PERFORMANCE-RELATED DEPENDABILITY EVALUATION OF SUPERCOMPUTER SYSTEMS*

J. ARLAT and J. C. LAPRIE

Laboratoire d'Automatique et d'Analyse des Systèmes du C. N. R. S.

7, Avenue du Colonel Roche 31400 Toulouse France

ABSTRACT

The paper presents an example of performance-related dependability evaluation of a supercomputer structure corresponding to an MIMD multiprocessor system. The approach presented addresses the problem of deriving a model that is tractable yet representative of the behavior of a complex system. This is achieved by means of an intensive validation study, and through the evaluation of measures of interest which account for the specific operating requirements characterizing the system under investigation: (i) maintain very high throughput over a long period of time, (ii) provide an efficient operational life-cycle.

INTRODUCTION

The most recent advances in both hardware and software technology have made it possible to develop multiprocessing systems, intended for very high speed computation through the use of parallel computing, known as supercomputers or supersystems.

Although primarily designed for high speed, supersystems exhibit essential dependability requirements resulting from the need for both continuous operation that is imposed by the large amount of computation to be handled, and an efficient operational life-cycle taking account of maintenance phases. Moreover, as previously stated in [AVI 78], the computation speed of these systems has reached such a level, as compared with the speed of manually controlled maintenance, that only the inclusion of fault-tolerance will allow an acceptable reduction in the amount of computation lost during repair . It follows that early attention has to be paid to dependability issues, as well as performance issues, and comprehensive evaluations have to be performed in order that objective decisions be taken at the different stages of the development process.

The major problem encountered in the evaluation of a large multiprocessing system is the derivation of a tractable model representative of its behavior in order to derive interesting measures of the quality of the service that can be expected. This paper presents a contribution to this problem on the basis of the evaluation of an actual supercomputer structure.

Section 1 introduces comprehensive evaluation measures that are relevant for the determination of the expected quality of service of such supersystems. A brief description of the structure of the system under investigation is presented in **section 2. Section 3** deals with the study of system behavior in the presence of faults, emphasizing the constraints imposed by the characteristics of the considered interconnection network. In **section 4** a representative simplified model is derived and discussed; in particular, its validity with respect to omitted states is verified in detail. Proper exploitation of the model is then presented in **section 5**: life-cycle reliability, availability and performance related measures that were introduced in section 1 are evaluated; the performance-related measures provide a broader insight into system behavior thus allowing for motivated choices in the determination of actual system parameters especially regarding the allowed number of degradations.

DEFINITION OF RELEVANT EVALUATION MEASURES

Various attempts to derive adequate performance-related measures have been reported; they correspond mainly to extended dependability measures obtained with more or less refined procedures [ALV 64, BEA 78, LAP 79, GAY 79, MEY 80, HUS 81].

The complexity of the system under investigation led us to adopt a "divide and conquer" strategy and thus consider the same approach as the one introduced in [ALV 64] and more recently stated in [LAP 79] and [HUS 81] where the state probabilities are weighted by coefficients representative of system performance which are evaluated by a separate model. In order to precisely define the considered joint dependability-performance measures, we present first the dependability measures to which they are related.

Dependability Measures

Classically, the most relevant measure for repairable systems is **availability** and more precisely, when only physical faults are accounted for (which is currently the most common case), the asymptotic availability which is a measure of the proportion of time spent in the operating states with respect to total time.

Reliability, as a measure of continuous accomplishment of system tasks, from an initial instant of reference, is not so often considered for maintainable systems, mainly because the time for accomplishing a given set of tasks is usually small when compared to the time to failure. However, this is no longer true for supercomputers as the computation time required for the processing

* This work was performed under SINTRA contract N° 84255/55295/AA/BS.

of a long and complex task becomes comparable to the time to failure.

Thus, reliability is here a measure of prime interest, but, as the system is repairable, the adequate instant of reference to consider is the time when the system has just been repaired. We introduce the notion of **Life-Cycle Reliability:** $LCR(t_0, t_0+t)$, to characterize this measure. It is defined as the **probability of failure-free operation of the system during the time interval of length t starting at the instant t_0 when the system is brought up after a failure.**

When all the considered events (error manifestation, repairs, etc.) are assumed to be exponentially distributed, i.e. characterized by constant hazard rates, system behavior can be modeled as a Markov process. Although this is widely recognized for error manifestation, constancy of repair rates is a priori unrealistic, but constitutes in many cases a practically satisfactory hypothesis, as shown in [LAP 75, LAP 81].

