computer Science and Information Systems, Zolnierska 49, 71-210, Szczecin, Poland*
ozaikine@wi.ps.pl

** *University of Technology of Troyes, 12 rue Marie Curie 10010, Troyes Cedex, France*
dolgui@utt.fr

Abstract: An approach, based on queuing modeling and simulation, to resource allocation for a distributed production system is examined in the paper. The problem is formulated as an objective function minimization for a queuing network. The optimization is based on the combination of analytical modeling and simulation. The analytical model is used for primary evaluation and branching for the optimization algorithm. The simulation is used for validation of analytical results and algorithm stop decision.

Keywords: Production systems, Queuing network models, Resource allocation, Simulation, Optimization.

1. INTRODUCTION

Sharing of resources and hence waiting in queues is a common phenomenon for logistics, distributed production, corporate and telecommunication networks, etc. In spite of different physical contents, research of these systems requires queuing approach and specific optimization methods.

A rich background of queuing modeling exist. Several aspects of queuing analysis are presented in (Buzacott and Shanthikumar, 1993; Chee Hock, 1997; Hall, 1991). Techniques for optimization using simulation are studied, for example, in (Azadivar and Lee, 1988; Biethan and Nissen, 1994; Gordon and Newell, 1967; Guariso *et al.*, 1996).

In this paper, the queuing approach is considered to resource allocation and performance optimization for a distributed printing company. The problem consists of optimal allocation of production resources to different factories of the company (Dolgui and Zaikin, 2000; Zaikin *et al.*, 2000). An algorithm is proposed which is based on queuing modeling and on the use of queuing models analysis, simulation and optimization techniques.

2. PROBLEM STATEMENT

A big printing company from Poland including several regional (local) factories and a central one is studied (see Fig.1). Offset printing machines equip the local factories and more expensive digital printing machines equip the central one. The local factories answer to local customer demands.

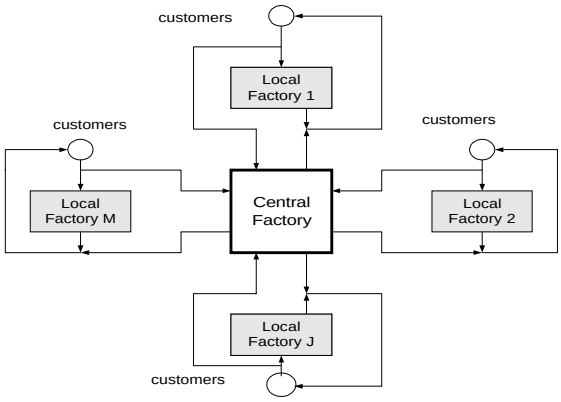

Fig. 1. Distributed printing company.

However when some of them is overcharged, their demands are transferred to the central factory. It is obviously, that the central digital printing is more expensive than local offset one. On the other hand the lead time of central factory is considerably smaller than lead time of local factory.

The rates of customer's demand for each local area (factory) and the processing time for all equipment are known. It is necessary to define the number of offset machines for each local factory and the number of digital machines for the central factory minimizing the total cost of printing and taking into account the customer's demand and the required lead times (delays).

In terms of queuing modeling, the studied system can be presented as an open queuing network given by an oriented graph $G=\{S,F\}$. Vertices of the graph $\{S\}$ are processing nodes and arcs $\{F\}$ are flow processes between them.

For the studied case, the following queuing network with star configuration is examined. There are a set of local processing nodes (PN), each PN serves the corresponding local area, and one center node (CN). Each local PN consists of a number of private servers, operating in parallel (offset machines).

If all the servers of a corresponding local PN are occupied, then the incoming jobs go to the CN with limited number of common (leased) servers (digital machines). If all leased servers of CN are busy, then the incoming job goes to the waiting queue of the corresponding PN. The jobs in PN and CN are served in FCFS discipline. There are no restrictions on the number of jobs waiting in the queue.

