Chapter 121
Modeling and Simulation of Troubleshooting Process for Automobile Based on Petri Net and Flexsim

Wei-zhen Liao, Kai-hu Hou, Yu-jie Zheng and Xiao-ling He

Abstract In this paper, Yunnan Y Automotive Company is taken as a research subject. Firstly, the troubleshooting methods from shallower to the deeper were summarized to find the origin of the defects quickly and efficiently. Then, the model of troubleshooting method was built by Petri net. What is more, the model was corresponded to single-queue and multiserver model and the optimal value of C corresponding to the optimal number of the quality engineer was calculated adopting marginal solving method. Finally, the results of calculation were simulated and verified using the simulation software Flexsim. Results indicate the troubleshooting method can detect the fault source efficiently.

Keywords Fault source · M/M/C queuing model · Modeling and simulation · Petri net · Troubleshooting

121.1 Introduction

Domestic studies on automotive failure: among the literatures, some scholars (Song and Yao 2009) used thousands of car failure data about auto after-sales service, from the perspective of statistical theory analysis to establish the reliability of theoretical models; some scholars (Luo and Zhu 2005) used a new pattern of support vector machine identification method to analysis and forecast short-term after-sales failure data, it was more reliable; in addition, reasoning

W. Liao (✉) · K. Hou · Y. Zheng · X. He
Department of Industrial Engineering, Kunming University of Science and Technology, Kunming, People's Republic of China
e-mail: 897056987@qq.com

K. Hou
e-mail: kaihu_kmustie@163.com

E. Qi et al. (eds.), *The 19th International Conference on Industrial Engineering and Engineering Management*, DOI: 10.1007/978-3-642-38391-5_121,

method which was widely used in the automotive fault diagnosis expert system is fault tree and fuzzy set theory (Kong and Dong 2001; Ji 2003; Su 2011). In conclusion, most of the literature on the automotive failure analysis is simply about the after-sales data, although the created model has a certain reference value, but there is a strong lag; the fault tree, fuzzy set analysis methods can not be a good handling with fuzzy and concurrency of car fault feature extraction; in addition, these literature did not have simulation, their reliability needs to be elegant.

To make a breakthrough of three aspects which are mentioned above, this paper focuses on the process of cars production, to identify a process-oriented and experience-oriented approach that can gradually find fault source; combines with Petri Net and Flexsim to finish the modeling and simulation, handling with fuzzy, parallelism and concurrency of cars fault; using queuing theory to make quantitative analysis is more reliable.

121.2 About Petri Nets

Petri Net is a system model that uses P-element to represent the state, uses T-element to represent the changes and associates resources (material, information) flowing. Overall, it contains the state (Place), change (Transition), and flow.

So its mathematical definition (Su 2011) is a triple $N = (P, T; F)$

$P = \{p_1, p_2,\ldots,p_n\}$ is Place set, n is the number of the Place;
$T = \{t_1, t_2,\ldots,t_m\}$ is Transition set, m is the number of Transition;
F is a Set of ordered pair that consists of a P-element ana a T-element. And it meets $F \subseteq (P \times T) \cup (T \times P)$.

The characteristics of Petri Nets are mainly reflected in two aspects: first, it is realizability, the Petri Net systems must ensure that each Transition meets the laws of nature, so it can be achieved; the most prominent feature of Petri Nets is suitable for description and analysis of asynchronous concurrent systems on the various levels of abstraction.

121.3 Car Troubleshooting Method

In order to solve quality problems, the old and the new seven tools of quality management have been widely used in all aspects of business operations. These methods in the practical application have their respective strengths and focus, but they do not meet the authors' requirements: to identify the sources of the failure efficiently, step by step, from easy to difficult (Fig. 121.1 and Table 121.1).

According to the production of cars, combined with the experience of the staffs of the Yunnan Y Automotive Company, a new troubleshooting method is born.

