
一个多节点声纳系统中同步时钟机制的 可靠性评估和系统优化问题

冯绍庭 520021911362

摘要：本文以某种包含主从节点和时钟总线的多节点声纳系统作为研究对象，利用统计学和蒙特卡洛方法，探究该系统中同步时钟机制的可靠性、平均寿命以及相对应的最优节点数量。由于系统随机性较强，需要采用蒙特卡洛方法大量模拟。本文将系统分层级建模，采用变步长马尔科夫链，探究系统性质。按照系统失效原因分析提出双总线系统作为优化方案，并且对原系统给出理论数值进行比较。

关键词：可靠性评估，系统平均寿命，蒙特卡洛算法，变步长马尔科夫链

Reliability evaluation and system optimization of synchronous clock mechanism in a multi-node sonar system

ABSTRACT: This paper takes a certain multi-node sonar system including master-slave nodes and clock bus as the research object, and uses statistics and Monte Carlo methods to explore the reliability, average life and corresponding optimal selection of the quantity of nodes of the synchronous clock mechanism in the system. Due to the strong randomness of the system, a large number of simulations using the Monte Carlo method are required. In this paper, the system is modeled hierarchically, and the variable-step Markov chain is used to explore the properties of the system. According to the analysis of system failure reasons, the dual bus system is proposed as an optimization scheme, and the theoretical values of the original system are compared.

Key words: reliability evaluation, system's average life, Monte Carlo algorithm, variable-step Markov chain

目录

1 引言	3
2 问题描述与模型抽象	3
2.1 切换器的状态	3
2.2 节点的状态	4
2.3 系统的状态	4
2.4 整体思路	5
3 结果分析与模型改进	5
3.1 可靠性测试	5
3.2 平均工作寿命测试	6
3.3 模型改进——双总线系统	7
4 理论求解	8
4.1 系统可靠性	8
4.2 系统平均寿命	9
5 结论	9
6 进一步工作任务	9
7 参考文献	9
8 附录	10
8.1 单总线系统的可靠性	10
8.2 单总线系统的平均寿命	14
8.3 双总线系统	17

1 引言

本课题假设一个多节点声纳系统，每个节点都是分布式部署的，一共有 n 个节点。一旦电子元件出现故障，就无法进行人工维修。为了保证系统的正常运行，各个节点必须保持严格的时钟同步，并通过时钟总线传输系统时钟信号。只有一个节点可以工作在主时钟模式，其他节点必须工作在从时钟模式。当主模式节点的时钟电路发生故障时，可以自动退出主模式，其余节点按照一定的机制随机选择一个节点接管主模式下的工作。

因此，我们提出了一种冗余设计，如图 1。在时钟正常同步的前提下，有 3 个节点正常工作才能使整个系统正常运行，节点总数 n 可以大于 3；当正常工作的节点数小于 3 时，系统失效。虽然这样尽可能地提高了电路的可靠性设计，但节点电路仍有可能出现阻塞时钟总线的故障，导致系统故障。因此，我们需要对系统的可靠性指标和平均工作寿命进行定量评价，求解出最优的 n 值。

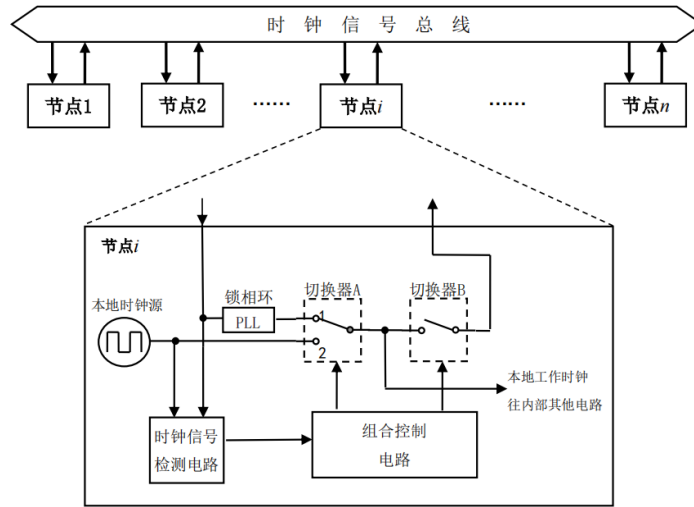


图 1 一个多节点声纳系统中的时钟同步机制示意图^[1]

2 问题描述与模型抽象

2.1 切换器的状态

切换器可能出现故障，如图 2、3。对于切换器 A， g_{A0} 表示其处于正常工作状态，而 g_{A1} 、 g_{A2} 、 g_{A3} 分别表示处于故障 A1、A2、A3 状态。同理，切换器 B 是一个 3 状态元件。

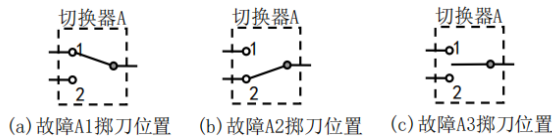


图 2 切换器 A 的故障类型^[1]