It is necessary to define the number of private servers in each local PN and leased servers in the CN, taking into account the difference of productivity and the difference of cost of the idle time for local and leased servers.

So, the formal model of resources allocation can be represented as follow:

Input data

$S=\{s\}$ is a set of PN,

$F=\{f_S\}$ is a set of flow processes, incoming in PN,

$\Pi(f)=\{\Psi_f(n,t),\lambda_f\}$ is the set of parameters of FP '*f*', where $\Psi_f(n,t)$ is the distribution law of arrivals of customer demands, λ_f is the rate of arrivals,

$\tilde{\tau}_S, \tilde{\tau}_C$ is the average processing time of a job in local PN and CN (including delivery time), respectively,

γ_S, γ_C are the costs of a server (acquisition, exploitation, amortization) reduced to a time unit for private and leased server, respectively.

Control parameters

$N_S, s \in S$ is a number of private servers assigned for each local PN $s \in S$,

N_C is a number of leased servers assigned to the CN.

Objective function

The objective function has the three following components:

1. The total processing time of all flow processes for the time interval T_o,

$$T_\Sigma = \sum_f \lambda_f \tilde{\tau}_S T_0 \qquad (1)$$

2. Total costs of equipment reduced to the time interval T_o,

$$C_\Sigma = (N_C \cdot \gamma_C + \sum_S N_S \cdot \gamma_S) \cdot T_0 \qquad (2)$$

3. Total costs of server's utilization during the time interval T_o,

$$U_\Sigma = \left(\frac{\tilde{\lambda}_C}{N_C \mu_C} \tilde{\delta}_C + \sum_S \frac{\tilde{\lambda}_S}{N_S \mu_S} \tilde{\delta}_S \right) T_0 \qquad (3)$$

where

$f \in F$ is the flow process index,

$s \in S$ is the PN index,

λ_f is the rate of the arrival of jobs for the flow process '*f*',

$\tilde{\tau}_S$ is the average processing time of a job in PN '*s*',

μ_S is the rate of processing of jobs in PN '*s*',

N_S is the number of private servers in PN *s*,

N_C is the number of leased servers in CN,

$\tilde{\lambda}_S, \tilde{\lambda}_C$ are the effective arrival rates to private and leased server, respectively,

γ_S, γ_C are the costs of private and leased server by time unit,

$\tilde{\delta}_S, \tilde{\delta}_C$ are the costs of idle time for private and leased servers (by time unit),

T_o is a time interval for optimization.

The first component T_Σ shows the total quality of processing, the second C_Σ and third U_Σ components give the total costs of installation and processing, respectively. These objective function components depend on the control parameters N_S and N_c in different ways. Nevertheless, it is possible to define such values of control parameters, which provide a minimum of the objective function:

$$CR = \alpha T_\Sigma + \beta U_\Sigma + C_\Sigma = min, \qquad (4)$$

here α and β are weighting coefficients.

3. SOLUTION METHOD

3.1 Analytical queuing model.

In general case, the formulated problem has no analytical solution. However, an evaluation for some simplified cases can be obtained based on the theory of queuing systems (Hall, 1991).

In Fig. 2, a structure of multi-server queuing system with an input flow f and two kinds of servers, private N_S and leased N_C, is represented. An incoming job goes to private server for processing. If all private servers are busy, the job goes to leased servers. In the case when the all servers are busy, incoming job goes to the waiting queue and at fixed time interval τ_r returns for processing. Several repetitions of waiting time for the same job are admitted.

Here, the following notation is used:

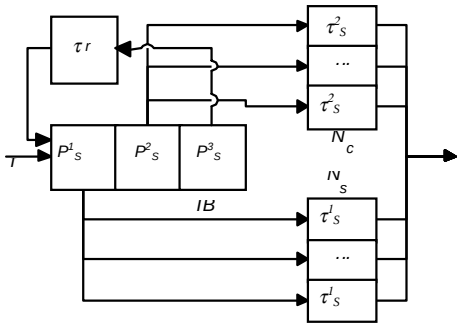

Fig. 2. A particular case of the studied problem.

