

《プログラムのページ》

担当 中西正和

7410 並列システムの信頼度計算

田窪昭夫 (三菱電機(株)計算機製作所)

1. はじめに

M 個の要素が並列接続し、そのうち N 個が“生きている”ことが必須である N out of M タイプの並列システムにおいて、各要素の MTBF が判っているとき、全体の MTBF を求めるアルゴリズムと、それを BASIC 言語で記述したプログラムについて述べる。

2. N out of M 並列システムの信頼度

MTBF, あるいは、その逆数で表現される故障率 λ は時間に無関係に一定であり、信頼度函数 $R(t)$ は次の指数関数で表現されるものとする。

$$R(t) = e^{-\lambda t} \quad (1)$$

(1) 式を逆変換すると、 λ は次式で表現される。

$$\frac{1}{\lambda} = -\int_0^{\infty} R(t) dt \quad (2)$$

N out of M 並列システムの信頼度は M 個のうち、 N 個の“生”と、 $(M-N)$ 個の“死”の“場合”の和で表現される。

$$R(t) = \sum_{N \leq i \leq M} \prod_{i \neq j} R_i(t) Q_j(t) \quad (3)$$

ただし、 $Q(t) = 1 - R(t)$

$$M = 2, 3, 4, \dots; N = 1, 2, \dots, M$$

各要素が同等で、信頼度函数 $R_0(t) = e^{-\lambda_0 t}$ を持つ並列システムでは (3) 式は次のように表現される。

$$R(t) = \sum_{i=N}^M \binom{M}{i} R_0^i(t) (1 - R_0(t))^{M-i} \quad (4)$$

3. アルゴリズム

(4) 式の場合、 λ は λ_0 の定数 (M と N で決定される) 倍で表現されるので、ここでは、(3) 式について、 λ を算出するアルゴリズムを述べる。

M 個の要素の“生死”の“場合”を、 M 桁の 2 進数 I で表現し、桁と要素を対応させ、“0”は“死”を、“1”は“生”を表わすものとする。場合“ I ”の信頼度 R_I は次式で表わされる。

$$R_I = \prod_{k=1}^M R_k \quad (5)$$

ただし、 $R_k = \begin{cases} R_k \cdots I \text{ の桁 } k \text{ が } 1 \text{ の時} \\ 1 - R_k \cdots I \text{ の桁 } k \text{ が } 0 \text{ の時} \end{cases}$

(5) 式の右辺の“結合式”(カッコ表現)を開いて、各項の係数を、 2^M 次のテーブル T に保持することを考える。テーブル T の各エントリは、上の“場合”の表現と同じように、インデックス i (オリジンは 0 とする) を 2 進表現し、“1”の桁の要素の“積”(直列システム)の係数を保持する。 $T(0)$ は定数項に対応し、 $T(0) = 1$ 、 $T(i) = 0$ ($i = 1, 2, \dots, 2^M - 1$) に初期設定しておく。 R_k を掛けることは、各エントリを 2^{i-1} だけ移動することで表わされ、 $1 - R_k$ を掛けることは、各エントリの補数を 2^{i-1} だけ移動したものと、元のエントリの和で表わされる。

テーブル T を完成すると、各エントリに対応する直列システムの信頼度 (または MTBF) に係数を掛けて総和すれば、(5) 式の右辺が求められたことになる。

N out of M 並列システムの信頼度 R は、(5) 式の R_I のうち、 I の“1”の桁が N 個以上のものについての和で表わされる。

$$R = \sum_{(\pm) \geq N} R_I = \sum_{(\pm) \geq N} \prod_{k=1}^M R_k \quad (6)$$

4. BASIC プログラム

上のアルゴリズムを BASIC 言語で記述したプログラムを図-1 に示した。プログラムでは、(5) 式ごとに MTBF を算出するのではなく、(6) 式において MTBF を算出するようになっており、上のテーブル T の他に (6) 式に必要なテーブル (サイズ $2^M - 1$) と各要素の MTBF を保持するテーブルを持っている。