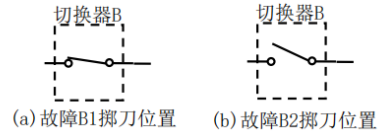


图 3 切换器 B 的故障类型^[1]

切换器 A 和 B 的使用寿命 T 的概率密度分布和各种故障概率如式 1-7 所示。在程序中，使用 `exprnd` 随机产生使用寿命，再使用轮盘赌判定其故障的类型。

$$f_{T_A}(\tau) = \lambda_A e^{-\lambda_A \tau} \text{ 其中 } \frac{1}{\lambda_A} = 5.90 \times 10^4 \text{ hour} \quad (1-7)$$

$$P_{EA1} = \Pr(A1 \text{ occurs} \mid A \text{ is failed}) = 0.20$$

$$P_{EA2} = \Pr(A2 \text{ occurs} \mid A \text{ is failed}) = 0.15$$

$$P_{EA3} = \Pr(A3 \text{ occurs} \mid A \text{ is failed}) = 0.65$$

$$f_{T_B}(\tau) = \lambda_B e^{-\lambda_B \tau} \text{ 其中 } \frac{1}{\lambda_B} = 2.20 \times 10^5 \text{ hour}$$

$$P_{EB1} = \Pr(B1 \text{ occurs} \mid B \text{ is failed}) = 0.45$$

$$P_{EB2} = \Pr(B2 \text{ occurs} \mid B \text{ is failed}) = 0.55$$

2.2 节点的状态

节点的状态由其内部的切换器状态的组合来确定，如表 1。其中 g_{N0} 表示节点性能完好，定义别名 g_{PF} （意为 perfectly functioning）； g_{N1} 表示只能作为从节点，别名 g_{SO} （意为 slave only）； g_{N2} 表示或者作为主节点，或者作为不阻塞总线的失效节点，别名 g_{DM} （disable/master）； g_{N3} 表示只能作为主节点，否则就会阻塞总线，别名 g_{MO} （master only）； g_{N4} 表示成为不阻塞总线的失效节点，别名 g_{DN} （disable node）； g_{N5} 表示节点总是阻塞总线，别名 g_{FB} （failed bus）。

表 1 切换器-节点状态映射关系^[1]

切换器 A 状态	切换器 B 状态	节点状态	切换器 A 状态	切换器 B 状态	节点状态
g_{A0}	g_{B0}	g_{PF}	g_{A2}	g_{B0}	g_{DM}
	g_{B1}	g_{MO}		g_{B1}	g_{MO}
	g_{B2}	g_{SO}		g_{B2}	g_{DN}
g_{A1}	g_{B0}	g_{SO}	g_{A3}	g_{B0}	g_{DN}
	g_{B1}	g_{FB}		g_{B1}	g_{DN}
	g_{B2}	g_{SO}		g_{B2}	g_{DN}

程序中，使用二维矩阵 `node_state`，其中每行对应一种切换器 A 状态，每列对应一种切换器 B 状态，根据切换器状态确定节点状态。将六种节点状态对应的节点个数分别定义为 PF，SO，DM，MO，DN 和 FB。

2.3 系统的状态

系统的整体工作状态由所有的 n 个节点状态的组合来确定。

- C1: $FB \geq 1$
- C2: $MO \geq 2$
- C3: $PF + MO + DM == 0$
- C4: $PF + SO + 1 (MO + DM > 0) < 3$
- C7: $(MO == 0 \&\& PF == 0 \&\& DM \geq 1 \&\& SO \geq 2) \mid \mid (MO == 0 \&\& PF \geq 1 \&\& PF + SO \geq 3)$
- C5: $FB == 0$
- C6: $MO == 1 \&\& PF + SO \geq 2$
- C8: $FB + MO == 0$
- C9: $PF \geq 1 \&\& DM \geq 1 \&\& PF + SO == 2$

当 C1、C2、C3、C4 至少有一个满足，则系统处于状态 1，即确定不能正常工作；当 C1~C4 都不满足，C5 满足且 C6 和 C7 中至少有一个满足，则系统处于状态 2，即确定正常工作；如果系统既不处在状态 1，也不处在状态 2，且 C8 和 C9 均满足，有 $DM / (DM + PF)$ 的概率进入状态 3，即恰好能正常工作，有 $PF / (DM + PF)$ 的概率进入状态 4，即恰好不能正常工作。

按照当前的判别方式，当 C8 和 C9 均满足时，无论上一次故障崩溃的节点是主节点还是从节点，无论那个节点原先为什么状态，都会触发主节点重选机制，即没有考虑“历史遗留问题”。

