A Highly Efficient Redundancy Scheme: Self-Purging Redundancy

by

Jacques Losq

July 1975

Technical Report No. 62

This work was supported by National Science Foundation Grant GJ-40286 with partial salary from l'Institut de Recherche en Informatique et Automatique (IRIA, France).

DIGITAL SYSTEMS LABORATORY

STANFORD ELECTRONICS LABORATORIES

STANFORD UNIVERSITY • STANFORD, CALIFORNIA

A HIGHLY EFFICIENT REDUNDANCY SCHEME: SELF-PURGING REDUNDANCY

by

Jacques Losq

July 1975

Technical Report No. 62

DIGITAL SYSTEMS LABORATORY
Department of Electrical Engineering   Department of Computer Science
Stanford University
Stanford, California

This work was supported by National Science Foundation Grant GJ-40286 With partial salary from l'Institut de Recherche en Informatique et Automatique (IRIA, France).

Paper submitted to a special issue on fault-tolerant computing of IEEE Transactions on Computers.

Digital Systems Laboratory
Department of Electrical Engineering   Department of Computer Science
Stanford University
Stanford, California

Technical Report No. 62

July 1975

A HIGHLY EFFICIENT REDUNDANCY SCHEME: SELF-PURGING REDUNDANCY

Jacques Losq

ABSTRACT

The goals of this paper are to present an efficient redundancy scheme for highly reliable systems, to give a method to compute the exact relia-, bility of such schemes and to compare this scheme with other redundancy schemes. This redundancy scheme is self-purging redundancy; a scheme that **uses** a threshold voter and that purges the failed modules. Switches for self-purging systems are extremely simple: there is no replacement of failed modules and module purging is quite simply implemented. Because of switch simplicity, exact reliability calculations are possible. The effects of switch reliability are quantitatively examined. For short mission times, switch reliability is the most important factor: self-purging systems have a probability of failure several times larger than the figure obtained when switches are assumed to be perfect. The influence of the relative frequency of the diverse types of failures (permanent, intermittent, stuck-at,...) are also investigated. Reliability functions, mission time improvements and switch efficiency are displayed. Self-purging systems are compared with other redundant systems, like hybrid or NMR, for their relative merits in reliability gain, simplicity, cost and confidence in the reliability estimation. The high confidence in the reliability evaluation of self-purging systems makes them a standard for the validation of several models that have been proposed to take into account switch reliability. The accuracy of models using coverage factors can be evaluated that way.

Index terms: Reliability, self-purging redundancy, switch, mission time, Poisson distribution, threshold gates, convolutions, dormancy factors, coverage factors.

I INTRODUCTION

The need for ultra-reliable computers has been increasing very rapidly since the introduction of computers in areas where a malfunction can lead to a catastrophe .

There are many ways to improve the reliability of systems . One way is to use Stand-by Redundancy, [1, 3], in which, as soon as a fault is detected, the faulty module is switched off and replaced by a spare performing the same logical function . Another method is to use Massive Redundancy, [4, 6], in which replication of modules allows an instantaneous and automatic masking of the faults that occur. . Combining these two methods, there are various solutions that are included under the title of Hybrid Redundancy, [7, 9] (Fig. 1).

Hybrid redundant systems present several advantages over both massive and stand-by redundant systems, but they require a fairly complicated switching **mechanism,** [10, 11]. . Complicated switching mechanisms introduce additional causes of system failure . Furthermore, very accurate modeling of hybrid systems, that takes into account the reliability of the switching **mechanisms,is** extremely complex . Another redundancy scheme, Self-purging Redundanc[12, 133 has many of the advantages of hybrid redundancy and few of the disadvantages . Self-purging systems have very simple switching mechanisms, straightforward design and they are simpler, cheaper and more reliable than hybrid systems . Computation of the exact reliability is possible for self-purging systems .

This paper focuses on the determination of the exact reliability expression for self-purging systems . The influence of the reliability of switching mechanisms on the overall reliability will be determined . The evolution of reliability as a function of the mission time will be obtained . Simple bounds for reliability will be derived . The results given by various approximative models applied to self-purging systems will be compared to the exact reliability . This will characterize the degree of confidence that can be granted to each model .

2 SELF-PURGING REDUNDANCY

2-A Description

Single output self-purging systems are formed by a set of P modules, a disagreement detector, a switch and a threshold voter , There is no differentiation between spare modules and core modules as in hybrid systems . Each module takes part in the vote taking proccess as long as it is fault-free . When it fails, it is disconnected from the voter , The voter is a threshold gate with threshold of M and weight of one for each input (Fig. 2) . All the modules are initially fault-free . They all send a 1 (or 0) to the voter which responds by sending a 1 (or 0) on its output . When an error occurs at the output of one module, this error is easily detected by comparison of the module output with the voter output . Failed modules are forced to send a 0 on their output . This is logically equivalent to disconnecting failed modules from the voter . The voter output is one if and only if the weighted sum of its inputs is equal or greater to its threshold . So, a 0 on one of the voter input does not influence the voter output, as long as M inputs are correct .

Self-purging systems with threshold of M operate properly (assuming perfect switch) as long as there are M or more fault-free modules . The interest of this redundancy scheme is that the information needed to switch-off a module depends only on the state of this module . The state of a module is easily determined by comparison of its output to the voter output . Another interest of self-purging systems is that modules are never switched-on during system use .

Self-purging systems with multiple outputs (n outputs) can be one of two types . Type I systems have only one disagreement detector while type II systems have n disagreement detectors (one for each output) . Type II systems are more reliable because they fail only when corresponding outputs of P-M+1 modules have been subject to an error, while type I systems fail when P-M+1 modules have been subject to an error .

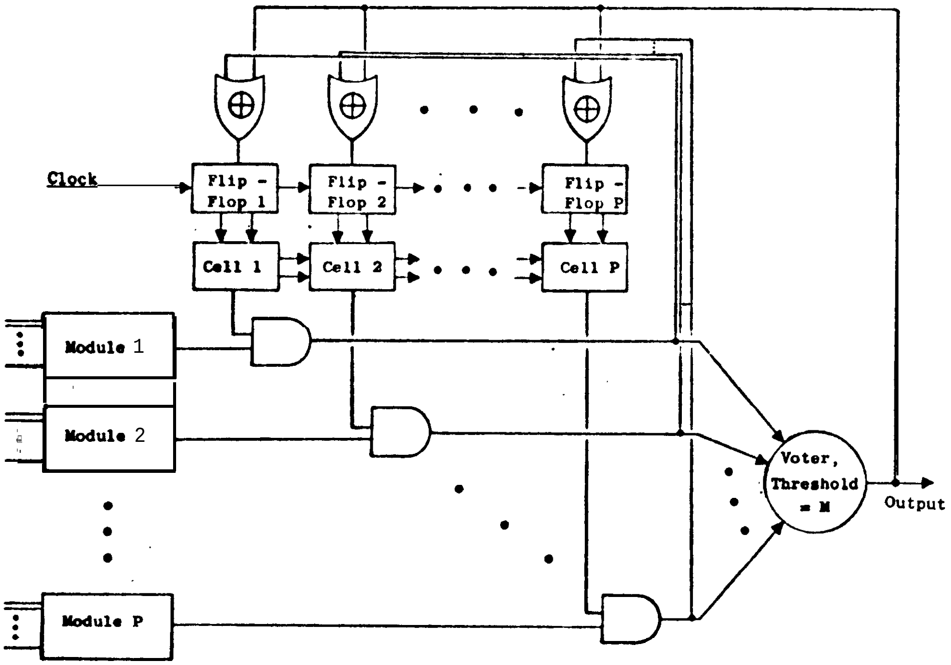


Fig. 1 . Hybrid systems with iterative cell array switch .

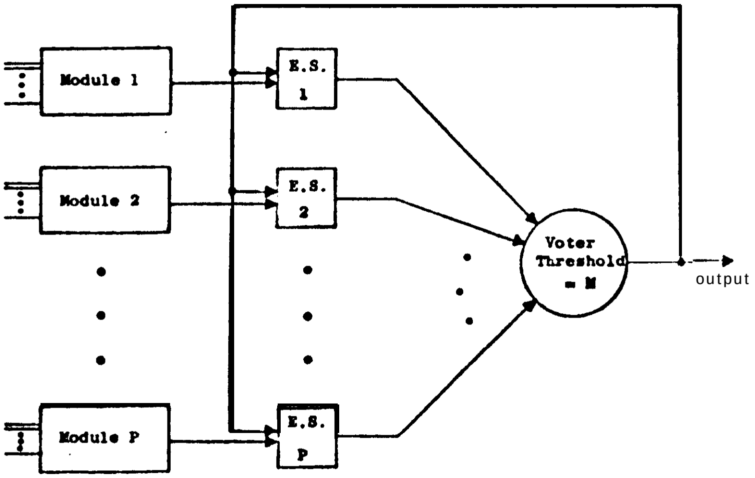


Fig. 2 . Self-purging systems .

2-B Switch for self-purging systems

Because the information needed to switch-off a module depends only on the state of this module, switches for self-purging systems with P modules can be decomposed into P elementary switches, one for each module , Furthermore, elementary switches are very simple . They need only to detect the first disagreement between the module output and the voter output, to keep this state (has disagreed or not) and to logically force the output of the faulty-modules to 0 . This can be realized with an EXCLUSIVE OR gate, a flip-flop and an AND gate (Fig. 3) .

Another advantage of self-purging systems results from the fact that switches can be divided into elementary switches . It is possible to include the elementary with with the module to form a modified module (Fig. 4) . Furthermore, because switches are only a method to force the output of failed modules to a logical zero, there is no need for switches if the modules always fail to stuck-at-zero (or to stuck-at-one if the threshold is changed from M to P-M+1) .