一様並列システム ((4) 式)、直列システムについての算出プログラムにも含まれており、TSS ベースでの使用を前提として、会話モードで任意にシステムを選択でき、種々の形態のシステムについての MTBF が計算されるようになっている。

```

10 DIM A(231),B(209)
20 DIM C(10),E(1023),D(1024)
80 L1=20
81 L2=10
82 J7,J8=0
90 A(1),A(2),A(3)=1
100 FOR M1=2 TO L1
110 FOR N1=0 TO M1
120 K1=M1*(M1+1)/2+N1+1
130 IF N1=0 THEN 170
140 IF N1=M1 THEN 170
150 A(K1)=A(K1-M1)+A(K1-M1-1)
160 GOTO 180
170 A(K1)=1
180 NEXT N1
190 NEXT M1
191 REM
192 REM
200 FOR M2=2 TO L1
210 FOR N2=1 TO M2
220 K2=M2*(M2-1)/2+N2-1
230 B(K2)=0
240 FOR M3=0 TO M2-N2
250 K3=0
260 FOR N3=M3 TO M2-N2
270 K4=1-2*(M3+N3-2*INT((M3+N3)/2))
280 K5=M2*(M2+1)/2+M2-N3+1
290 K6=N3*(N3+1)/2+M3+1
300 K3=K3+K4*A(K5)*A(K6)
310 NEXT N3
320 B(K2)=B(K2)+K3/(M2-M3)
330 NEXT M3
340 NEXT N2
350 NEXT M2
351 REM
352 REM
380 PRINT
390 INPUT J1
400 IF J1<-2 THEN 1400
410 IF J1=-2 THEN 700
420 IF J1=-1 THEN 550
430 IF J1=0 THEN 500
440 IF S1=0 THEN 480
450 S1=S1*J1/(S1+J1)
460 PRINT ,S1
470 GOTO 390
480 S1=J1
490 GOTO 390
500 PRINT
510 PRINT 'SERIES'
520 PRINT
530 S1=0
540 GOTO 390
541 REM
550 PRINT
560 PRINT 'PARALLEL-1'
570 PRINT
580 INPUT J2
590 IF J2<-2 THEN 1400
600 IF J2=-2 THEN 700
610 IF J2=-1 THEN 580
620 IF J2=0 THEN 500
621 INPUT J3,J4
630 IF J2>J3 THEN 580
640 IF J3>L1 THEN 580
650 IF J3<2 THEN 580
660 IF J4<=0 THEN 580
670 S1=J4*B(J3*(J3-1)/2+J2-1)
680 PRINT ,S1
690 GOTO 580
691 REM
700 PRINT
710 PRINT 'PARALLEL-2'
720 PRINT
730 INPUT J5
740 IF J5<-2 THEN 1400
750 IF J5=-2 THEN 730
760 IF J5=-1 THEN 550
770 IF J5=0 THEN 500
771 INPUT J6
780 IF J5>J6 THEN 730
790 IF J6>L2 THEN 730
800 IF J6<2 THEN 730
810 FOR M4=1 TO L2
820 C(M4)=0
830 NEXT M4
840 FOR N4=1 TO J6
850 INPUT C(N4)
855 IF C(N4)<=0 THEN 850
860 NEXT N4
861 IF J5<>J7 THEN 863
862 IF J6=J8 THEN 1210
863 J7=J5
864 J8=J6
870 D(1)=1
880 FOR M5=1 TO 2**L2-1
890 D(M5+1),E(M5)=0
900 NEXT M5
910 I1=1
920 D(1)=1
921 FOR N5=1 TO 2**L2-1
922 D(N5+1)=0
923 NEXT N5
925 X1=I1*2
930 X2=0
940 X1=INT(X1/2)
950 IF X1=0 THEN 1010
960 X3=X1-2*INT(X1/2)
970 IF X3=0 THEN 940
980 X2=X2+1
990 IF X2>=J5 THEN 1040
1000 GOTO 940
1010 I1=I1+1
1020 IF I1>=2**J6 THEN 1210
1030 GOTO 920
1040 I2=0
1050 I3=0
1060 X4=INT(13/2**I2)
1070 X5=X4-2*INT(X4/2)
1080 IF X5=1 THEN 1130
1090 X6=INT(I1/2**I2)
1100 X7=X6-2*INT(X6/2)
1110 D(I3+1+2**I2)=D(I3+1)*(2**X7-1)
1120 D(I3+1)=D(I3+1)*(1-X7)
1130 I3=I3+1
1140 IF I3<2**J6 THEN 1060
1150 I2=I2+1
1160 IF I2<J6 THEN 1050
1170 FOR I4=1 TO 2**J6-1
1180 E(I4)=E(I4)+D(I4+1)
1190 NEXT I4
1200 GOTO 1010
1201 REM
1210 S1=0
1220 I5=1
1230 IF E(I5)=0 THEN 1360
1240 S2=0
1250 I6=0
1260 Y1=INT(I5/2**I6)
1270 Y2=Y1-2*INT(Y1/2)
1280 IF Y2=0 THEN 1330
1290 IF S2=0 THEN 1320
1300 S2=S2*C(I6+1)/(S2+C(I6+1))
1310 GOTO 1330
1320 S2=C(I6+1)
1330 I6=I6+1
1340 IF I6<6 THEN 1260

```

1350	S1=S1+S2*E(I5)		73,3,3000,3000,3000
1360	I5=I5+1		1000
1370	IF I5<2**J6 THEN 1230		
1380	PRINT ,S1		7-1
1390	GOTO 730		PARALLEL-1
1400	STOP		73,3,3000
1410	END		1000.00
>RUN			72,3,4000
14:01	04/10		3333.33
70			7-2
SERIES			PARALLEL-2
71000			72,3,4000,4000,4000
72000			3333.33
72000	666.667		7-3
7-1	500.000		1400 HALT
PARALLEL-1			>
71,2,2000			
72,2,2000	3000		
7-2	1000		
PARALLEL-2			
71,3,1000,2000,3000			
	3928.79		
71,3,3000,2000,1000			
	3928.79		
72,3,1000,2000,3000			
	1525.76		
72,3,3000,1000,2000			
	1525.76		

5. おわりに

上のような組み合わせ論的なアルゴリズムにおいてはバッファ・サイズとか、プログラム・ループは、指数関数的に増加する性質をもっており、この点、図-1のプログラム* は最適とは言い難い。BASIC 言語の本来の目的（初心者向けの言語）を考慮して、プログラムの流れの簡潔性に主眼をおいて作成している。

* 計算機 MELCOM 7700 を使用したプログラム例であり、この BASIC 言語では、“LET” は省略可能である。

(昭和49年4月30日受付)