由于系统只可能处于这四种状态，所以程序只需要判断系统此时是否处于其中三种状态。

2.4 整体思路

根据问题要求，限定系统寿命最大值为 200000，节点个数范围是 3-20，在每个节点个数下，对每个切换器 A 和 B 的寿命进行随机数取值，即采用 `exprnd` 函数得到它们的寿命，之后采用**变步长算法**，在每一个切换器故障的时刻，判断相应节点的状态，并根据一定条件得到整个系统的状态。

由于求解可靠性和平均寿命时每个系统模拟的截止时间不同。考虑到求解单一目标（尤其是可靠性）的省时需求，故将两个问题的求解分为两个程序。

3 结果分析与模型改进

3.1 可靠性测试

系统可靠性定为系统工作寿命超过 30000 hour 的概率，用频率代替概率，采用蒙特卡洛方法，模拟 100000 套同型系统的运行状况。

表 2 可靠性测试

节点总数	3	4	5	6	7	8	9	10	11
Rw	0.3289	0.6285	0.8028	0.8789	0.9084	0.9122	0.9040	0.8883	0.8764
节点总数	12	13	14	15	16	17	18	19	20
Rw	0.8607	0.8429	0.8291	0.8079	0.7921	0.7741	0.7556	0.7384	0.7244

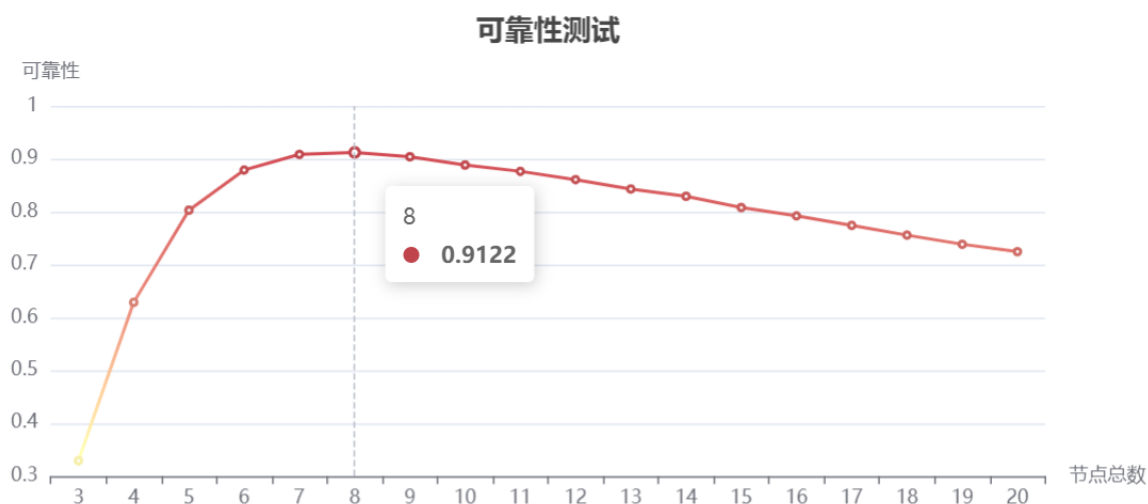


图 4 可靠性测试

实验结果如表 2 和图 4 所示。节点总数为 8 时，系统可靠性最大，为 0.9122。

引起系统失效的原因大体可以分为三类：有效节点数不足 3 个（C4 和 sys4），分别记为 bug_1 和 bug_2 ；主节点数量不足（C3），记为 bug_3 以及出现了阻塞总线的节点（C1 和 C2），记为 bug_4 。若同时出现多个 bug，按照 $bug_1 > bug_2 > bug_3 > bug_4$ 的优先级判定。在节点总数为 8 时再次实验，统计四种 bug 出现的比例，如图 6。可知**阻塞总线**为引起系统失效的主要原因，占失效系统的 87.0%。

3.2 平均工作寿命测试

首次失效时间，又称工作寿命，是系统从初始时间到首次发生失效的时间。我们可以将 S 套系统的首次失效时间求平均值，用来替代首次失效时间的期望。

表 3 平均工作寿命测试

节点总数	3	4	5	6	7	8
$E(T_f)/\text{hour}$	28674.135	51035.685	68533.874	81825.690	91120.871	98301.097
节点总数	9	10	11	12	13	14
$E(T_f)/\text{hour}$	102345.203	104114.649	104803.970	104459.807	102028.938	100004.064
节点总数	15	16	17	18	19	20
$E(T_f)/\text{hour}$	96947.873	93734.349	89986.919	86812.690	83118.476	79847.423