P_S^1 is the probability of processing by a private server for an incoming job (there is an idle private server),

P_S^2 is the probability of processing by a leased server for an incoming job (all private servers are busy and there is an idle leased server),

P_S^3 is the probability of going to waiting queue for an incoming job (all the servers are busy),

τ_S^1, τ_S^2 are the average processing time by private and leased server, respectively,

$\tau_S^3 = \tau_r$ is the fixed waiting (repeating) time.

Let's $\tau_S^1 \geq \tau_S^2$ and $\tau_S^3 \geq \tau_S^1, \tau_S^2$.

It's obviously from condition of normalization, that

$$P_S^1 + P_S^2 + P_S^3 = 1.$$

Therefore, the average processing time $\tilde{\tau}_S$ of one job in PN *s* can be defined as follows:

$$\tilde{\tau}_S = (\tau_S^1 P_S^1 + \tau_S^2 P_S^2 + \tau_S^3 P_S^3) \frac{1}{1 - P_S^3} \qquad (5)$$

Probabilities P_S^1, P_S^2, P_S^3 can be found as follows.

The queuing system with m parallel servers and without input buffer can be deemed as Erlang loss system {M/M/m/m} according to Kendall notation. The probability of blocking P_b for this kind of queuing system is defined from the following expression:

$$P_b = \frac{\frac{\rho^m}{m!}}{\sum_{k=0}^{m} \frac{\rho^k}{k!}} = P_0 \frac{\rho^m}{m!} \qquad (6)$$

where

$P_0 = \left[\sum_{k=0}^{m} \frac{\rho^k}{k!} \right]^{-1}$ is the probability of the idle state of the queuing system,

$\rho_C = \lambda / \mu_C$ and $\rho_S = \lambda / \mu_S$ are traffic intensities.

Using expression (5), the probabilities P_S^1, P_S^2, P_S^3 can be obtained from the following formulas:

$$P_S^1 = 1 - P_m = 1 - \frac{\frac{\rho_S^{N_S}}{N_S!}}{\sum_{k=0}^{N_S} \frac{\rho_S^k}{k!}}, \qquad (7')$$

$$P_S^2 = (1 - P_S^1)(1 - \frac{\frac{\rho_C{}^{N_C}}{N_C!}}{\sum_{k=0}^{N_C} \frac{\rho_C{}^k}{k!}}), \qquad (7'')$$

$$P_S^3 = 1 - P_S^2 - P_S^1 . \qquad (7''')$$

By analogy, the effective rates $\tilde{\lambda}_S$ and $\tilde{\lambda}_C$ of jobs arrival in PN *s* and in CN are:

$$\tilde{\lambda}_S = P_S^1 \lambda_S \text{, and } \tilde{\lambda}_C = \sum_S P_S^2 \lambda_S , \qquad (8)$$

The values $\tilde{\tau}_S, \tilde{\lambda}_S, \tilde{\lambda}_C$ from expressions (5)-(8) can be used in (1)-(4) for the objective function evaluation.

3.2 Optimization approach.

The following branch and bound algorithm is proposed. The algorithm is based on sequential use of analytical calculation and simulation. The analytical model (see sub-section 3.1) is used for primary evaluation and choice of optimal direction of branching. It gives a Lower Bound for objective function. The simulation is used for validation of analytical results, for Upper Bound calculation and for stop condition evaluation of the algorithm.

The algorithm is composed of the three following components: Solution Tree, Lower Bounds, and Branching Algorithm.

A. *Solution Tree* (see Fig. 3)
The number of compared variants depends on:
a) number of PNs,
b) total number of servers (machines),
c) desired solution accuracy,
d) method for lower bound calculation.