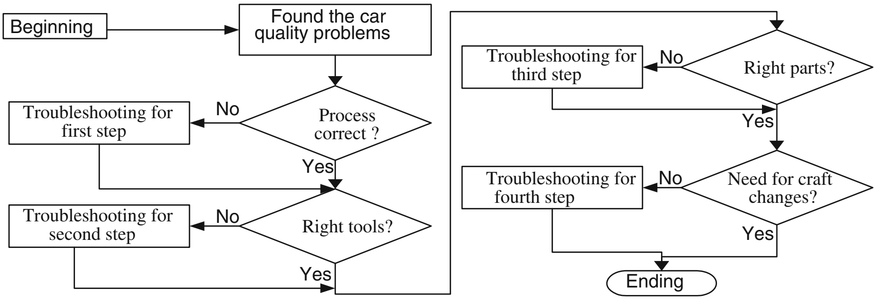

Fig. 121.1 Diagram for troubleshooting process

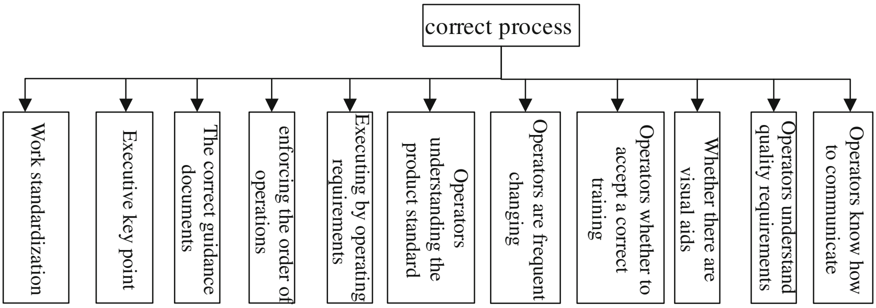

Fig. 121.2 Problems need to be shooting in the first step

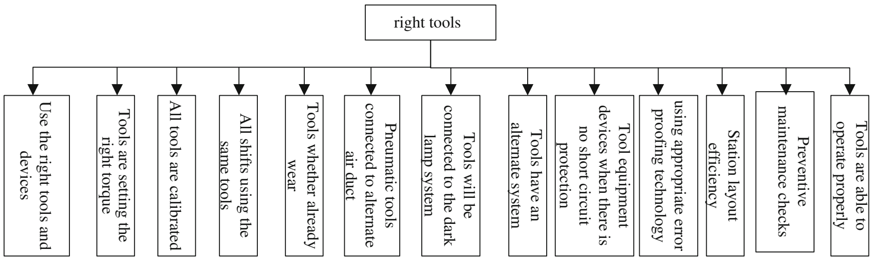

Fig. 121.3 Problems need to be shooting in the second step

It is the most effective method to determine the defect source of a standardized framework. It is used to promote the problem-solving. It can be decomposed into the following four steps: step 1—the correct process; step 2—the right tools; step 3—the right parts; step 4—the need for craft changes.

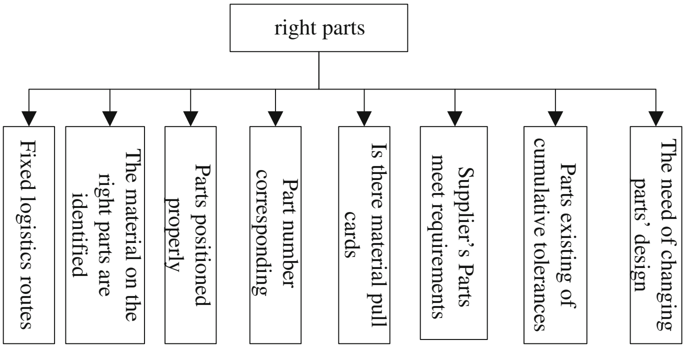

Fig. 121.4 Problems need to be shooting in the third step

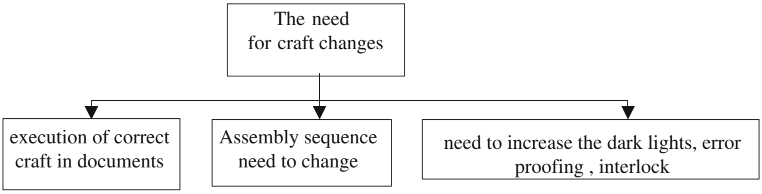

Fig. 121.5 Problems need to be shooting in the fourth step

121.4 Modeling and Analysis Combine with Petri Nets for Automotive Troubleshooting Process

121.4.1 Modeling with Petri Nets

Figure 121.6 is built models with basic Petri Net (Ren and Hao 2010; Su and Shen 2007; Xue et al. 2006). From Fig. 121.6, we know, nodes of the system are too much and model is too huge (Bourjij et al. 1993). In order to better express the logic relationship, so we references to the thought of Colored Petri Net (Wu and Yang 2007): through introduction of color transition and replacement reduces the complexity of system Petri Network model, making model intuitive and simple. modeling by using CPN-tools (Vinter et al. 2003) simplified as Fig. 121.7.