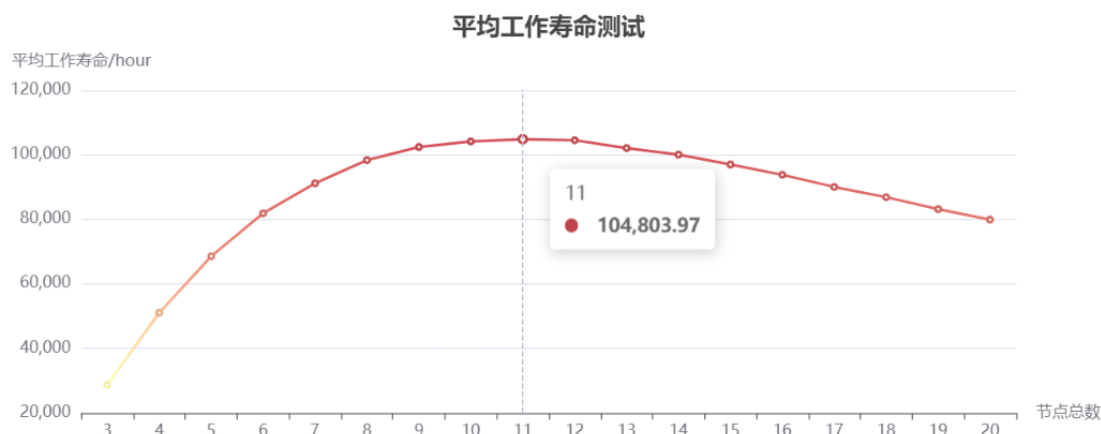


图 5 平均工作寿命测试

实验结果如表 3 和图 5 所示。节点总数为 11 时，系统平均工作寿命最大，为 104803.970 hour。统计节点总数为 11 时，四种 bug 出现的比例，如图 7。其中 r 定义为系统到达最大工作时长仍然没有失效。阻塞总线仍为引起系统失效的主要原因。

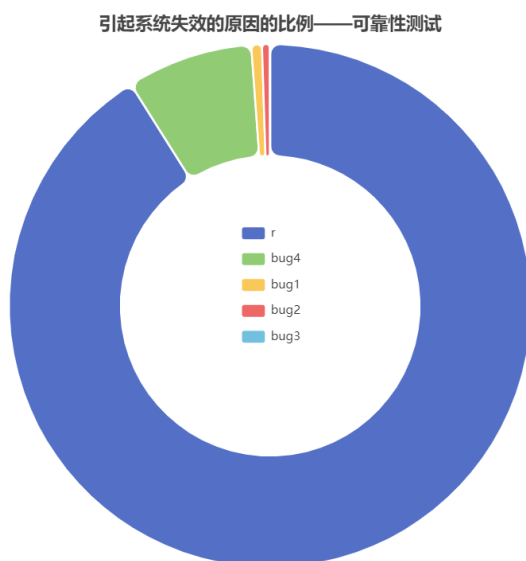


图 6 可靠性测试失效原因比例

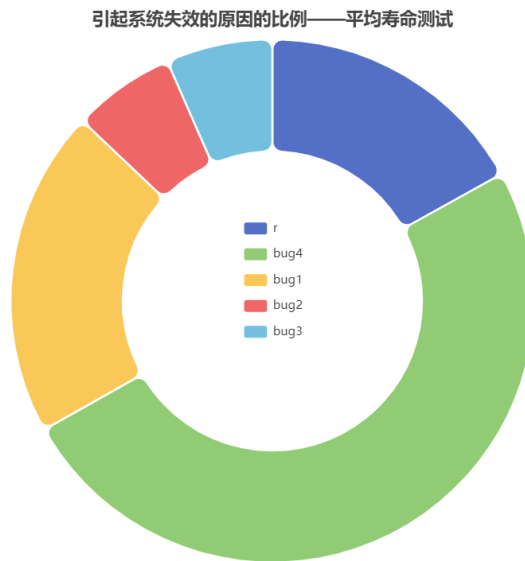


图 7 平均寿命测试失效原因比例

3.3 模型改进——双总线系统

由于阻塞总线为系统失效的主要原因，故可以采用**双总线结构**提高系统的可靠性和平均寿命。当一条总线被阻塞后，所有的节点切换到另一条总线工作。只有当两条总线均被阻塞时，系统才失效。切换器状态和节点状态的判别不变。系统状态的判别在[1]的基础上进行如下修正，其余条件定义与单总线系统相同。

- C1: $FB \geq 2$
 - C2-1: $FB == 1 \ \&\& \ MO \geq 2$
- C2-2: $FB == 0 \ \&\& \ MO \geq 3$
 - C8: $MO == 0 \ \&\& \ FB \leq 1$

当 C4 满足时，有效节点数不足 3 个，发生 bug_1 ；当 C8 和 C9 均满足，有 $\frac{PF}{PF+DM}$ 的概率有效节点数不足 3 个，发生 bug_2 ；当 C3 满足时，无主节点，发生 bug_3 ；当 C1、C2-1 和 C2-2 至少有一个满足，两条总线都被阻塞，发生 bug_4 ；其余情况下，系统正常工作。实验结果如表 4、5 和图 8、9 所示。

表 4 双总线系统可靠性测试

节点总数	3	4	5	6	7	8	9	10	11
Rw	0.3289	0.6387	0.8270	0.9206	0.9657	0.9826	0.9875	0.9886	0.9862
节点总数	12	13	14	15	16	17	18	19	20
Rw	0.9835	0.9794	0.9753	0.9694	0.9643	0.9568	0.9521	0.9439	0.9362