Self-purging systems can benefit from the use of fail-safe logic, 14, 15 . When ultra-high reliability is required, most of the system failures will be caused by the unreliability of the switching mechanism . So, duplicating each elementary switch such that each module feed two inputs of the voter (for which the threshold is doubled), decreases the system probability of failure . For example, if q is the probability that an elementary switch failure results in a permanent logical one on the corresponding voter input, the probability of system failure due to this kind of switch failure is :

$$f_1 = \binom{P}{M} \cdot q^{M} \cdot (1-q)^{P-M}$$

If every elementary switch is duplicated, the probability of this kind of failure is :

$$f_2 = \binom{2P}{2M} \cdot q^{2M} \cdot (1-q)^{2.(P-M)} \simeq \frac{4^{M}}{\binom{2.M}{M}} \cdot \mathrm{fl}^{2}$$

However, this increases the complexity of the voter (by doubling the number of its inputs) . This method is especially interesting when the threshold voter is realized by a ROM chip large enough to allow for twice as many inputs .

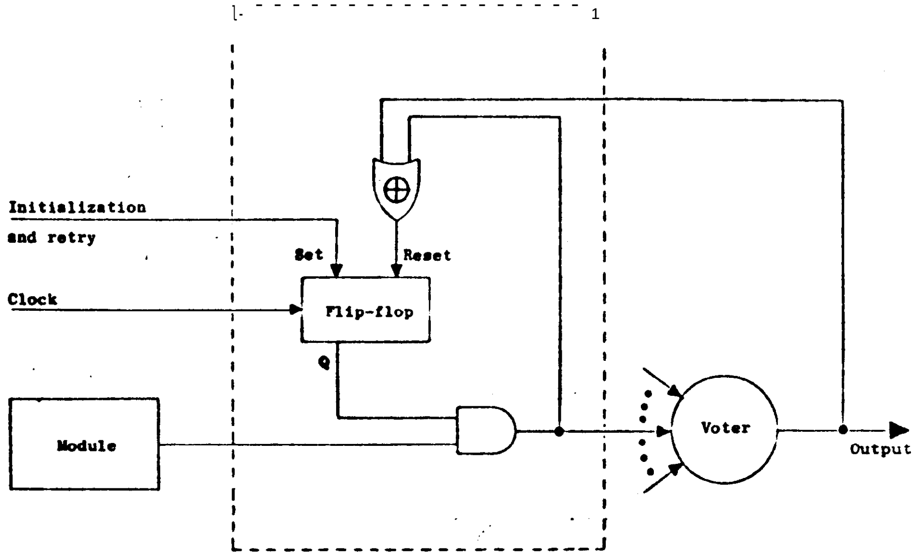


Fig. 3 . Elementary switches for self-purging systems .

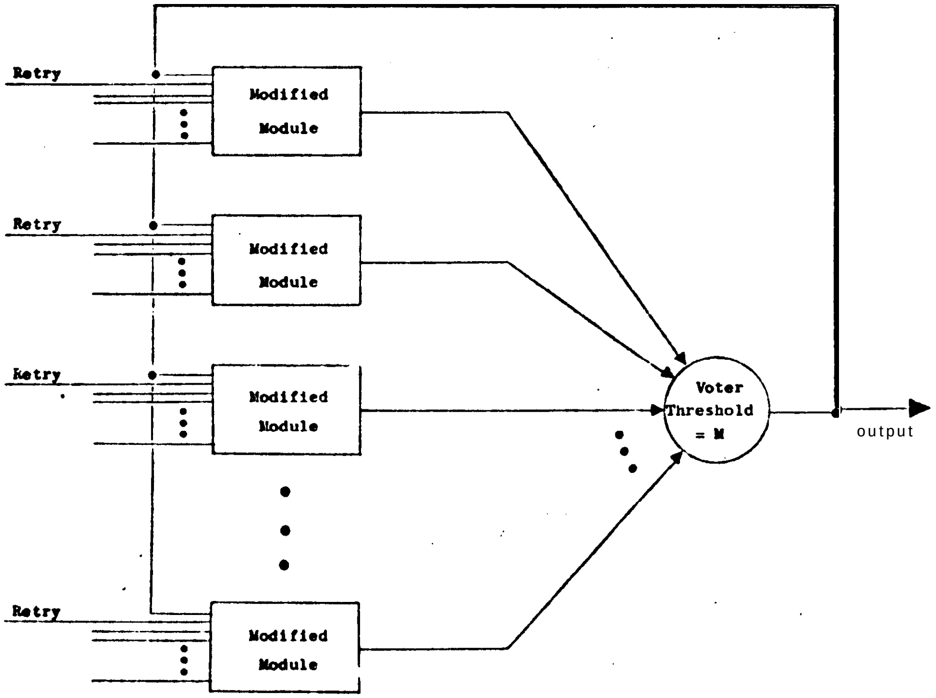


Fig. 4 . General representation of self-purging systems with modified modules .

2-C Retry

Many of the failures that occur in electronic components are intermittent failures . If modules in self-purging systems are declared faulty at the first disagreement, intermittent failures will result, as hard failures, in module removals . This decreases significantly the gain that can be obtained using redundancy . Retry procedures are used to avoid that an intermittent failure results in a module removal . Retry procedures are very simply achieved for self-purging systems . In order to retry a module, it is sufficient to reset its elementary switch to the state "has not yet failed" (using the asynchronous input of the flip-flop) . However, it is necessary to retry a module when there are still enough fault-free modules . The easiest way is to retry a module as soon as it disagrees with the voter output for the first time .

A small counter can be included in each elementary switch to implement an automatic retry procedure . A module will be declared faulty only if it disagrees with the voter output more than once during x clock cycles . Such switches make certain that transient failures do not result in a module removal . The choice of x, the counter period, depends on the practical characteristics of the electronic components used, on the ratio of module mean-life to clock cycle and on the price that one wishes to pay for such a protection against intermittent failures . However, the counter period should not be too small, otherwise a hard failure that produces output error for only some of the input combinations may be mistaken for a transient failure .

Self-purging systems are very interesting to use when module repair exists . The failed modules can be physically disconnected from their elementary switches (or even from the voter if an open circuit corresponds to a logical zero) . They can be repaired and then connected back to their elementary switches without system interruption . The only thing to do is to send a signal on the corresponding retry input . It is also possible to repair the elementary switches without system interruption, which is not possible with hybrid systems .

3 RELIABILITY OF SELF-PURGING SYSTEMS

3-A Reliabilitv assuming perfect switches

The reliability of self-purging systems with P modules and a voter with threshold of M will be denoted by $R_{P,0,M}(T)$. In general, $R_{a,b,c}(T)$ will denote the reliability of a redundant system with a powered modules (a modules in core), b spares and a voter with threshold of c . T represents time .

Self-purging systems with perfect voter, perfect switch and a voter threshold of M will perform correctly as long as they are M, or more, fault-free modules . If R(T) represents the reliability,at time T,of a module, the general reliability of self-purging systems with perfect switch and voter is :

$$R_{P,0,M}(T) = \sum_{i=M}^{P} \binom{P}{i} \left[R(T)\right]^{i} \cdot \left[1-R(T)\right]^{P-i}$$

$$= 1 - \sum_{j=0}^{M-1} \binom{P}{j} \left[R(T)\right]^{j} \cdot \left[1-R(T)\right]^{P-j}$$

For the most common case of self-purging systems with threshold of 2 (corresponding to hybrid systems with TMR core), the reliability is :

$$R_{P,0,2}(T) = 1 - \left[1-R(T)\right]^{P} - P.R(T).\left[1-R(T)\right]^{P-1}$$

This reliability is equal to the reliability of hybrid systems with P modules, TMR core, perfect switch and perfect voter . In general, self-purging systems with P modules and a threshold of M are equivalent to hybrid systems with P modules and NMR core (N=2.M-1) if switches and voters are perfect .

So, self-purging systems are as general as hybrid systems . Furthermore, switches for self-purging systems are simpler (two gates plus one flip-flop against seven gates plus one flip-flop for hybrid systems, [10], [11]) . Switches for hybrid systems will be less reliable, for the same complexity, than switches for self-purging systems because they are realized as iterative arrays which allow for the propagation of errors from one cell to the next one . So, self-purging systems are more reliable than hybrid systems . Exact reliability for self-purging systems will provide upper bound for hybrid systems .

3-B Exact reliability expression

The exact reliability for self-purging systems can not be obtained with combinatorial methods alone, as it is the case for massive redundant systems, [1] . For example, consider a self-purging system with 5 modules and a threshold of 2 . Assume that the system is in the following state at time T : one module and its elementary switch (module A and switch S_A) have failed to stuck-at-one, another module (module B) is stuck-at-one but its switch is fault-free, everything else is correct . Given this description, it is impossible to determine whether or not the system is working properly . The system is working correctly if the failure in module B occured before the failure in either module A or its switch S_A . But if the failure in module B occured after the failure in A and S_A, the system output is a constant one ; upon occurence of the failure in module B, two of the five voter inputs are one . The voter output will be one and the three fault-free modules will be declared faulty when, in fact, they are the only fault-free ones . This example shows that a study of the evolution in time is necessary .

As it can be seen'from this example, the problem exists because switches are fed by the voter output and not by the correct module output . The effective threshold will denote the real threshold of the voter minus the number of voter inputs that are stuck-at-one (because of switch failures) . Incorrect diagnosis by the switches happens only upon occurence of stuck-at-one module errors when the effective threshold has been reduced to one .

If one assumes that self-purging systems fail as soon as the effective threshold is reduced to one, one gets a lower bound for the reliability . If one assumes that diagnosis is possoble as long as the effective threshold is one, one gets an upper bound , Both these bounds can be obtained by combinational methods . The exact reliability is closer to the upper bound than to the lower bound because, when the effective threshold is reduced to one, correct detection and diagnosis of module stuck-at-zero failures are still possible .

3-B-l State of modified modules

Any of the lines O_i and Y_i and any of the control AND gates A_i (Fig. 5) can be either fault-free or faulty . When one of these lines (gates) is faulty, it may produce either an erroneous one or an erroneous zero depending on the type of failure and on the inputs . If, for a given fault, there exist some inputs for which the line (gate) produces an erroneous one, the line (gate) will be called stuck-at-one . In the other cases, the line (gate) will be called stuck-at-zero . So, the state of a line (gate) is one of the elements of the set $\{g, s.a.0, s.a.1\}$. The letter g means that the line (gate) is fault-free .