We adopt hereafter a matrix formulation, derived from [COR 75]. Let $S = \{ s_i, i \in I\}$, be the state space of the system; S may be partitioned into two subsets, $S_U = \{ s_i, i \in I_U\}$ and $S_D = \{s_i, i \in I_D\}$, corresponding to system up (success) and system down (failure) conditions, respectively. Let $\mathbb{A}= [\lambda_{ij}]$, i and $j \in I$, denote the transition matrix; the partition of S induces the following partition on $\mathbb{A}$:

$$\mathbb{A} = \begin{bmatrix} \mathbb{A}_{UU} & \mathbb{A}_{UD} \\ \mathbb{A}_{DU} & \mathbb{A}_{DD} \end{bmatrix} \begin{matrix} \} S_U \\ \} S_D \end{matrix}$$
$$\underbrace{\qquad}_{S_U} \qquad \underbrace{\qquad}_{S_D}$$

When only physical faults are considered, system behavior quickly converges towards an asymptote. The asymptotic state probabilities P_i are given by the relation:

$$\mathbb{P} = [P_i] = \mathbb{V} \cdot \mathbb{A}_m^{-1} \qquad (1)$$

where $\mathbb{A}_m$ is any modified transition matrix deduced from $\mathbb{A}$ by replacement of the mth column with 1's, and $\mathbb{V}$ is a line vector whose entries are all zero, except for the mth which is equal to one. Thus, the life-cycle reliability also quickly converges to an asymptote with respect to t_0; in the sequel, the term "life-cycle reliability" will denote:

$$LCR(t) = \lim_{t_0 \to \infty} LCR(t_0, t_0+t) \qquad (2)$$

and is defined as the **probability of failure-free operation of the system during the time interval of length t starting at the instant when the system becomes operational again after a failure, in the asymptotic behavior with respect to the up-down alternance.**

The expression for LCR(t) is thus:

$$LCR(t) = \mathbb{P}_U(0) \cdot \exp(\mathbb{A}_{UU}\, t) \cdot \mathbb{1}_U \qquad (3)$$

where $\mathbb{P}_U(0)$ is the initial probability vector for states in S_U, defined as:

$$\mathbb{P}_U(0) = \frac{\mathbb{P}_D\, \mathbb{A}_{DU}}{\mathbb{P}_D\, \mathbb{A}_{DU}\, \mathbb{1}_U},$$

and $\mathbb{1}_U$ is a summation row vector whose entries are all equal to one.

It is interesting to note the relationship between life-cycle reliability and asymptotic availability A. Indeed, **A = MUT/(MUT+MDT)**, where:

- **MUT** is the Mean Up Time which is the mean time to failure corresponding to the initial state probability vector defined for relation (3), i.e.,
$$MUT = \int_0^\infty LCR(t)\, dt, \qquad (4)$$
- **MDT** is the Mean Down Time which is the mean time to repair considering the initial instant to be the time when the system has just failed, in the asymptotic behavior.

Performance-Related Measures

For defining performance measures, it will be assumed that the states of the model are sorted according to the different modes of operation i, i=1, 2, ..., n, according to their respective performance level. A performance index k_i is associated with each state of the same mode.

In practice, the k_i may be any convenient representation of system performance rating the number of tasks achieved per second (e.g. the number of FLOPS), the time to process one task, the potential number of resources, etc.. It will be also considered that there exists an order relationship between the k_i, such that:

$$k_1 > k_2 > \ldots > k_n$$

According to these assumptions, joint dependability-performance measures can be easily derived for both LCR(t) and A.

The pointwise processing capacity before system down can be expressed as,

$$C(t) = \mathbb{P}_U(0) \cdot \exp(\mathbb{A}_{UU}\, t) \cdot \mathbb{K}_U$$

where $\mathbb{K}_U$ represents the performance index vector whose entries are the performance index associated to each state in S_U; the mean capacity cumulated before sytem down is defined as:

$$MCCTF = \int_0^\infty C(t)\, dt = -\mathbb{P}_U(0) \cdot \mathbb{A}_{UU}^{-1} \cdot \mathbb{K}_U$$

The asymptotic capacity is given by:

$$AC = \mathbb{P} \cdot \mathbb{K}$$

where P represents the asymptotic state probability vector as defined by relation (1) and K is the performance index vector over S.