B. *Lower Bound.*
Let $\dot{N}^k = (n_C^k, n_{S1}^k, ...n_{SI}^k)$ be the *k-th* step of the algorithm, and

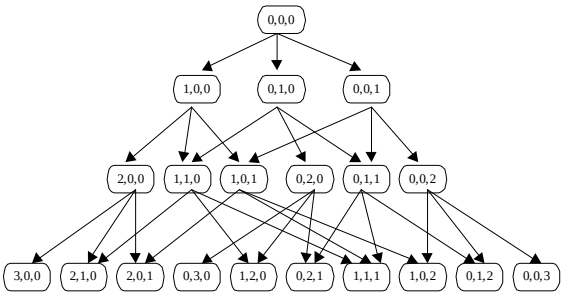

Fig. 3. Solution tree for 3 machines and 3 PNs.

$$\dot{N}_1{}^{k+1} = (n_C^{k+1} - 1, n_{S1}^{k+1} + 1, ...n_{SI}^k), \ldots,$$

$$\dot{N}_I{}^{k+1} = (n_C^{k+1} - 1, n_{S1}^{k+1}, ...n_{SI}^k + 1)$$

be I variants for the next *(k+1)-th* step.

Lower Bound for the variant $i \in I$ is calculated by using the model from Fig. 2 (sub-section 3.1) separately for each PN. For each PN, the structures of queuing system with $(n_c^{k+1} - 1)$ leased servers and n_S^{k+1} or $(n_S^{k+1} + 1)$ private servers are examined. The objective function value is calculated under the assumption of independence between PNs.

C. Branching Algorithm.
Branching is started from the initial capacity allocation where all the resources are assigned to CN. The branching algorithm tries to reallocate a part of the resources from CN to local PNs:
a) choice of the resources to reallocate from CN to local PN,
b) analytical calculation for each local PN of the criterion value using the model from sub-section 3.1 under condition that the resources are allocated to this PN,
c) Lower Bound estimation for the current solution,
d) if Lower Bound < Upper Bound, then continue this branch, otherwise return to the previous branch,
e) choice of PN (branch) for which the criterion value is minimal, reallocate the resource from CN to the PN,
f) use of the Simulation model for criterion value calculation for the obtained solution (Upper Bound evaluation).

4. NUMERICAL EXAMPLES

The objectives of the numerical examples are:
a) verify the dependence of objective function on control parameters;
b) test the convergence of the proposed branch and bound algorithm.

4.1 Validation of the analytical model.

The first objective of simulation consists on validation of proposed in sub-section 3.1 analytical model.

To develop the simulation model the system ARENA has been used (Kelton *et al.*, 1997). Simulation experiments were conducted for the following conditions:
1. Queuing model consist on arbitrary number of PNs and one CN.
2. The simulation time is *1000 tu*. Provided simulation experiments shown that Warm-up time is 200 tu.
3. The simulation model is realized for:

a) Poisson law of jobs arrival and exponential distribution of processing times;
b) Poisson law of jobs arrival and deterministic distribution of processing times;
c) Poisson law of jobs arrival and the Erlang law of the 2nd degree for processing time distributions.

For example, for two PNs, the results of simulation are given in Table 1. Simulation tests have shown the following results:

1. Conducted simulation experiments have shown that the results obtained by simulation are the same or very closed to the results calculated with analytical model. The degree of accuracy depends on the traffic intensity (ρ). Smaller the traffic is, closer the results are.
2. Different distributions of processing time do not affect the path of the optimal solution searching. Simulation experiments have shown that presented analytical model for network with exponential servers is also valid for a network where servers are deterministic or with Erlang distribution. Comparing the results for network with different kind of servers (different distribution of processing times), it follows that the results differ no more than 3%.