The effects of transient failures inside modules are normally cancelled if elementary switches are provided with an automatic retry procedure . So, only permanent failures'need to be taken into account . However, it is important to note that, once the effective threshold is reduced to one, transient erroneous ones result in a system error . For permanent faults, any fault that can produce an erroneous one will be called a stuck-at-one fault, even though it may as well be a bridging fault as a stuck-at fault .

The state of the-modified module i is completely specified by the triple $\langle x_o, x_y, x_a \rangle$, where x_o, x_y and x_a are respectively the state of the line O_i, the line Y_i and the AND gate A_i . Because each module, switch and AND gate can be in one of three states, each modified module can be in one of twenty seven states . However, it is possible to reduce the number of states for modified modules from twenty seven to seven . These seven states are :

g : the module, elementary switch and control AND gate are fault-free,

s.a.g : the module and control AND gate are fault-free but the elementary switch is stuck-at-one,

f.0 : the module is stuck-at-one but the switch and control AND gate are fault-free (forcing the line X to zero),

s.a.0 : the module or the elementary switch is stuck-at-zero but the control AND gate is fault-free,

s.a.1 : both the module and elementary switch are stuck-at-one but the control AND gate is fault-free,

a.0 : the control AND gate is stuck-at-zero, whatever the states of the module and elementary switch are,

a.1 : the control AND gate is stuck-at-one, **whatever** the states of the module and elementary switch are .

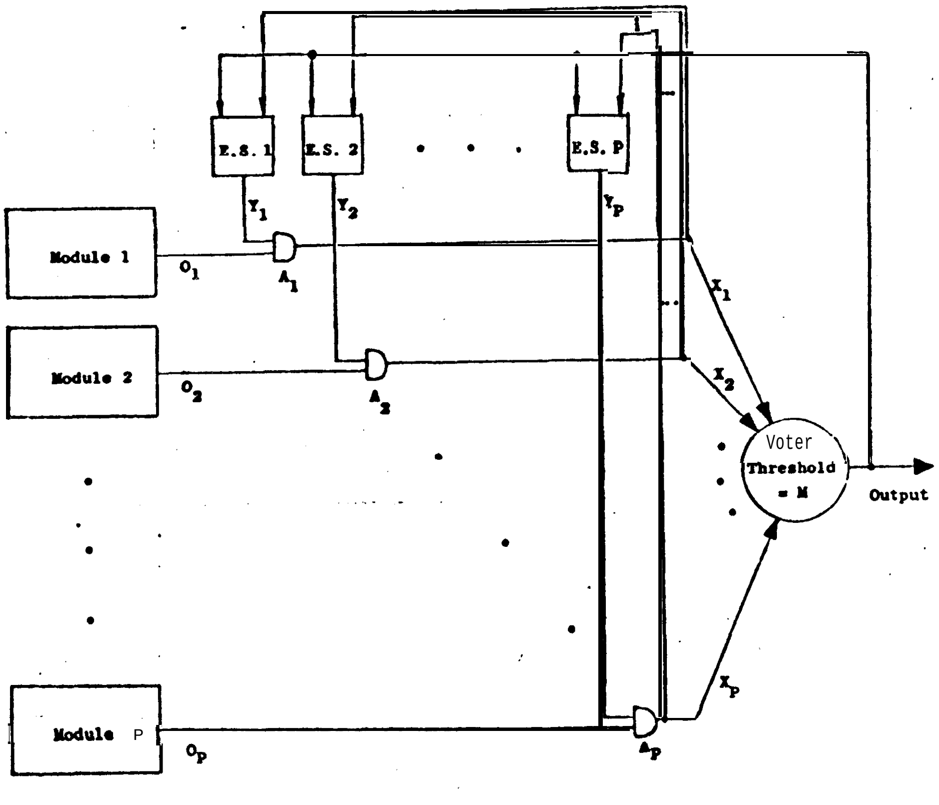


Fig. 5 . Self-purging systems .

These seven states are sufficient for the description of modified modules (Table 1) . If the control AND gates are assumed to be perfect, the last two states can be disregarded, leading to five states . The state of a modified module as a function of the state of the module, switch and control AND gate is given in Table 1 .

Assuming a Poisson distribution of failures, the transition probability matrix, M, is given in Table 2 . The vector giving the probability, at time T, for the states of the modified modules is :

$$\begin{bmatrix} P_{g} \\ P_{s.a.g} \\ P_{f.0} \\ P_{s.a.0} \\ P_{s.a.1} \\ P_{a.0} \\ P_{a.1} \end{bmatrix} = \left[\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} . . \underset{dT \to 0}{\text{Limit}} \begin{bmatrix} M^{\frac{T}{dT}} \end{bmatrix} \right]^{t} = \begin{bmatrix} \exp(-(\lambda+\mu+\nu).T) \\ \exp(-(\lambda+\nu).T).\left(1-\exp(-\mu.T)\right).X_s \\ \exp(-(\mu+\nu).T).\left[1-\exp(-\lambda.T)\right].X_m \\ \exp(-\nu.T).(1-((1-X_m)\exp(-\lambda.T)+X_m) \\ \qquad .((1-X_s)\exp(-\mu.T)+X_s)) \\ \exp(-\nu.T)(1-\exp(-\lambda.T))(1-\exp(-\mu.T)).X_m.X_s \\ (1-X_a).(1-\exp(-\nu.T)) \\ X_a.(1-\exp(-\nu.T)) \end{bmatrix}$$

λ, μ and v are respectively the module, elementary switch and control AND gate failure rate . X_m, X_s and X_a are the ratio of the stuck-at-one failures to the total number of failures .

3-B-2 Probability of fault-free operation of self-purging systems with effective threshold larger than one

When all the modules are identical, the states of self-purging systems can be fully characterized by the number of modified modules in each of the seven possible states . If there is no interdependence between module reliabilities, occurence of multiple failures is extremely unlikely . So, when the effective threshold is greater than one, the fault-detection and diagnosis performed by the switches is perfect .

Self-purging systems with effective threshold of M' perform correctly if, and only if,the voter is fault-free and the number of modified modules in either the state g or the state s.a.g is greater or equal to M' . The voter reliability limits severely the overall reliability . However, it is worth

Switch / Module	Fault-free AND gate			Stuck-at-zero AND gate			Stuck-at-one AND gate		
	g	s.a.0	s.a.1	g	s.a.0	s.a.1	g	s.a.0	s.a.1
g	g	s.a.0	s.a.g	a.0	a.0	a.0	a.1	a.1	a.1
s.a.0	s.a.0	s.a.0	s.a.0	a.0	a.0	a.0	a.1	a.1	a.1
s.a.1	f.s.a.0		s.a.1	a.0	a.0	a.0	a.1	a.1	a.1

Table 1 . State of modified modules as function of the states of modules, elementary switches and control AND gates .

Next state / Present state	g	s.a.g	f.0	s.a.0	s.a.1	a.0	a.1
g	$1-dT.(\lambda+\mu+\nu)$	$\mu.X_s.dT$	$\lambda.X_m.dT$	$[(1-X_m).\lambda+(1-X_s).\mu].dT$	0	$(1-X_a).\nu.dT$	$X_a.\nu.dT$
s.a.g	0	$1-dT.(\lambda+\nu)$	0	$(1-X_m)\lambda.dT$	$X_m.\lambda.dT$	$(1-X_a).\nu.dT$	$X_a.\nu.dT$
f.0	0	0	$1-dT.(\mu+\nu)$	$(1-X_s)\mu.dT$	$X_s.\mu.dT$	$(1-X_a).\nu.dT$	$X_a.\nu.dT$
s.a.0	0	0	0	$1-\nu.dT$	, 0	$(1-X_a)\nu.dT$	$X_a.\nu.dT$
s.a.1	0	0	0	0	$1-\nu.dT$	$(1-X_a).\nu.dT$	$X_a.\nu.dT$
a.0	0	0	0	0	0	1	0
a . 1	0	0	0	0	0	0	1

Table 2 . Transition matrix, M, for modified modules .

noting that not every voter failure causes a system failure , Failures on the voter inputs are equivalent to modified module failures . The voter reliability can be increased by the use of TMR on the voter . Effective thresholds are equal to real threshold M minus the number of modified modules in states s.a.1 or a.1 . So, self-purging systems with P modules and a threshold M perform correctly if the number of modified modules in states f.0, s.a.0 or a.0 is less or equal to P-M .

If $P_{a,b,c,d,e,f,g}$ represents the state of self-purging systems with respectively a,b,c,d,e,f and g modified modules in states g, s.a.g, f.0, s.a.0, s.a.1, a.0 and a.1 , the probability that self-purging systems perform correctly with a effective threshold larger than one is :

$$R' = \sum_{\substack{e+g<M-1 \\ c+d+f \leqslant P-M}} \text{Probability of the state } P_{a,b,c,d,e,f,g}$$

$$\text{Probability of the state } P_{a,b,c,d,e,f,g} = \binom{P}{a,b,c,d,e,f,g} \cdot P_g^{\,a} \cdot P_{s.a.g}^{\,b}$$

$$\cdot\, P_{f.0}^{\,c} \cdot P_{s\,a.0}^{\,d} \cdot P_{s\,a.1}^{\,e} \cdot P_{a\,0}^{\,f} \cdot P_{a\;1}^{\,g} \;.$$

R' is the lower bound of the reliability . The upper bound, R_u, can be obtained similarly if it is assumed that fault-detection and diagnosis work perfectly even when the effective threshold is reduced to one . This is an upper bound because, when the effective threshold is one, the switch can not correctly diagnose a stuck-at-one failure . The upper bound R_u is :

$$R_u = \sum_{\substack{e+g<M \\ c+d+f<P-M}} \text{Probability of the state } P_{a,b,c,d,e,f,g}$$

These bounds are very tight . Their difference is proportional to the probability that M-l modified modules be stuck-at-one, which is equal to the probability that M-l modules and their corresponding elementary switches be stuck-at-one . Both these bounds can be obtained by combinational methods, simply implemented on computer ,

3-B-3 Probability of correct operation with an effective threshold of one

When the effective threshold is reduced to one, there is no possibility to detect and diagnose the occurence of module stuck-at-one failures . So, given a self-purging system initially in a state $P_{a,b,c,d,e,f,g}$ with an effective threshold of one (e+g=M-1), the probability that it operates correctly for a time period u is :

$R''_{a,b,c,d,e,f,g}(u)$ = Probability that no stuck-at-one failure occurs and that the number of modified modules giving a correctoutput, a+b, is equal to, or larger than, the effective threshold .