From Fig. 121.7, readers can stick out a mile to the entire car troubleshooting process; connection with Fig. 121.6, readers can get detailed reference, also can grasp from the overall and specific well for the whole troubleshooting process.

Table 121.1 The meaning of place and transition in Fig. 121.6

	Meaning
Place elements	
p_1	Appear car quality problems
p_{11}, p_{13}, …, p_{119}	No error state by searching 10 problems in Fig. 121.2
p_{12}, p_{14}, …, p_{120}	Error state by searching 10 problems in Fig. 121.2
p_2	The car still have failure after the first step
p_{21}, p_{23}, …, p_{215}	No error state by searching 8 problems in Fig. 121.3
p_{22}, p_{24}, …, p_{216}	Error state by searching 8 problems in Fig. 121.3
p_3	The car still have failure after the second step
p_{31}, p_{33}, …, p_{325}	No error state by searching 13 problems in Fig. 121.4
p_{32}, p_{34}, …, p_{326}	Error state by searching 13 problems in Fig. 121.4
p_4	The car still have failure after the third step
p_{41}, p_{43}, p_{45}	No error state by searching 3 problems in Fig. 121.5
p_{42}, p_{44}, p_{46}	Error state by searching 3 problems in Fig. 121.5
p_5	The car trouble shooting, returned to normal
Transition elements	
t_{11}, t_{12}, …, t_{110}	Search the corresponding 10 problems in Fig. 121.2 one by one
t_{21},t_{22}, …, t_{210}	Solve the problem of the first step
t_3	Check if the car can run normally
t_{41}, t_{42}, …, t_{48}	Search the corresponding 8 problems in Fig. 121.3 one by one
t_{51}, t_{52}, …, t_{58}	Solve the problem of the second step
t_5	Check if the car can run normally
t_{61}, t_{62}, …, t_{613}	Search the corresponding 13 problems in Fig. 121.4 one by one
t_{71},t_{72}, …, t_{713}	Solve the problem of the third step
t_8	Check if the car can run normally
t_{91}, t_{92}, t_{93}	Search the corresponding 3 problems in Fig. 121.5 one by one
t_{101}, t_{102}, t_{103}	Solve the problem of the fourth step
t_{10}	Start the car to show normal