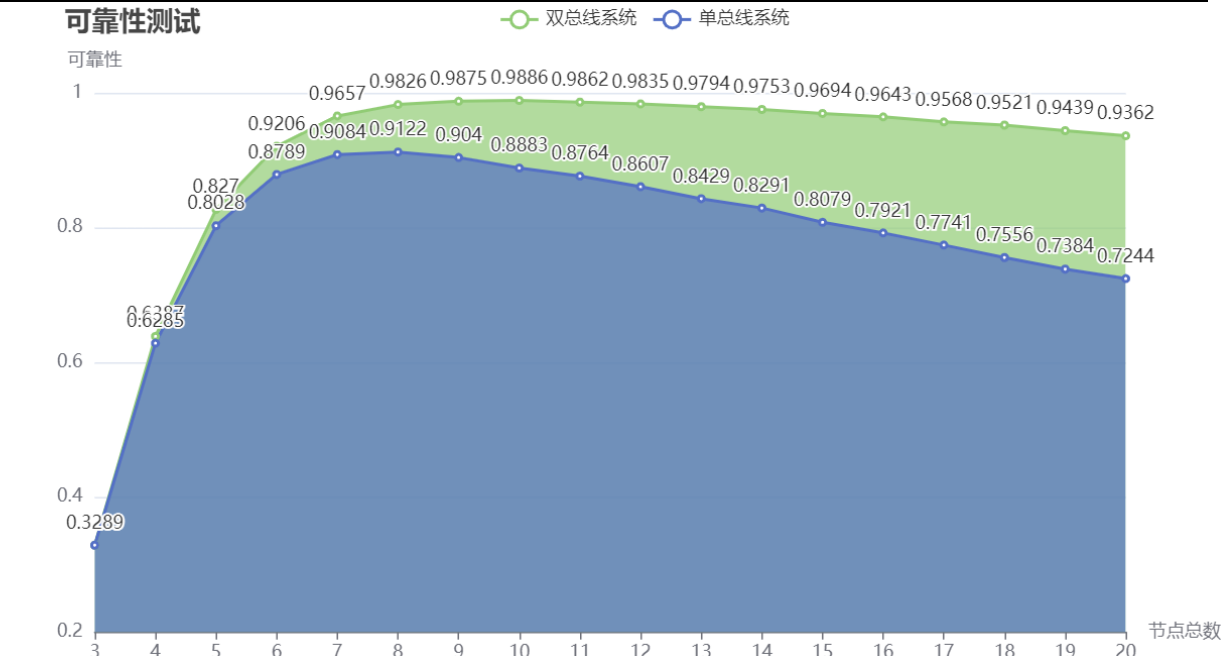


图 8 双总线系统可靠性测试

表 5 双总线系统平均工作寿命测试

节点总数	3	4	5	6	7	8
$E(T_f)/hour$	28663.651	52105.179	72199.265	89737.487	104647.195	116986.848
节点总数	9	10	11	12	13	14
$E(T_f)/hour$	126745.892	134115.243	139972.310	144254.858	146829.967	147790.725
节点总数	15	16	17	18	19	20
$E(T_f)/hour$	147937.232	147072.786	145385.843	143324.825	140506.110	137421.222