4.2 Tests of the convergence.

First example, a network with one central and two local PNs has been examined. The input data are the following:

a) input flows laws for local PNs are Poisson ones with arrival rates $\lambda_{S1} = 20$, and $\lambda_{S2} = 10$,

Table 1 Analytical and simulation results for model with various numbers of servers, one central node and two local nodes

NC	PN_1	PN_2	Anal.	Simulation			ρ
				Exp.	Const	Erlang	
5 servers							
2	2	1	16.286	**17.144**	16.715	16.890	0.4839
1	3	1	**15.335**	17.174	**16.272**	**16.721**	0.536
6 servers							
1	4	1	**16.135**	16.975	**16.390**	16.677	0.4546
1	3	2	16.298	**16.812**	16.390	**16.613**	0.455
7 servers							
1	5	1	**17.080**	17.661	17.221	17.434	0.395
1	4	2	17.097	**17.262**	**17.081**	**17.186**	0.395
8 servers							
1	5	2	**18.042**	**18.125**	**18.019**	**18.055**	0.349

b) leased and private servers have exponential processing times with $\mu_{S1} = \mu_{S2} = 10$, $\mu_C = 16$ and $\tilde{\tau}_{S1} = \tilde{\tau}_{S2} = 0.1$, $\tilde{\tau}_C = 0.016$,
c) repeating time $\tau_R = 0.5$,
d) weighting coefficient $\alpha = 1$ and $\beta = 2$.

In Fig. 4 and in Table 2, the searching path in the solution tree is presented.

Second example, a network with one central and five local PNs is examined. The input data are the following:

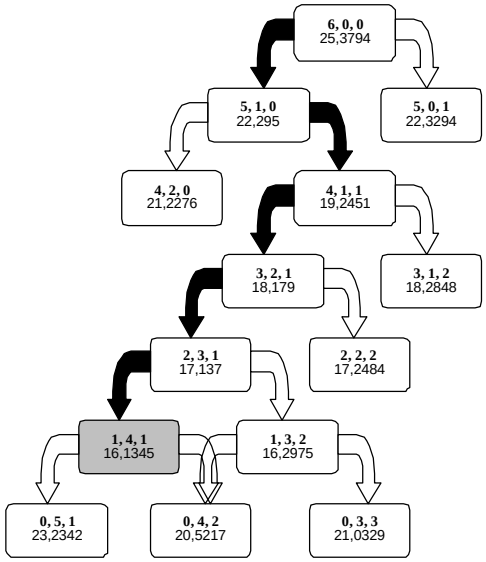

Fig. 4. Searching path.

Table 2 Results for model with two PNs and 6 servers

NC	PN_1	PN_2	Cost	Time	Utilization	Total
6	0	0	12	1.88191	5.74872	25.3794
5	1	0	11	2.13025	4.58239	22.295
5	0	1	11	2.09477	4.61733	22.3294
4	2	0	10	2.32907	4.44928	21.2276
4	1	1	10	2.34257	3.45129	19.2451
3	2	1	9	2.54327	3.31784	18.179
3	1	2	9	2.60302	3.34087	18.2848
2	3	1	8	2.69539	3.2208	17.137
2	2	2	8	2.85652	3.19595	17.2484
1	4	1	7	2.83065	3.15192	**16.1345**
1	3	2	7	3.16999	3.06373	16.2975
0	5	1	6	8.38095	4.42661	23.2342
0	4	2	6	5.30263	4.60952	20.5217
0	4	2	6	5.30263	4.60952	20.5217
0	3	3	6	6	4.51645	21.0329

a) input flows laws for each local PNs are Poisson ones with arrival rates $\lambda_{S1} = 20$, $\lambda_{S2} = 10$, $\lambda_{S3} = 30$, $\lambda_{S4} = 40$ and $\lambda_{S5} = 25$;

b) leased and private servers are exponential with $\mu_{S1} = \mu_{S2} = \mu_{S3} = \mu_{S4} = \mu_{S5} = 10$, $\mu_C = 16$ and average processing times $\tilde{\tau}_{S1} = \tilde{\tau}_{S2} = \tilde{\tau}_{S3} = \tilde{\tau}_{S4} = \tilde{\tau}_{S5} = 0.1$, $\tilde{\tau}_C = 0.016$;

c) repeating time $\tau_R = 0.5$;

d) weighting coefficient $\alpha = 5$ and $\beta = 10$.