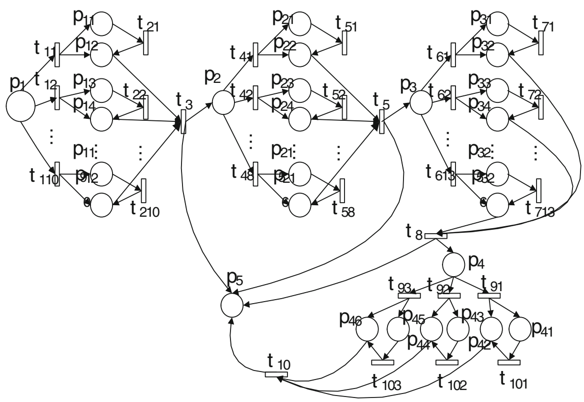

Fig. 121.6 Modeling of automotive troubleshooting based on Petri Net

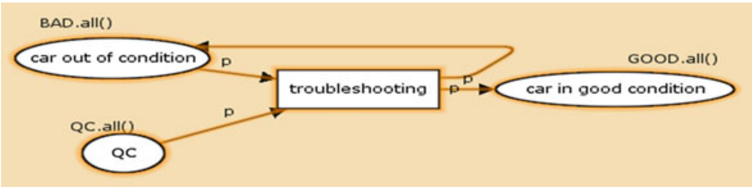

Fig. 121.7 Modeling of automotive troubleshooting based on Colored Petri Net

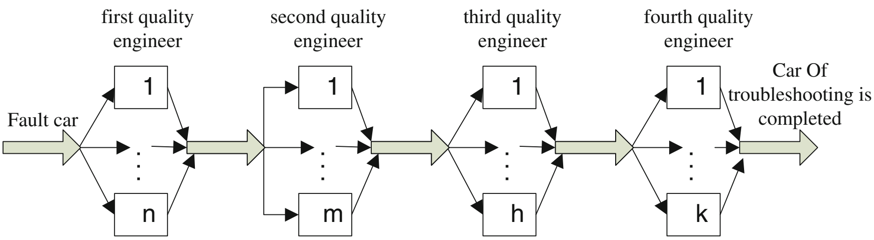

Fig. 121.8 Queuing model of car troubleshooting

121.4.2 With the Queue Theory Knowledge to Determine the Optimal Parameters of the Model (Service Desk C)

In Petri Nets model, if regarding problems that every step need to search as object (customers) waiting for service, the quality engineer as the service desk, and more than one quality engineer can work simultaneously, then the whole process can be simplified into the following multi-server single queue tandem queuing system model:

Assuming that the arrival of customers (fault cars) obey the Poisson, the quality engineer‘s checking time obeys negative exponential distribution, makes each step independent, then this model is the M/M/C queuing model. In the situation that the probability of cost and fault car basically turns to stabilized, to determine the optimal number of service desk can reduce costs and maximize the benefits. Seeking the optimal number of service desk is obtained in Fig. 121.8 n, m, h, k.

To improve the reliability and accuracy of the calculation, using the following references (Ai et al. 2007) to seek the method of M/M/C model optimal service desk number C.

In steady-state case, the expectations of the unit time full cost (service costs and waiting costs):

$$z = c_s' \cdot c + c_w \cdot L \tag{121.1}$$

where c is the number of service desk; c_s' is the unit time cost of each desk; c_w is the unit time costs of each customer stay in the system; L is the system's customer average number L_s or the queue's customer average L_q, the service desk number setting up has the deep impact on it. Because c_s' and c_w can get the statistics through the actual situation, so (121.1) is the function $z(c)$ about c, the purpose is through getting the optimal solution c^* makes $z(c)$ minimize.

c^* is an integer, using the marginal analysis method:

$$\begin{cases} z(c^*) \leq z(c^* - 1) \\ z(c^*) \leq z(c^* + 1) \end{cases} \tag{121.2}$$

Substituting 'z' of formula (121.1) into formula (121.2), then

$$\begin{cases} c_s' \cdot c^* + c_w \cdot L(c^*) \leq c_s' \cdot (c^* - 1) + c_w \cdot L(c^* - 1) \\ c_s' \cdot c^* + c_w \cdot L(c^*) \leq c_s' \cdot (c^* + 1) + c_w \cdot L(c^* + 1) \end{cases} \tag{121.3}$$

Simplify formula (121.3), then

$$L(c^*) - L(c^* + 1) \leq {c_s'}/{c_w} \leq L(c^* - 1) - L(c^*) \tag{121.4}$$

According to the related analysis to thousands of cars failure data of Yunnan Y Automotive Company collected at the scene, the car failure source in the four steps of the searching, which belongs to the first step is about 25 %, the second step is about 40 %, about 30 % is the third step, the last step is only 5 %. Due to the influence of external factors, every step's average service time is different: 240, 300, 300, 600 s, and obey exponential distribution.

Yunnan Y Automotive Company's internal data shows that: the costs of each searching incurred due to delays in other processes about 8 Yuan, the services costs to each time set an quality engineer (salaries and equipment wear and tear) for about 37 Yuan per hour.

121.4.2.1 The Determination of the Numbers of the Optimal Service Desks in the First Step

According to the statistics, in the first step, the fault cars arrived time obeys the average arrival rate of 26 times per hour to the Poisson distribution; service time is negative exponential distribution which average service rate is 15 times per hour.

So, $c_s' = 37$ Yuan/quality engineer, $c_w = 8$ Yuan/time, $\lambda = 26$, $\mu = 15$, $\lambda/\mu = 1.73$, assuming the number of quality engineers to c, makes c respectively as 1, 2, 3, 4, 5. According to the $W_q \cdot \mu$ value of multi-server desks (Ai et al. 2007), with linear interpolation algorithm to find the corresponding value of $W_q \cdot \mu$, as shown in the Table 121.2.

Substituted L_s into the formula (121.4), obtained the following data in Table 121.3 by the formula (121.1):

From Table 121.3, 128.09 Yuan is the lowest total cost, the corresponding c equals to 3, so the result coming out is the lowest cost that needs to set three quality engineers.