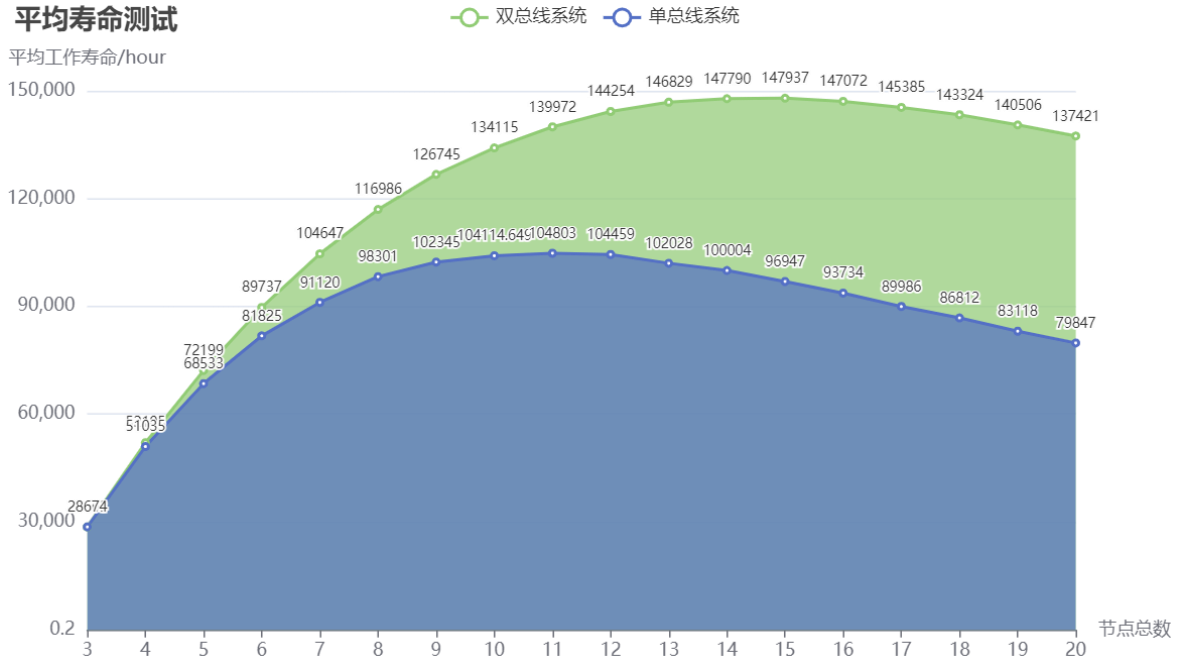


图 9 双总线系统平均寿命测试

随着节点总数的提高，系统失效的原因是阻塞总线的可能性提高，所以双总线对系统可靠性和平均寿命的提升更加明显。当使用双总线系统时，使用 10 和 15 个节点可分别使可靠性和平均寿命最大。

4 理论求解

4.1 系统可靠性

系统可用性是指系统在时刻 t ，瞬时状态为正常工作状态的概率。由于系统会发生先失效而后又恢复功能的所谓“复活”现象，所以，可用性数值会比可靠性为高。而“复活”现象的发生概率比较小，所以用系统可用性计算结果作为可靠性近似解是合理的。下面通过数学推导出系统可用性。

当系统处于 t 时刻，切换器 A、B 各个状态下的概率如式 8-14。

$$\begin{aligned}
 p_{A0}(t) &= e^{-\lambda_A t} \\
 p_{A1}(t) &= 0.20(1 - e^{-\lambda_A t}) \\
 p_{A2}(t) &= 0.15(1 - e^{-\lambda_A t}) \\
 p_{A3}(t) &= 0.65(1 - e^{-\lambda_A t}) \\
 p_{B0}(t) &= e^{-\lambda_B t} \\
 p_{B1}(t) &= 0.45(1 - e^{-\lambda_B t}) \\
 p_{B2}(t) &= 0.55(1 - e^{-\lambda_B t})
 \end{aligned} \tag{8-14}$$

于是，我们可以确定节点的六种状态所对应的概率如式 15-20。

$$\begin{aligned}
 P_{PF}(t) &= p_{A0}(t)p_{B0}(t) \\
 P_{MO}(t) &= p_{A0}(t)p_{B1}(t) + p_{A2}(t)p_{B1}(t) \\
 P_{SO}(t) &= p_{A0}(t)p_{B2}(t) + p_{A1}(t)p_{B0}(t) + p_{A1}(t)p_{B2}(t)
 \end{aligned} \tag{15-20}$$

$$\begin{aligned}
P_{FB}(t) &= p_{A1}(t)p_{B1}(t) \\
P_{DM}(t) &= p_{A2}(t)p_{B0}(t) \\
P_{DN}(t) &= p_{A2}(t)p_{B2}(t) + p_{A3}(t)(p_{B0}(t) + p_{B1}(t) + p_{B2}(t))
\end{aligned}$$

通过各个节点的状态概率，我们可以进一步分析系统的状态概率如式 21。

$$A(t) = C_5(t) \times (C_6(t) + C_7(t)) + C_8(t) \times C_9(t) \times Pr(t) \quad (21)$$

其中各参量如式 22-27。

$$\begin{aligned}
C_5(t) &= (1 - P_{FB}(t))^n \quad (22-27) \\
C_6(t) &= C_n^1 P_{MO}(t) (1 - P_{MO}(t))^{n-1} \left(\sum_{i=2}^{n-1} C_{n-1}^i (P_{PF}(t) + P_{SO}(t))^i (1 - P_{PF}(t) - P_{SO}(t))^{n-1-i} \right) \\
C_7(t) &= (1 - P_{MO}(t))^n (1 - (1 - P_{PF}(t))^n) \left(\sum_{i=3}^{n-1} C_{n-1}^i (P_{PF}(t) + P_{SO}(t))^i (1 - P_{PF}(t) - P_{SO}(t))^{n-1-i} \right) \\
&\quad + (1 - P_{MO}(t) - P_{PF}(t))^n (1 - (1 - P_{DM}(t))^n) \left(\sum_{i=2}^{n-1} C_{n-1}^i P_{SO}(t)^i (1 - P_{SO}(t))^{n-1-i} \right) \\
C_8(t) &= (1 - P_{FB}(t) - P_{MO}(t))^n \\
C_9(t) &= C_n^2 (P_{PF}(t) + P_{SO}(t))^2 (1 - P_{PF}(t) - P_{SO}(t))^{n-2} (1 - (1 - P_{PF}(t))^n) (1 - (1 - P_{DM}(t))^n) \\
Pr(t) &= \frac{P_{DM}(t)}{P_{DM}(t) + P_{PF}(t)}
\end{aligned}$$