However, effective thresholds are not always step decreasing functions of time . For example, the effective threshold increases if a stuck-at-zero failure occurs in the control AND gate of a modified module previously in a state s.a.1 . The term $R''_{a,b,c,d,e,f,g}$ can be expressed as the probability of correct operation with a effective threshold of one plus a convolution term giving the probability that the effective threshold increases and that the resulting system survives until time u . The term giving the probability of correct operation with an effective threshold of one can be obtained simply . The convolution term is more complex . It may require up to 2(P-M)+1 integrations . One integration is required each time the effective threshold changes and it can change up to 2(P-M)+1 times .

If the control AND gates are assumed to be perfect, effective threshold are decreasing function of time . The number of modified modules in states a.0 and a.1 is zero . The probability that self-purging systems in state $P_{a,b,c,d,1,0,0}$ survives for a time period u is :

$R''_{a,b,c,d,1,0,0}(u) = R''_{a,b,c,d,1}(u)$ = Prob(no stuck-at-one failure occurs during time interval u) . Prob(at time u,there is at least one modified module in state g or s.a.g. no stuck-at-one failure has occured)

These terms can be easily obtained from the matrix $\text{Limit}_{dT\to 0}\left[M^{\frac{T}{dT}}\right]$ (Table 3) .

The probability that the effective threshold is reduced from 2 to 1, at time T, are obtained directly from the matrix M .

Final state Initial state	g	s.a.g	f.0	8.8.0	s.a.1	a [redacted]	[redacted] .1
g	$e^{-(\lambda+\mu+\nu).T}$	$x_s.e^{-(\lambda+\nu).T}$ $.(1-e^{-\mu.T})$	$x_m.e^{-(\mu+\nu).T}$ $.(1-e^{-\lambda.T})$	$e^{-\nu.T}.[1-$ $((1-x_m)e^{-\lambda T}+x_m$ $((1-x_a)e^{-\mu T}+x_s)]$	$e^{-\nu.T}.$ $x_m.(1-e^{-\lambda.T}).$ $x_s.(1-e^{-\mu.T})$	$1-x_a).(1-e$ [redacted] $^{\nu.T})$	$x_a.(1-e^{\nu.T})$
s.a.g	0	$e^{-(\lambda+\nu).T}$	0	$(1-x_m).e^{-\nu.T}$ $.(1-e^{-\lambda.T})$	$x_m.e^{-\nu.T}$ $.(1-e^{-\lambda.T})$	$1-x_a).(1-e^{-\nu.T})$	$x_a.(1-e^{-\nu.T})$ I
1.0	0	0	$e^{-(\mu+\nu).T}$	$(1-x_s).e^{-\nu.T}$ $.(1-e^{-\mu.T})$	$x_s.e^{-\nu.T}$ $.(1-e^{-\mu.T})$	$1-x_a).(1-e^{-\nu.T})$	$x_a.(1-e^{-\nu.T})$
s.a.0	0	0	0	$e^{-\nu.T}$	0	$1-x_a).(1-e^{\nu.T})$	$x_a.(1-e^{\nu.T})$
• .0.1	0	0	0	0	$e^{-\nu.T}$	$1-x_a).(1-e^{-\nu.T})$	$x_a.(1-e^{-\nu.T})$
a.0	0	0	0	0	0	1	0
a.1	0	0	0	0	0	0	1

Table **3** . Limit of the matrix $M^{\frac{T}{dT}}$ as **dT** goes to zero .

3-B-4 Reliability expression

The general expression for self-purging system **reliability** regroups the probability of correct operation with and effective threshold larger than **one,(term R'),** plus a convolution term giving the probability that the effective threshold be reduced to one and that the resulting system survives until the end of mission time . For self-purging systems with perfect control AND gates, the reliability, R, can be expressed as :

$$R(T) = R'(T) + \sum \int_0^T b.P_{a,b,c,d,2}(\tau).M\left[2,5\right].R''_{a,b-1,c,d,1}(T-\tau).d\tau$$

$$+ \sum \int_0^T c.P_{a,b,c,d,2}(\tau).M\left[3,5\right].R''_{a,b,c-1,d,1}(T-\tau).d\tau$$

The summations are taken on **the** set of all valid system states with effective threshold of two . $M\left[2,5\right]$ and $M\left[3,5\right]$ represent respectively the element of the second row, fifth column and the element of the third row, fifth column of the matrix M .

4 PERFORMANCES OF SELF-PURGING SYSTEMS

4-A Reliabilitv and mission time

A computer program was used to obtain the bounds and the exact figure for self-purging system reliability . All the modules were assumed to have the same failure rate (no interdependence between the modules) . Module complexity is expressed by the number of gates inside each module . The probabilities of stuck-at-one and stuck-at-zero failures at the output of the elementary switches were computed from the failure rates of their gates and flip-flops . Flip-flops were assumed to have the reliability of ten gates . When mission times are used in curves, the unity of time is the mean-life of the modules .

Fig. 6 shows the upper and lower bounds for the probability of failure of three self-purging systems . As it can be expected, the longer the mission time, the tighter the bounds . These bounds are quite useful when self-purging systems are designed for mission times of the same order as the module mean-life : the time-consuming computations for the exact reliability are not needed . Fig. 7 gives the reliability curves for some self-purging systems . For comparison, module reliability has also been plotted . For short mission times, the curves displaying the probability of failure are more meaningful (Fig. 8) .

The mission time improvement obtained by using self-purging redundancy is plotted in Fig. 9 as a function of the reliability at the end of the mission .

4-B Effects of module failure mode

Self-purging system reliability depends on the failure mode of the modules . It was already mentioned that no switch is needed when failures never cause module outputs to give erroneous ones . Self-purging system reliability is sensitive to the relative frequency of stuck-at-one faults (any permanent fault that can produce an erroneous one) to stuck-at-zero faults (permanent faults that produce only erroneous zero) . This sensitivity can be seen in Fig. 10 which shows the relative difference in probability of failure between real self-purging systems and self-purging systems with perfect switching mechanisms .

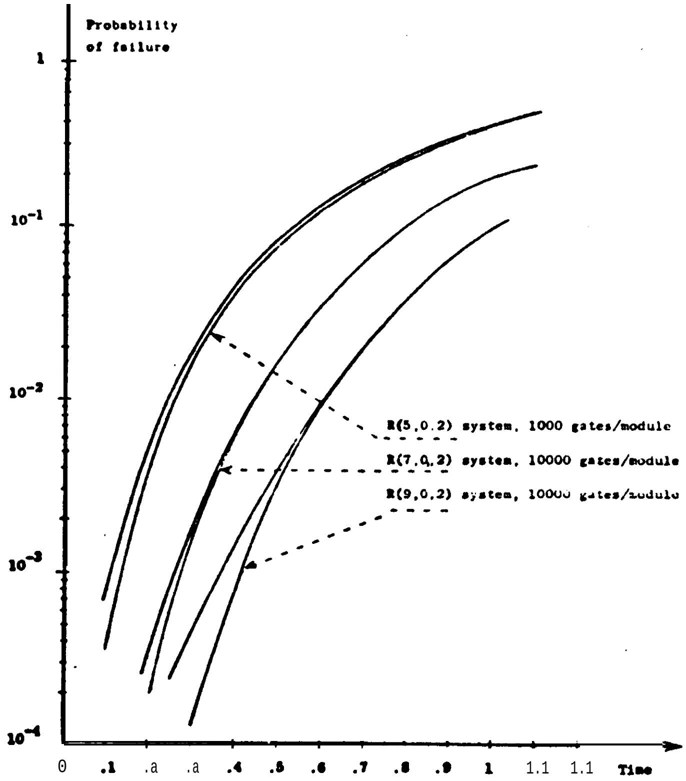


Fig. 6 . Bounds for the reliability of self-purging systems $(X_m=X_s=1)$.