Table 121.2 Average number of customers (L) in the system

c	1	2	3	4	5
$\lambda/c\mu$	1.730	0.865	0.577	0.433	0.346
$W_q \cdot \mu$	–	3.393	0.235	0.054	0.012
$L_s = \frac{\lambda}{\mu}\left(W_q \cdot \mu + 1\right)$	–	7.600	2.136	1.823	1.751

Table 121.3 Data calculated by marginal analysis

The number of quality engineers c	The number of car waiting to search $L_s(c)$	$L(c) - L(c+1) \sim L(c) - L(c-1)$	The total costs (every hour) $z(c)$
1	∞	∞	∞
2	7.6000	5.4640–∞	134.80
3	2.1360	0.313–5.4640	128.09
4	1.8230	0.116–0.313	162.58
5	1.7510	–	199.01

121.4.2.2 The Determination of the Numbers of the Optimal Service Desks in the Second Step

Easy to know that fault cars reached in the second step still obey to Poisson distribution (Winston 2004), if N is Poisson random variables, then $E(N) = \operatorname{var} N = \lambda$. So the fault cars from the first step into the second step reduce 25 %, then $\lambda = 26 \times 75\,\% = 19.5 \approx 20$, service time also obeys negative exponential distribution, $\mu = 12$ times per hour.

It can be gained the smallest total cost is 109.30 yuan, corresponding to the optimal service desks c = 2, this step sets two quality engineers is best.

121.4.2.3 The Determination of the Numbers of the Optimal Service Desks in the Third and the Fourth Step

Similarly, it is known that the fault cars reached in the last two steps obey Poisson distribution with $\lambda = 26 \times 35\% = 9.1$, $\lambda = 1.3$ respectively; service time also obeys index distribution, Service rate are $\mu = 12$, $\mu = 6$ times per hour. Because of the average arrival rate λ less than service rate μ, so the last two steps only need to set up 1 quality engineer.

According to the above theoretical calculation result, it is known that first step should set three quality engineers, the second step shall set up two quality engineers, the last two steps shall just set one quality engineer that can make the benefit of the system to achieve optimal.

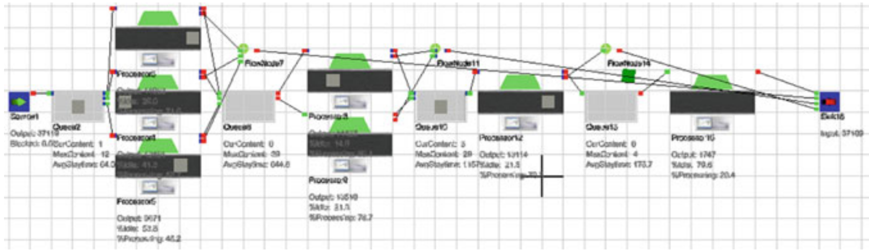

Fig. 121.9 Modeling plan

121.5 Flexsim Simulation Analysis

121.5.1 Modeling and Parameter Settings

According to the previous section, fault car's arrival interval time is about 138 s, average service time is exponentially distributed, their means, respectively, as 240, 300, 300, and 600 s (Chen et al. 2007). Using Flexsim to modeling, the plan is shown in the Fig. 121.9.

As shown in Fig. 121.9: using a Source as fault car's generator; four Queues to achieve car's cache; Processor represents quality engineer, there are different proportion of car finished troubleshooting inflow to Sink, so uses three Flow Node to provide path, achieve the shunting of the first three steps.

121.5.2 Analysis of Simulation Data

121.5.2.1 Confirm the Simulation Time

According to the work schedules of Yunnan Y Automotive Company, the company belongs to the mode of two shifts 1 day, 8-hour one shift, two for 16 h, so the length of simulation time is 16 h.

121.5.2.2 Confirm the Simulation Method

Using independent replications (Lin 2003) to simulate, namely, use of different random variables and select independent starting state to simulate R times.