A(w)为理论求解系统可靠性的结果。

matlab 系统求解由于时间原因还未能完成。

4.2 系统平均寿命

5 结论

节点总数为 8 时，系统可靠性最大，为 91.22%。节点总数为 11 时，系统平均工作寿命最大，为 104803.970 hour。在这两种情况下，引起系统失效的主要原因均为阻塞总线，故采用双总线系统作为优化模型。

随着节点总数的提高，双总线系统的优化效果更明显。在双总线系统中，节点总数为 10 时，系统可靠性最大，为 98.86%。节点总数为 15 时，系统平均工作寿命最大，为 147937.232 hour。

6 进一步工作任务

- 在主节点重选时考虑历史遗留问题
- 彻底完成理论计算

7 参考文献

[1] 上海交通大学电子工程系. 工程问题建模与仿真之案例课题 2_V2.12 20220305[EB].

8 附录

8.1 单总线系统的可靠性

Question1.mlx

```
1 clear
2
3 tic
4 %% Initiation
5 S = 100000; % 系统数量
6 w = 30000; % 需要的有效工作时间
7 % perfectly functioning: ns = 0
8 % slave only: ns = 1
9 % disable/master: ns = 2
10 % master only: ns = 3
11 % disable node: ns = 4
12 % failed bus: ns = 5
13 node_state = [0, 3, 1; % 切换器A为状态0
14               1, 5, 1; % 切换器A为状态1
15               2, 3, 4; % 切换器A为状态2
16               4, 4, 4]; % 切换器A为状态3
17 r = zeros(1,20); % 可靠系统数量
18 R = zeros(1,20); % 可靠系统比例
19
20 %% Body
21 for point_num = 3:20 % 节点总数
22     for s = 1:1:S
23         %% Initiation
24         lifeA = exprnd(59000,1,point_num); % 切换器A的使用寿命
25         lifeB = exprnd(220000,1,point_num); % 切换器B的使用寿命
26         stateA = zeros(1,point_num); % 切换器A的状态
27         stateB = zeros(1,point_num); % 切换器B的状态
28         ns = zeros(1,point_num); % 节点的状态
29
30         No_A_B = [(1:point_num),(1:point_num);ones(1,point_num),2*
                    ones(1,point_num);lifeA,lifeB]'; % 第一列是序号，第二列中
                    "1"表示swA，"2"表示swB，第三列是寿命
```

```

31     No_A_B = sortrows(No_A_B,3); % 对切换器按照寿命排序
32     No_A_B = No_A_B'; % 第一行是序号，第二行中"1"表示swA，
        "2"表示swB，第三行是寿命
33     No = No_A_B(1:2,:);
34     life_A_B = No_A_B(3,:);
35
36     %% Simulation
37     for i = 1:2*point_num
38         time = life_A_B(i); % 采用变步长，选取每一个时间
39         if time >= w
40             break
41         end
42         if No(2,i) == 1 % 切换器A发生故障
43             stateA(No(1,i)) = swA;
44         else % 切换器B发生故障
45             stateB(No(1,i)) = swB;
46         end
47         ns(No(1,i)) = node_state(stateA(No(1,i))+1,stateB(No(1,i))
            +1);
48         PF = length(find(ns==0)); % perfectly functioning的节点个
            数
49         SO = length(find(ns==1)); % slave only的节点个数
50         DM = length(find(ns==2)); % disable/master的节点个数
51         MO = length(find(ns==3)); % master only的节点个数
52         FB = length(find(ns==5)); % failed bus的节点个数
53         sys_state = system_state(PF,SO,DM,MO,FB); % 系统的状态
54         if sys_state==1 || sys_state==4
55             break;
56         end
57     end
58
59     %% Count
60     if time >= w

```

```

61         r(point_num) = r(point_num) + 1;
62     end
63 end
64 R(point_num) = r(point_num) / S;
65 end
66
67 %% Display
68 R % 不同节点总数下的系统可靠性
69 [Rw,nice_point_num] = max(R) % 最大的系统可靠性&使系统可靠性最大
    的节点总数
70 toc

```

swA.m

```

1 function [stateA] = swA
2     p = rand(1,1);
3     if p < 0.20
4         stateA = 1;
5     elseif p < 0.35
6         stateA = 2;
7     else
8         stateA = 3;
9 end

```

swB.m

```

1 function [stateB] = swB
2     p = rand(1,1);
3     if p < 0.45
4         stateB = 1;
5     else
6         stateB = 2;
7 end