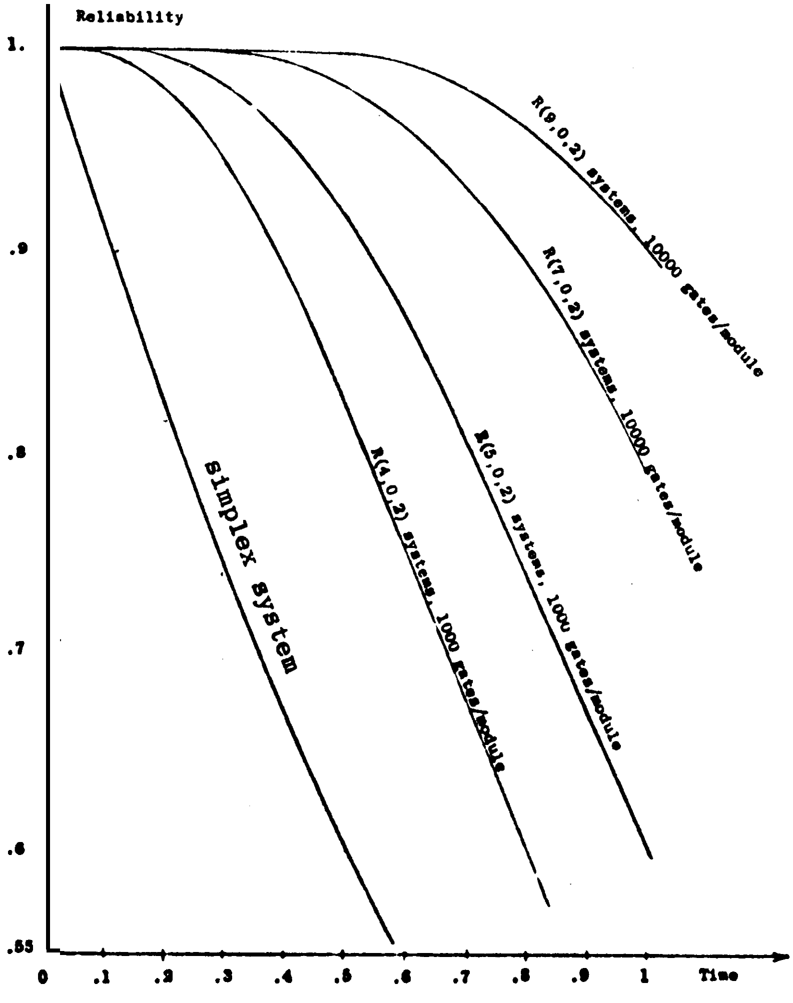
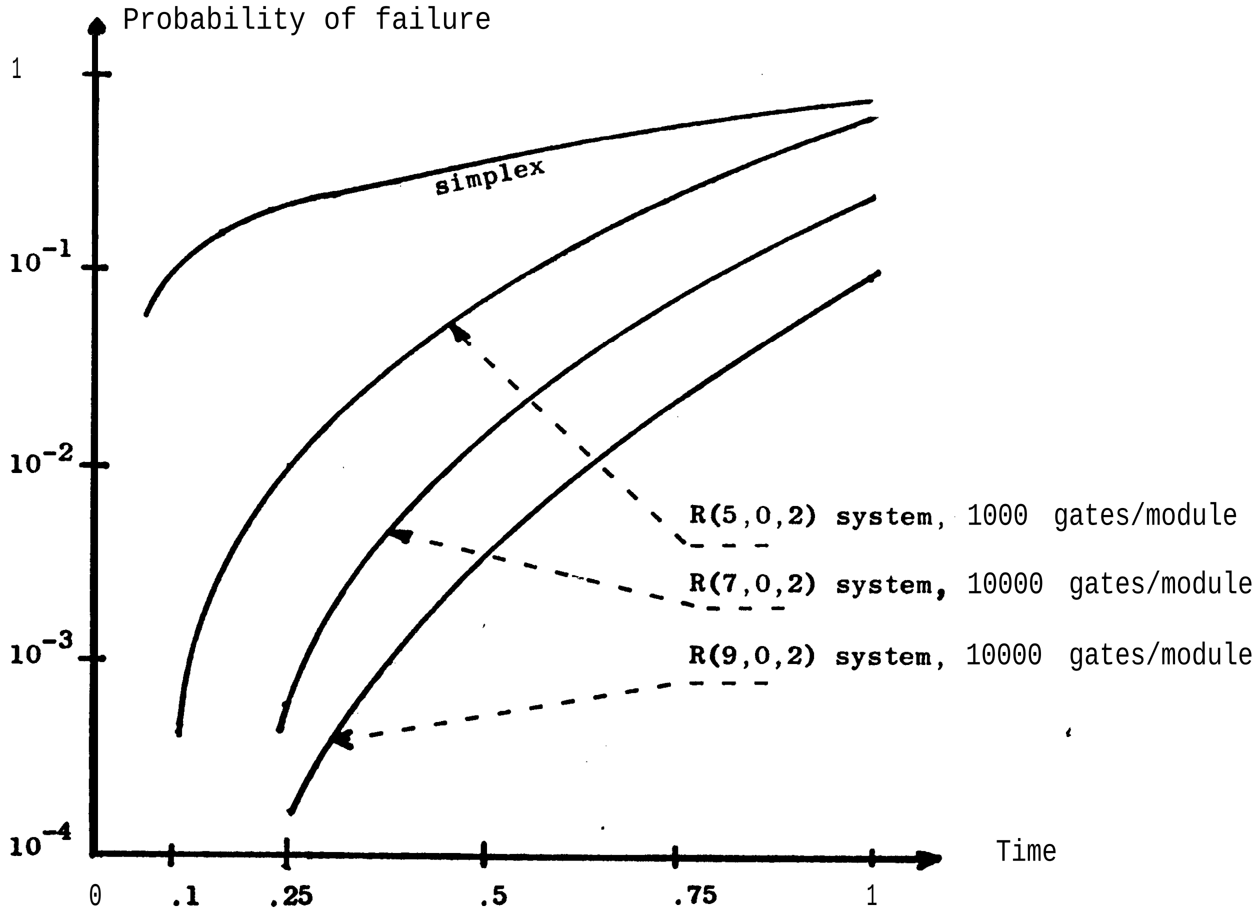


Fig. 7 . Reliability of self-purging systems $(X_m= X_s=1)$.



Fig, 8 . Probability of failure as a function of the mission time .

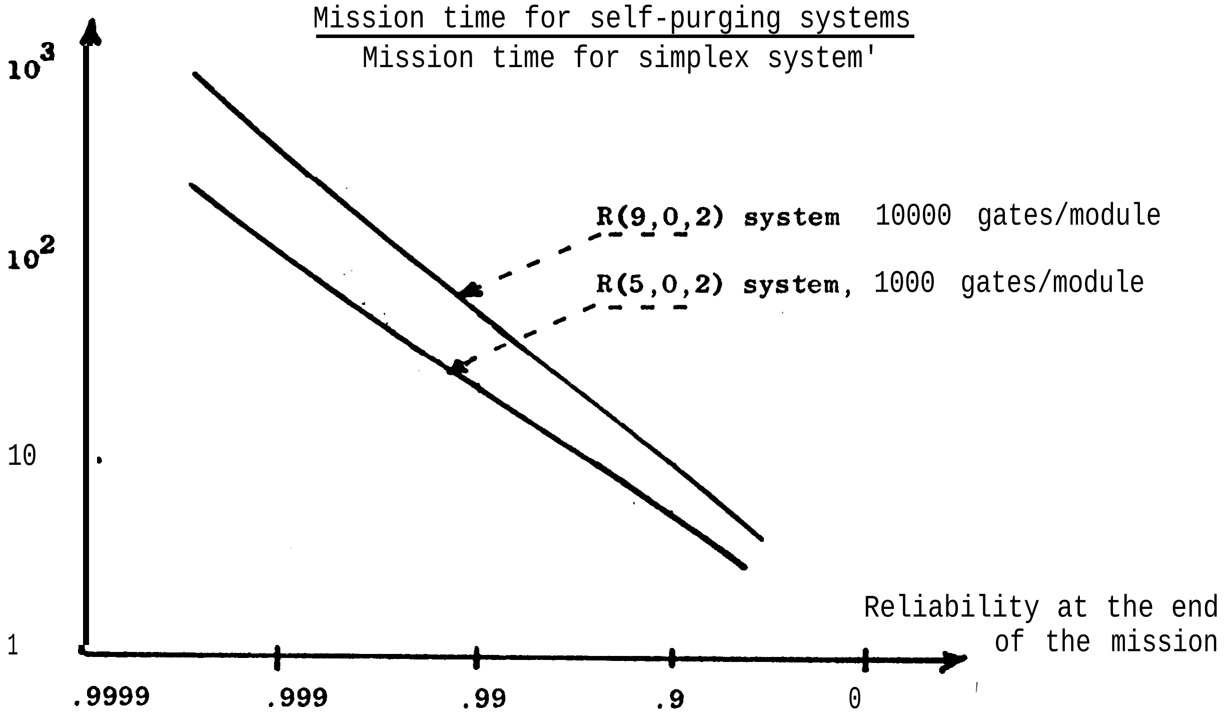


Fig. 9 . Mission time improvement

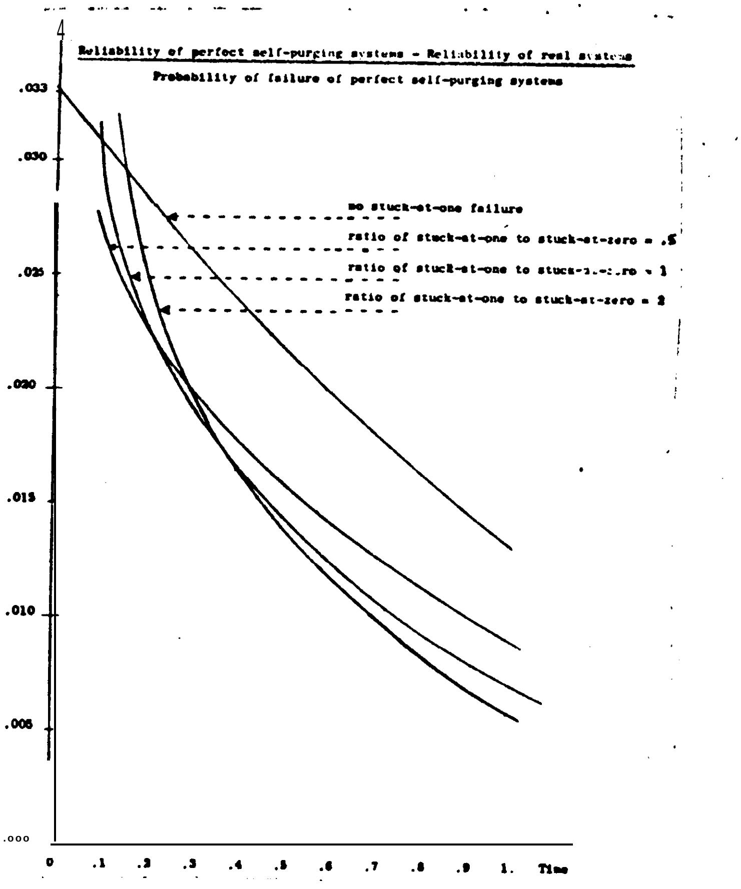


Fig. 10 . Influence of the relative frequency of stuck-at-one failures (4 modules, 1000 gates/module) .

For short mission times (of the order of one tenth of the module mean-life) the frequency of stuck-at-one faults should be reduced . For such small mission times, most of the system failures are due to stuck-at-one modified modules . System failure due to exhaustion of fault-free modules is unlikely . On the other hand, for longer missions, the modules should have a high relative frequency of stuck-at-one faults . Most of the system failures are due to module exhaustion . But the voter effective threshold is decreased by the existence of stuck-at-one modified modules, which allows the systems to operate with less fault-free modules .