121.5.2.3 Confirm the Simulation Times of Established Interval

Utilization of different random variables, after R simulations, generated R observed values (average values $\hat{\theta}_r$), using the R observed values to conduct point estimation values is as follows:

$$\hat{\theta} = \frac{1}{R}\sum_{r=1}^{R} \hat{\theta}_r \tag{121.5}$$

The sample standard deviation:

$$S^2 = \sum_{r=1}^{R} \frac{(\hat{\theta}_r - \hat{\theta})^2}{R-1} \tag{121.6}$$

The estimate value $\hat{\theta}$:

$$\hat{\sigma}^2(\hat{\theta}) = \frac{S^2}{R} = \frac{1}{(R-1)R}\sum_{r=1}^{R}\left(\hat{\theta}_r - \hat{\theta}\right)^2 \tag{121.7}$$

In order to confirm the optimal times of simulation, assuming that a parameter's interval estimation within the scope of a particular, half-length interval is less than a certain value ε, we need to simulate at least R times to meet the required half interval length. Just starts our first R_0 simulations, general 4–5 times. Through formula (121.6) obtains standard deviation S_0, when the half-length interval less than ε, can be expressed as follows:

$$h.l. = t_{\alpha/2,R-1}\hat{\sigma}(\hat{\theta}) \leq \varepsilon \tag{121.8}$$

Formula (121.8) substituted the formula (121.7), then:

$$R \geq \left(\frac{t_{\alpha/2,R-1}S_0}{\varepsilon}\right) \tag{121.9}$$

Owing to $t_{\alpha/2,R-1} \geq z_{\alpha/2}$ ($z_{\alpha/2}$ is the $\alpha/2$ quantile of the standard normal distribution), so R meets the following minimum integer, and $R \geq R_0$, then:

$$R \geq \left(\frac{z_{\alpha/2}S_0}{\varepsilon}\right)^2 \tag{121.10}$$

Utilization of formula (121.5) to find out the point estimation value of each Processor in 5 different random variables simulation is shown in Table 121.4. It is clear that each Processor's utilization remains relatively low, so the three Processors of the first step should be reduced to two, then analysis the simulation. The data obtained from simulation analysis is shown in Table 121.5:

Five simulations is shown in Table 121.5, the data gained from Table 121.5 substituted the formula (121.6) to (121.10), followed by count $\hat{\sigma}^2(\hat{\theta})$, S_0^2, $(\frac{z_{0.025}S_0}{\varepsilon})^2$.

Table 121.4 Simulation data of first model

Times	Processor						
	1	2	3	4	5	6	7
1 (%)	77.70	65.50	55.74	88.72	83.39	71.33	16.93
2 (%)	72.02	58.28	47.33	80.79	73.68	82.70	21.47
3 (%)	72.04	61.21	45.20	77.87	69.90	72.37	22.05
4 (%)	69.53	60.00	47.78	79.66	71.15	77.03	20.40
5 (%)	70.64	62.62	54.54	80.83	71.91	81.58	11.21
Mean	72.39	61.52	50.12	81.57	74.01	77.00	18.41

Table 121.5 Simulation data of second model

Times	Processor					
	1	2	3	4	5	6
1 (%)	89.66	81.84	78.49	68.69	70.84	21.86
2 (%)	99.28	98.25	82.59	81.25	90.40	23.22
3 (%)	91.05	86.26	84.17	77.84	70.32	22.06
4 (%)	90.47	86.47	79.69	70.79	73.27	21.29
5 (%)	79.83	71.00	78.31	69.92	70.23	25.33
Means (%)	90.06	84.76	80.65	73.70	75.01	22.75

Table 121.6 Simulation times had been determined

$\hat{\sigma}^2(\hat{\theta})\,(\times E-03)$	0.96	1.93	0.14	0.61	1.51	0.05
$S_0^2\,(\times E-03)$	4.77	9.63	0.68	3.05	7.56	0.26
$\left(\frac{z_{0.025}S_0}{\varepsilon}\right)^2$	7.34	14.8	1.05	4.69	11.6	0.40
Simulation times R	8	15	5	5	12	5

Ultimately to determine how many times need to be simulated if 95 % processor busy rates of the confidence interval between ±0.05. The result is shown in Table 121.6:

In the whole model, the highest simulation times determine the final simulation times. From Table 121.6, knowing that should simulate more than 12 times, if we takes 15 times, 10 more times need to be simulated to insure that processors' busy rates could be drawn between the confidence interval ±0.05. The resulting data of the entire simulation process is shown in Table 121.7.