Presented numerical examples prove a good convergence of the proposed algorithm. To find an optimal configuration only 7 steps in the first example and 12 steps in the second example have been needed. It proves the quality of the Lower Bound.

The value of the objective function decreases in the first example from 25.38 at the beginning to the 16.13, for the optimum (i.e. 36%). For the second example the value of the objective function decreases from 275.31 to 208.98 (24 %).

5. CONCLUSION

The problem of resources allocation for studied distributed production system can be formulated as a problem of parameter's optimization in a queuing network.

Under some assumptions about kind of flow process and processing times, an analytical solution can be obtained for primary evaluation of possible decisions. In this paper, such analytical results are obtained for multi-channel queuing systems with a special structure.

Table 3 Results of for model with 5 PNs and 15 servers

NC	PN_1	PN_2	PN_3	PN_4	PN_5	Cost	Time	Utilis.	Total
15	0	0	0	0	0	30	7.8126	20.625	275.313
14	0	0	0	0	1	29	8.1251	19.417	263.792
13	0	0	0	1	1	28	8.4250	18.217	252.292
12	0	0	1	1	1	27	8.7063	17.029	240.823
11	1	0	1	1	1	26	8.9563	15.863	229.406
10	1	1	1	1	1	25	9.1439	14.738	218.094
9	1	1	1	2	1	24	9.4208	14.553	216.632
8	1	1	1	2	2	23	9.6712	14.386	215.217
7	1	1	1	3	2	22	9.9182	14.221	213.805
6	1	1	2	3	2	21	10.169	14.055	212.399
5	1	1	2	4	2	20	10.384	13.914	211.062
4	2	1	2	4	2	19	10.620	13.772	209.823
3	**2**	**1**	**3**	**4**	**2**	**18**	**11.018**	**13.589**	**208.984**

For the optimization of the parameters of the queuing models, a branch and bound algorithm is proposed. This algorithm uses analytical method for Lower Bound calculation and a simulation procedure for final results analysis.

Conducted tests confirmed that value of objective function is critical to resources allocation in the queuing networks. The proposed algorithm give an optimal resources allocation decision.

The simulation also provides the following results:

a) validation of the analytical model by simulation confirms a good adequacy of the model,

b) proposed optimization algorithm has a fast convergence.

REFERENCES

Azadivar, F. and Y. Lee, (1988). Optimization of discrete variable stochastic systems by computer simulation. *Mathematics and Computers in Simulation*, **2**, 331-345.

Biethan, J. and V. Nissen, (1994). Combinations of simulation and evolutionary algorithms in management science and economics. *Annals of Operations Research*, **32**, 183-208.

Buzacott, J. and J. Shanthikumar, (1993). *Modeling and analysis of manufacturing systems*. Wiley&Sons, N.Y.

Chee Hock, Ng., (1997). *Queuing Modelling Fundamentals*: John Wiley & Sons, New York.

Dolgui, A. and O. Zaikin, (2000). Queuing Models for a capacity allocation problem. In: *Proceedings of the 14th European Simulation Multiconference "Simulation and Modelling: Enablers for Better Quality of Life"*, May 23-26, 2000, pp. 315-317. Ghent, Belgium.

Gordon W., and G. Newell, (1967). Closed Queuing Systems with Exponential Servers. *Operation Research*, **15**, 254-265.

Guariso, G., M. Hitz and H. Werthner, (1996). An integrated simulation and optimization modeling environment for decision support. *Decision Support Systems*, **1**, 103-117.

Hall, R.W., (1991). *Queuing methods for service and manufacturing*. Prentice Hall, Englewood Cliffs. N. Y.

Kelton, W.D., R.P. Sadowski and D.A. Sadowski, (1997). *Simulation with Arena*, McGraw-Hill, N. Y.

Zaikin O. , A. Dolgui and P. Kraszewski, (2000). Queuing modeling of the resource assignment in the High Tec assembly manufacturing. In: *New Frontiers in Computational Intelligence and its Applications*, M. Mohammadian (Ed.), pp. 340-346. Amsterdam, IOS Press.