If an automatic retry procedure is implemented inside each elementary switch, transient errors at the ouput of modules do not result in module removals . So, transient errors do not affect the system output as long as the effective threshold is larger than one . Once the effective threshold is reduced to one, transient faults that produce erroneous zero do not affect the system output (as long as there are enough fault-free modules) . On the other hand, a transient one will result in a incorrect system output for that particular time . All the fault-free modules will be considered to have disagreed with the voter output . But, because an automatic retry procedure exists in every elementary switch, transient ones will cause the system output to be erroneous only during the transient duration . So, as long as the effective threshold is larger than one, that is to say during most of the system life time, transient faults in modules do not have any effect on the overall system reliability . Fig. 11 gives some more quantitative results on the effects of transient module errors .

4-C Influence of switches on overall reliability

One of the most important advantages of self-purging systems is the switch simplicity . Switches for self-purging systems compare favorably with switches for hybrid systems [10, 11] . Switch efficiency can be characterized by the relative difference between the real probability of failure and the probability of failure that is obtained when switches are considered perfect :

$$\frac{\text{Probability of failure} - \text{Probability of failure with perfect switch}}{\text{Probability of failure with perfect switch}}$$

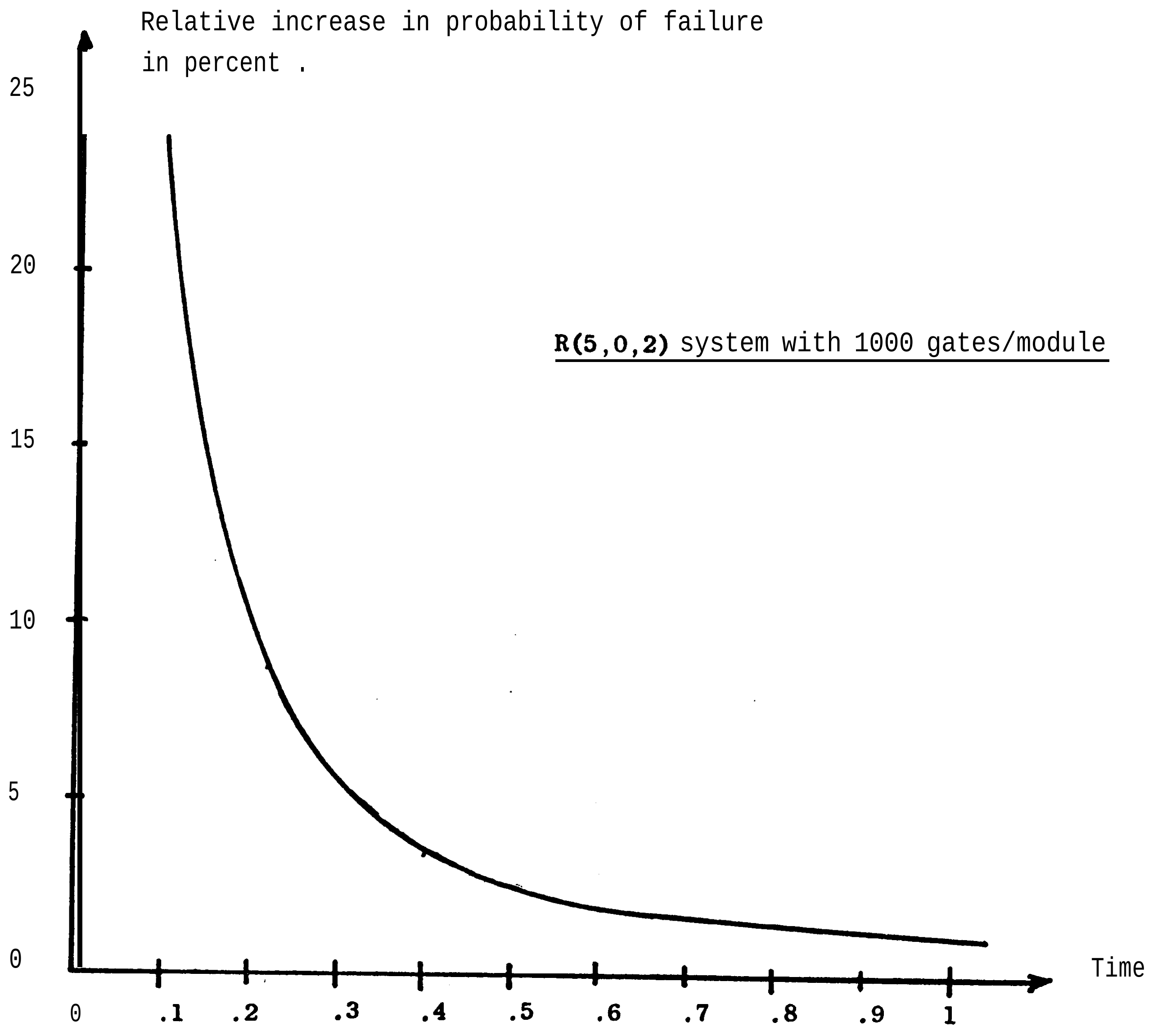


Fig. 11 . Maximum increase in probability of failure due to transient failures .

A ratio of zero means that switches are perfect, while a ratio of x means that the real probability of failure is (x+1) times higher than what is computed assuming perfect switches . Fig. 12 shows this ratio for three self-purging systems with the same modules . For a given mission time, switch efficiency decreases as more modules are added to self-purging systems . Switch efficiency approaches zero as mission times are reduced (Fig. 13) . On the other hand, switch efficiency approaches one as mission times increase . The importance of careful reliability modeling for systems involving switching mechanisms must be emphazised even when switches are as simple as for self-purging systems . Self-purging systems with 6 modules (one thousand gates per module) have a probability of failure 35 percents larger than what is computed when switches are not taken into account, for a mission time equal to one fifth of the module mean-life . The curves of Figs. 12 and 13 give a simple way to obtain self-purging system reliability from the estimated (ideal) reliability .

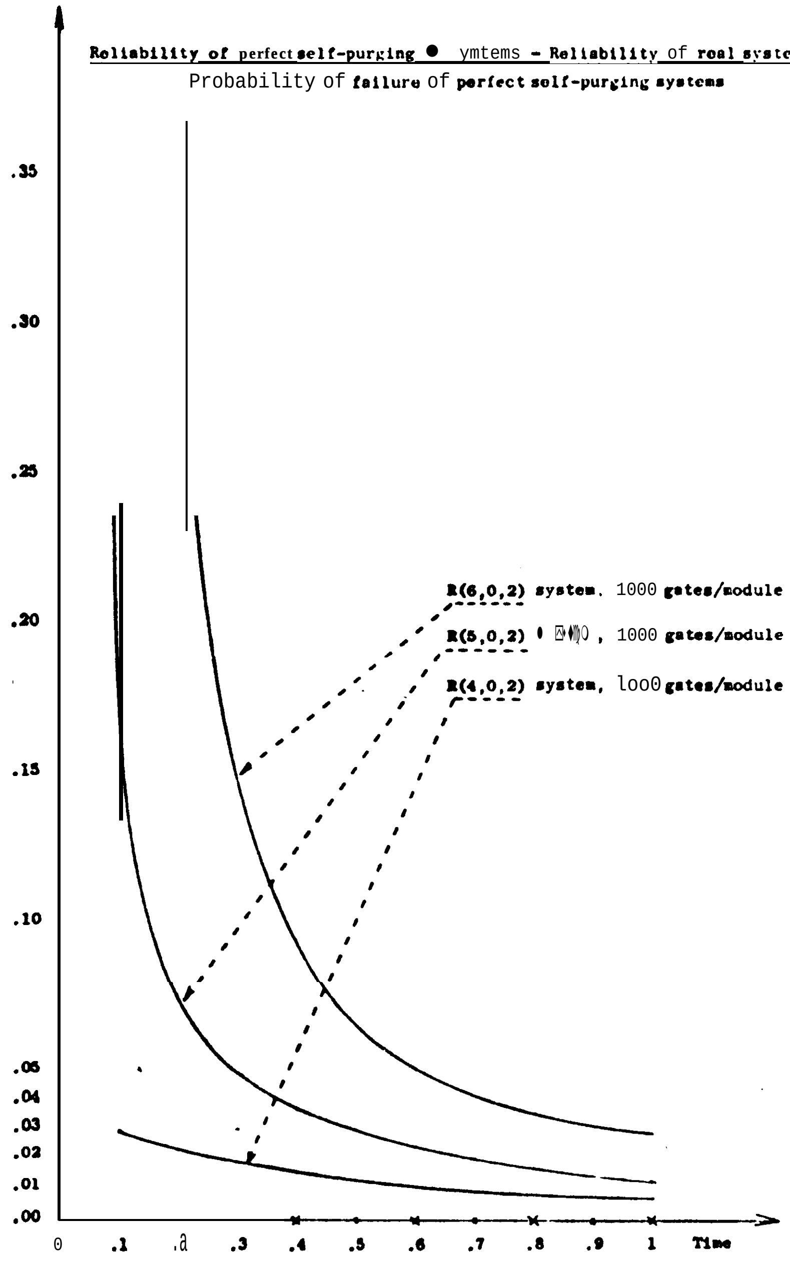
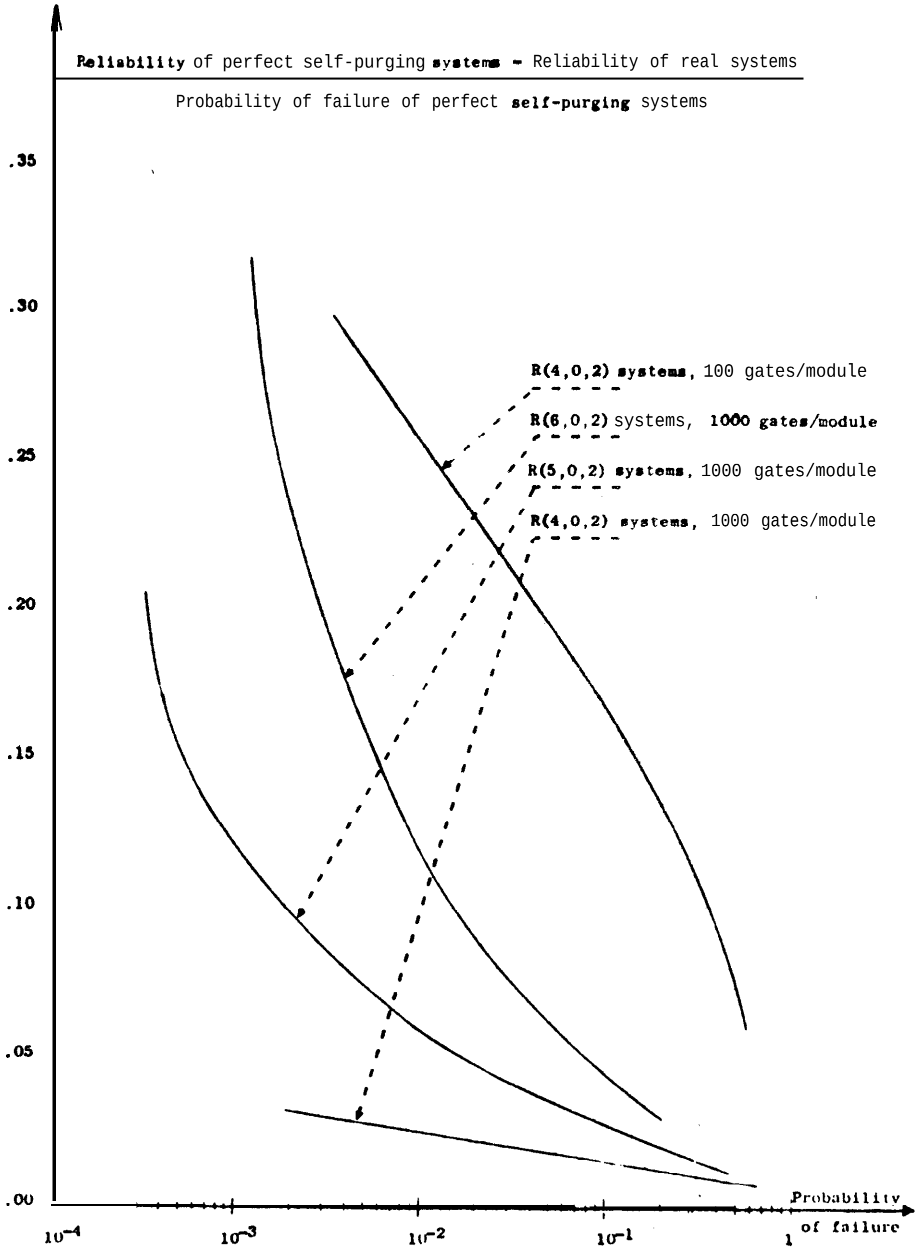