Confidence intervals can know from Table 121.7, half-length interval of 95 % confidence interval for all processors is less than 0.05. Finally it is turned out to be confidence in the 95 %: busy rates of Processor1 will fall in the interval 0.8843 ± 0.0425; busy rates of Processor2 will fall in the interval 0.8321 ± 0.0363; busy rates of Processor3 will fall in the interval 0.8126 ± 0.0375; busy rates of Processor4 will fall in the interval 0.7387 ± 0.0421; busy rates of Processor5 will

Table 121.7 Data analysis of the whole simulation process

Value	Processor					
	1	2	3	4	5	6
Mean	88.43	83.21	81.26	73.87	72.58	21.35
Standard deviation	0.0549	0.0797	0.0392	0.0544	0.0739	0.0447
Confidence interval	±0.0425	±0.0363	±0.0375	±0.0421	±0.0407	±0.0428

fall in the interval 0.7258 ± 0.0407; busy rates of Processor6 will fall in the interval 0.2135 ± 0.0428.

Combined with the calculation of third section and results of two different models' simulation, in the first step two quality engineers are more reasonable; the efficiency of Processor6 is still low, to further improve the efficiency and benefits of system, should train the technology-packed quality engineers. The problems of the last two steps will be finished by one quality engineer.

121.6 Conclusion

Automotive troubleshooting method that is described in this paper is a more universal method to find out the fault source. It is from simple to complex, is trending to standardization procedures to solve problems.

Modeling with Petri Net and Colored Petri Net, from global and local troubleshooting on cars have a good control and master; effectively solve the parallel reasoning of the car fault detection system; reduce the complexity of the fuzzy inference reasoning method.

And collecting data in real-time from car production process, the use of queuing theory related knowledge obtains the optimal number of quality engineers.

Modeling and Simulation with Flexsim have great reference value for reasonable arrangements to quality inspectors. Using of the troubleshooting method makes all activities of the enterprises with strong timeliness.

References

Ai Y et al (2007) Operations research, vol 11. Tsinghua University Press, Beijing, pp 336–337 (in Chinese)

Bourjij A, Zasadzinski M, Darouach M, Krzakala G, Musset M (1993) On the use of hybrid Petri Nets for control process safety: application to a steam2boilers network simulator. In: IEEE international conference on systems, man and cybernetics, no. 2

Chen G, Wu H, Chen Y (2007) Industrial engineering and system simulation, vol 6. Metallurgical Industry Press, Beijing, pp 79–253 (in Chinese)

Ji C (2003) Development of automotive diagnostic system based on fault tree. Veh Power Technol 1:52–57 (in Chinese)

Kong F, Dong Y (2001) Failure diagnosis and reasoning based on fault tree knowledge. Automot Eng 23(3):209–213 (in Chinese)
Lin Z (2003) Theory and application of system simulation. Press by Canghai Bookstore, Nanchang, Jiangxi, China, p 357 (in Chinese)
Luo Z, Zhu S (2005) A new ε-insensitivity function support vector inductive regression algorithm and after-sales service data model forecast system. Comput Sci 32(8):134–141 (in Chinese)
Ren J, Hao J (2010) Petri network-based modeling analysis. J Xi'an Aerotech College 28(3):50–52 (in Chinese)
Song L, Yao X (2009) Research on probability model of vehicle quality based on the broken-down number per thousand cars. J Chongqing Technol Business Univ 26(6):543–547 (in Chinese)
Su C (2011) Modeling and simulation for manufacturing system. Mechanical Industry Press, Beijing, p 120 (in Chinese)
Su C, Shen G (2007) Development for system reliability modeling and simulation based on generalized stochastic Petri net (GSPN). Manuf Inf Eng China 36(9):45–48 (in Chinese)
Vinter RA, Liza W, Henry Machael L, et al (2003) CPN tools for editing, simulting, and analysing coloured petri net. In: Proceeding of the applications and theory of petri nets presented at the 24th international conference, Eindhoven, The Netherlands
Winston WL (2004) Operations research introduction to probability models, 4th edn. A Division of Thomson Learning Asia Pte Ltd, Belmont, pp 333–336
Wu H, Yang D (2007) Hierarchical timed coloured petri-net based approach to analyze and optimize medical treatment process. J Syst Simul 19(4):1657–1699 (in Chinese)
Xue L, Wei C, Chen Z (2006) Modeling design and simulation of hybrid systems review and analysis. Comput Simul 23(6):1–5 (in Chinese)