```

system_state.m

```
1 % 根据各种状态节点数目来判断系统状态1-4
2 function [sys_state] = system_state(PF,SO,DM,MO,FB)
3     if FB==0 && ((MO==1&&PF+SO>=2) || ((MO==0&&PF>=1&&
4         PF+SO>=3)||((MO==0&&PF==0&&DM>=1&&SO>=2))))
5         sys_state = 2; % 状态2: 系统确定能有效工作
6     elseif FB+MO==0 && (PF>=1&&PF+SO==2&&DM>=1)
7         pr = rand(1,1);
8         if pr < DM / (DM+PF)
9             sys_state = 3; % 状态3: 系统恰能有效工作
10        else
11            sys_state = 4; % 状态4: 系统恰不能有效工作
12        end
13    else
14        sys_state = 1; % 状态1: 系统确定不能有效工作
15    end
```

8.2 单总线系统的平均寿命

Question2.mlx

```
1 clear
2
3 tic
4 %% Initiation
5 S = 100000; % 系统数量
6 longest = 200000; % 首次失效时间最大值
7 % perfectly functioning: ns = 0
8 % slave only: ns = 1
9 % disable/master: ns = 2
10 % master only: ns = 3
11 % disable node: ns = 4
12 % failed bus: ns = 5
13 node_state = [0, 3, 1; % 切换器A为状态0
14               1, 5, 1; % 切换器A为状态1
15               2, 3, 4; % 切换器A为状态2
16               4, 4, 4]; % 切换器A为状态3
17 mean_Tf = zeros(1,20); % 系统平均首次失效时间
18
19 %% Body
20 for point_num = 3:20 % 节点总数
21     Tf = zeros(1,S); % 系统首次失效时间
22     for s = 1:1:S
23         %% Initiation
24         lifeA = exprnd(59000,1,point_num); % 切换器A的使用寿命
25         lifeB = exprnd(220000,1,point_num); % 切换器B的使用寿命
26         stateA = zeros(1,point_num); % 切换器A的状态
27         stateB = zeros(1,point_num); % 切换器B的状态
28         ns = zeros(1,point_num); % 节点的状态
29
30         No_A_B = [(1:point_num),(1:point_num);ones(1,point_num),2*
                    ones(1,point_num);lifeA,lifeB]'; % 第一列是序号，第二列中
                    "1"表示swA，"2"表示swB，第三列是寿命
```

```

31     No_A_B = sortrows(No_A_B,3); % 对切换器按照寿命排序
32     No_A_B = No_A_B'; % 第一行是序号，第二行中"1"表示swA，
        "2"表示swB，第三行是寿命
33     No = No_A_B(1:2,:);
34     life_A_B = No_A_B(3,:);
35
36     %% Simulation
37     for i = 1:2*point_num
38         time = life_A_B(i); % 采用变步长，选取每一个时间
39         if time >= longest
40             Tf(s) = longest;
41             break
42         end
43         if No(2,i) == 1 % 切换器A发生故障
44             stateA(No(1,i)) = swA;
45         else % 切换器B发生故障
46             stateB(No(1,i)) = swB;
47         end
48         ns(No(1,i)) = node_state(stateA(No(1,i))+1,stateB(No(1,i))
            +1);
49         PF = length(find(ns==0)); % perfectly functioning的节点个
            数
50         SO = length(find(ns==1)); % slave only的节点个数
51         DM = length(find(ns==2)); % disable/master的节点个数
52         MO = length(find(ns==3)); % master only的节点个数
53         FB = length(find(ns==5)); % failed bus的节点个数
54         sys_state = system_state(PF,SO,DM,MO,FB); % 系统的状态
55         if sys_state==1 || sys_state==4
56             Tf(s) = time;
57             break;
58         end
59     end
60     if i == 2 * point_num + 1

```

```

61         Tf(s) = longest;
62     end
63 end
64     mean_Tf(point_num) = mean(Tf);
65 end
66
67 %% Display
68 mean_Tf % 不同节点总数下的平均工作寿命
69 [Tfw,nice_point_num] = max(mean_Tf) % 最大的平均工作寿命%使平均
    工作寿命最大的节点总数
70 toc

```

8.3 双总线系统

system_state_dual.m

```
1 % 根据各种状态节点数目来判断系统状态1-4
2 function [bug_state] = system_state_dual(PF,SO,DM,MO,FB)
3     master = 0;
4     if MO + DM > 0
5         master = 1;
6     end
7     if PF + SO + master < 3
8         bug_state = 1; % C4引起有效节点数不足3个
9     elseif (MO==0&&FB<=1) && (PF>=1&&PF+SO==2&&DM>=1)
10         pr = rand(1,1);
11         if pr >= DM / (DM+PF)
12             bug_state = 2; % sys4引起有效节点数不足3个
13         else
14             bug_state = 0;
15         end
16     elseif PF + MO + DM == 0
17         bug_state = 3; % 主节点数量不足
18     elseif FB >= 2 || (FB==1&&MO>=2) || (FB==0&&MO>=3)
19         bug_state = 4; % 阻塞总线
20     else
21         bug_state = 0;
22     end
```