Fig. 12 . Decrease in reliability due to switch reliability $(X_m=X_s=1)$.



Fig, 13 . Decrease in reliability due to switch reliability as a function of system reliability $(X_m = X_s = 1)$.

5 SELF-PURGING SYSTEMS VERSUS OTHER REDUNDANT SYSTEMS

Self-purging systems are simple, efficient and their performances can be accurately evaluated . However, they are not necessarily the best systems for every application . For each application, different kinds of redundancy should be compared for their ability to meet the requirements, their cost, simplicity and the confidence that can be given to the results of their models .

For extremely short missions and high cost of failure, like short manned space flight, it is of the upmost importance to have extremely reliable systems . Stand-by redundancy should not be used, for their reliability is severely limited by the reliability of the switching mechanisms [18] . Some single switch failures can cause system failure . In reference [18], it is shown that, for extremely short missions, the best number of spares is one and such stand-by systems are less interesting than simplex systems . Furthermore, stand-by systems are extremely difficult to model due to switch complexity . Hybrid and self-purging systems present the same disadvantages . Most of their failures are due to switch failures and not to exhaustion of fault-free modules . The most interesting redundancy is masking redundancy . There is no switch (only a voter) . Any single module failure is masked, and,for mission times that are extremely short, NMR reliability is almost equal to the voter reliability . A trade-off must be found between the number of modules and the voter reliability .

When mission times are larger than a few tenth of the simplex system mean-life, self-purging systems are more performent than NMR systems (Fig. 14) . When mission times are between a few tenth and a few times the simplex mean-life, the positive effect of large dormancy factor on the reliability of stand-by or hybrid systems is small . Furthermore, switches for hybrid and stand-by systems are more complex and less reliable than switches for self-purging systems . As it can be seen in Fig. 14, self-purging redundancy is likely to be the best and simplest solution . Moreover, the exact reliability can be computed for self-purging systems, while only approximate results are available for hybrid or stand-by systems (unless prototypes are used) .

For very large mission times (several times the simplex mean-life), stand-by and hybrid systems perform better than self-purging systems (for the same cost) if it is possible to take advantage of large dormancy factors . When dormancy factor3 are equal to one, or can not be taken advantage of, self-purging systems are more efficient than hybrid systems because of simpler

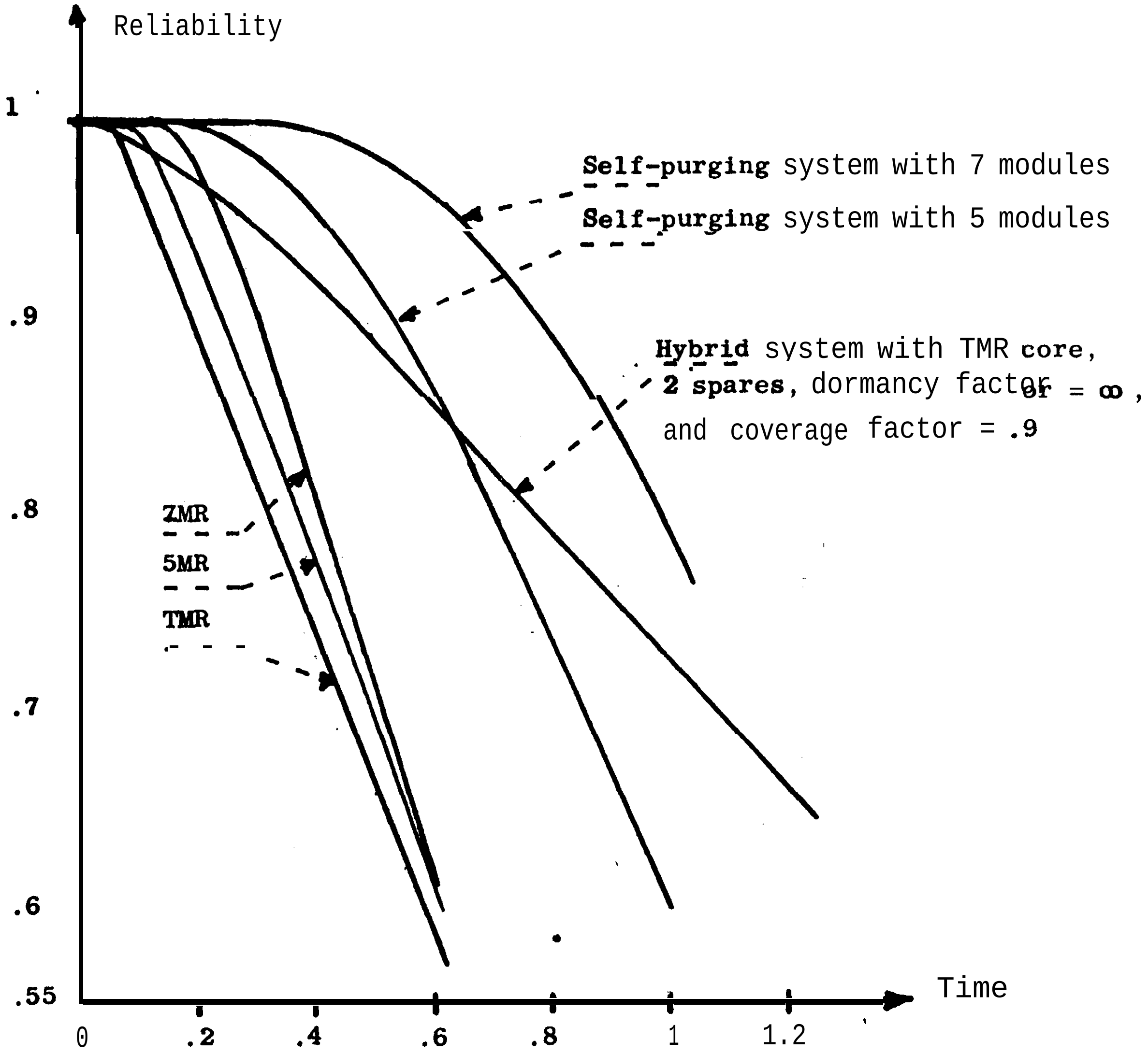


Fig. 14 . Comparison of various redundancy schemes .

switches .

So, for extremely short mission times, masking redundancy is optimal . When missions have duration of the same order as the simplex system mean-life,or when power-off electronics has the same failure rate as power-on electronics, self-purging redundancy should be used . For long mission times, stand-by systems provide the best ratio of performance over cost .

6 CORRECTNESS OF VARIOUS MODELS TO TAKE INTO ACCOUNT SWITCH RELIABILITY

Because of the accurate modeling of self-purging systems, it is possible to use them to check the accuracy of methods that are used to approximate the effect of switch reliability on the overall system reliability of redundant systems that use switching mechanisms .

The simplest, and also the crudest, way is to assume that a fault-free switch is required for correct system operation . This model applied to self-purging systems gives the results shown in Fig, 15 . As it can be seen, this is a pessimistic estimation . Not every switch failure cause a system failure. Switches, especially switches for self-purging systems, have some inherent fault-tolerance .

Another way to take into account switch reliability is to include each elementary switch into the corresponding module, to compute the reliability of the modified modules (a modified module is defined here as a module in cascade with its elementary switch) and then, to compute the system reliability assuming that switching is perfect . This method does not take into account the interdependence between elementary switches as it can be the case for hybrid systems (cf. the iterative cell switch of Fig. 1) . However, when applied to self-purging systems, this method gives good results for mission times that are not too small (Fig, 15) . But, for short mission times, the results are very optimistic . This method does not account for the system failures due to saturation of the voter by stuck-at-one modified modules (and these failures form the most important set for short missions) . So, the use of this method should be prohibited for ultra-reliable systems (the model accuracy approaches zero as mission duration gets shorter) .

Another method has been developed by W.G. Bouricius, W.C. Carter and R.P. Schneider, [1] . Each replacement of a failed module by a spare has a probability of success, c . The probability c is called the coverage factor . For self-purging systems, it is possible to associate such a probability to each removal of a failed module . The results given by this method are plotted in Fig. 15 . The coverage factor that has been chosen is the ratio of module complexity to modified module complexity (one minus the complexity of an elementary switch over the complexity of a module) . With these value, the reults are pessimistic . The reliability is higher than what is estimated . It may be possible to get a better approximation by choosing a slightly higher

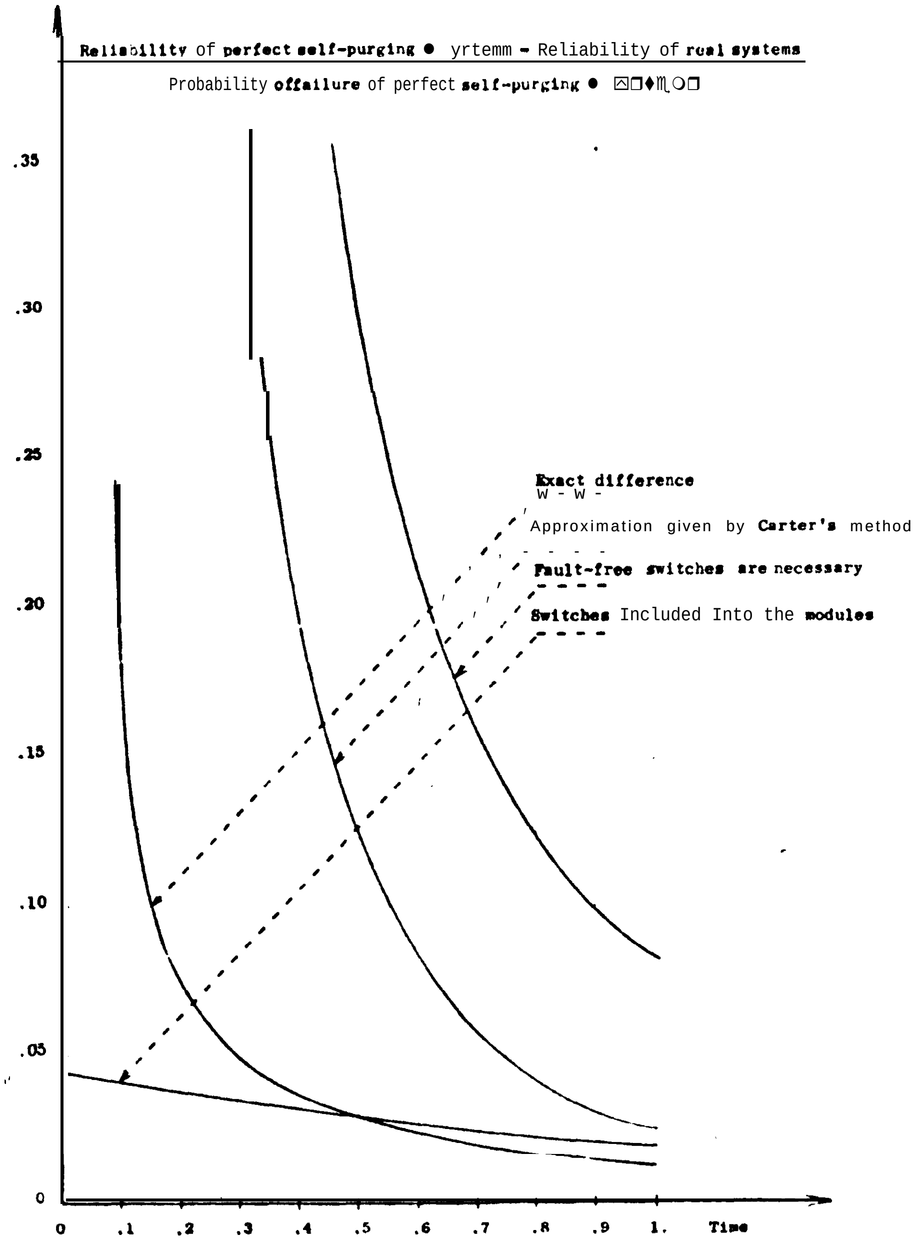


Fig. 15 . Comparison with results given by various models .

value for c . However, it is difficult, without more information, to decide what value should be taken for the coverage factor . The fact that coverage factors are the most critical factors in the reliability equations does not increase the confidence that can be put in this method . However, this method is very interesting when a detailed simulation or the use of prototypes yield a method to determine the coverage factors from the system characteristics .

7 CONCLUSIONS

Self-purging redundancy is an efficient method to increase the reliability of digital systems . Practical design is simple and straightforward , Switches can be decomposed into independent elementary switches which are only simple mechanisms to force the output of failed modules to zero . Because of their simplicity, switches are highly efficient and have some Ynherent fault-tolerance .

Exact reliability can be computed, even though the computations are complex . Tight bounds are obtained more simply . The possibility of obtaining reliability functions allows to optimize self-purging systems for the required applications . Reliability is fully computed from the module and switch failure rates . No use of critical parameters, like coverage factors, is made . Confidence in the results is limited only by the uncertaincy on the figures for the failure rates .

Switch efficiency was quantitatively characterized . The importance of switch reliability increases as the requirements for system reliability become more severe . The type of the most frequent module errors also influences the overall reliability . For ultra-reliable systems, the frequency of erroneous ones at the module outputs should be reduced . Transient failures have negligeable effect on system reliability when switches are provided with retry mechanisms that are simply implemented in hardware .

The domain of application for self-purging reliability regroups all applications asking for high reliability at the end of missions which duration is of the same order as the module mean-life , Ultra-high reliability at the end of short missions is best achieved by massive redundancy, while stand-by and hybrid systems provide better reliability for long missions if it is possible to take advantage of large dormancy factors . One other fact that acts in favor of self-purging systems is the possibility to use fail-safe logic for the module, thus reducing the probability of erroneous ones at the module outputs .

When redundancy is used to achieved ultra-high reliability, careful modeling must be made . Most of the methods that are used to take into account switch reliability give-only approximate results . The confidence that can be granted to their results must be critically examined, as it was shown by reference to self-purging systems .

REFERENCES

1. A. Avizienis, "Design of Fault-Tolerant Computers", FJCC, Vol. 31, pp. 733-743, 1967 .

2. W.G. Bouricius, W.C. Carter and P.R. Schneider, "Reliability Modeling Techniques for Self-Repairing Computer Systems", Proc. ACM 1969 Annual Conference, pp. 295-305, also IBM Report RC-2378 .

3. J. Goldberg, K.N. Levitt and R.A. Short, "Techniques for Realization of Ultra-Reliable Spaceborn Computers", Final Report, Phase I, SRI project 5580, Stanford Research Institute, Menlo-Park, California, September 1966 .

4. J. Von Neumann, 'Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components", Automata Studies (Annals of Mathematical Studies), C.E. Shannon and J. McCarthy, Eds., Princeton, N.J., Princeton University Press, 1965, pp. 43-98 .

5. R.E. Lyions and W. Vanderkulk, "The Use of Triple Modular Redundancy to Improve Computer Reliability", IBM J. Res. Develop. Vol. 6, 1962, pp. 200-209 .

6. R. Toeste, "Digital Circuit Redundancy", IEEE Trans. on Reliability, June 1964, pp. 42-61 .

7. F.P. Mathur and A. Avizienis, "Reliability Analysis and Architecture of a Hybrid-Redundant Digital System : Generalized Triple Modular Redundancy with Self-Repair", Proc. SJCC, Vol. 36, 1970, pp. 375-383 .

8. J.P. Roth, W.G. Bouricius, W.C. Carter and R.P. Schneider, "Phase II of an Architectural Study for a Self-Repairing Computer", SAMSO TR67-106, November 1967 .

9. J.K. Knox-Seith, "A Redundancy Thechnique for Improving the Reliability of Digital Systems", Stanford Electronics Lab., Tech. Rep. No, 4816-1, December 1963 , Stanford University, Stanford, California .

10. D.P. Siewiorek and E. J. McCluskey, "An Iterative Cell Switch Design for Hybrid Redundancy", IEEE Trans. on Computers, Vol. C-22, No. 3, March 1973 .

11. D.P. Siewiorek and E.J. McCluskey, "Switch Complexity in Systems with Hybrid Redundancy", IEEE Trans. on Computers, Vol. C-22, No. 3, March 1973, pp. 276-282 .

12. K.N. Chandy, C.V. Ramamoorthy and A. Cowan, " A Framework for Hardware-Software Tradeoffs in the Design of Fault-Tolerant Computers', FJCC, 1972, AFIPS, pp. 55-63 .

13. W.H. Pierce, 'Adaptive Vote-Takers Improve the Use of Redundancy', in Redundancy Techniques for Computing Systems, Spartan Books, Washington, D.C., 1962, pp. 229-250 .

14. N. Tokura, T. Kasami and A. Hashimoto, "Fail-safe Logic Nets', IEEE Trans. on Computers, Vol. C-20, March 1971 .

15. R.C. Ogus, "Fault-tolerance of the Iterative Cell Array Switch for Hybrid Redundancy", IEEE Trans. on Computers, Vol. C-23, No. 7, July 1974 .

16. J.A. Abraham and D.P. Siewiorek, "An Algoritm for the Accurate Reliability Evaluation of Triple Modular Redundancy Networks', IEEE Trans. on Computers, Vol. C-23, No. 7, July 1974 .

17. W.G. Bouricius, W.C. Carter, D.C. Jessep, R.P. Schneider and A.B. Wadia, "Reliability Modeling for Fault-tolerant Computers', IEEE Trans. on Computers, Vol. C-20, No. 11, November 1971 .

18 J. Losq, "Influence of Fault-detection and Switching Mechanisms on the Reliability of Stand-by Systems", FTC/S, Paris, June 18-20, 1975, Digest, pp. 81-86 .