Numerical values of these measures that will be presented in the sequel for the system under investigation were obtained through processing the models by the SURF evaluation program [COS 81].

STRUCTURE OF THE SYSTEM UNDER INVESTIGATION

Functional View

The general structure of the system under investigation is shown in figure 1. It is composed of N processor modules (P_0-P_{N-1}) and N memory modules (M_0-M_{N-1}) interconnected by a symmetric network. An Omega type network [LAW 75] has been selected in order to provide an acceptable tradeoff between the hardware and control complexity, flexibility,

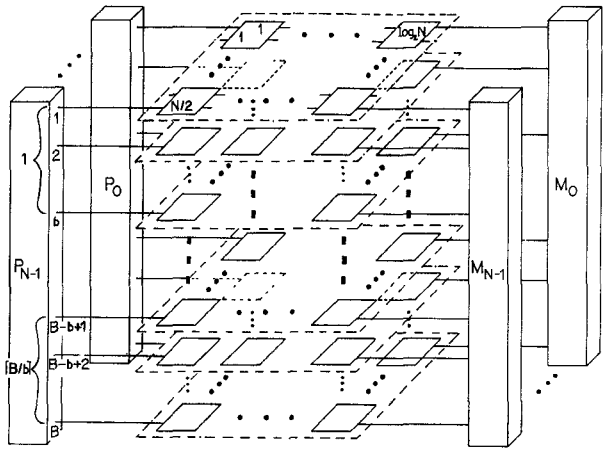

FIGURE 1: Structure of the System.

rearrangeability, delay, etc., [SHE 80, FEN 81].

Processor modules are standard array processors configured with a small local memory and **Memory** modules have each a capacity of 1 M words of 64 bits.

As indicated in figure 1, the **network** ensures bidirectional switching of B lines for each module; these lines represent address, data, and control bits. Simultaneous switching of these B bits may be obtained by the superposition of B identical 1-bit wide slices. However, in practice, specific circuits allowing the switching of b bits in parallel have been developed [LAR 81]. The network can thus be viewed as the superposition of $\lceil B/b \rceil$ slices logically equivalent to the NxN Omega network described in figure 2 (for the case where N=16).

Each slice is made up of $n=\log_2 N$ identical stages composed of N/2 (2bx2b) switching modules, each of them being able to take 2 states: the "through" state and the "cross" state. Hence, the total number of switching modules is:

$$N_S = (N/2) \lceil B/b \rceil \log_2 N$$

where it has been considered that, as a consequence of network structures developed in practice, N is a power of 2, and $\lceil x \rceil$ represents the smallest integer

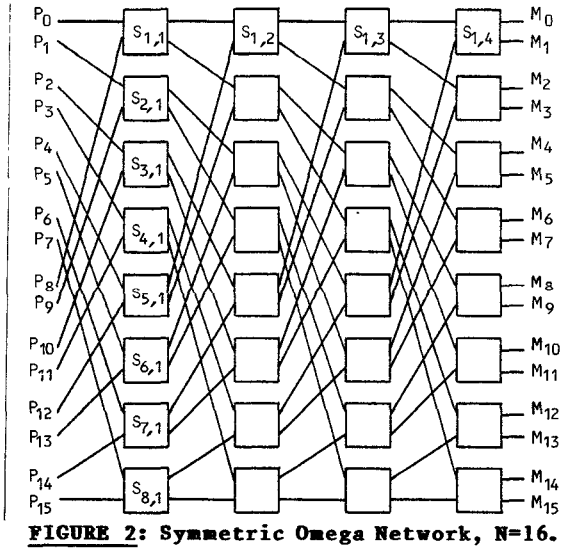

FIGURE 2: Symmetric Omega Network, N=16.

greater than or equal to x. The actual structure considered is characterized by N=16, B=100 and b=4, which leads to N_S=800. Each switching circuit slice is implemented on a single board.

Dependability Considerations

In practice, a supervisor module is required to manage the activity of the system and constitutes the hard core of the structure.

However, as a centralized implementation of this module is currently considered [PLA 79], its protection can be performed by means of classical techniques (duplex, TMR, etc.), provided that the dependability level achieved does not weaken the dependability of the whole system. Accordingly, in the sequel, we restrict our consideration to the functional active system made up of the processor, memory and switching modules previously identified.

The structure of the system is a priori well suited for implementing a graceful degradation strategy which is described in the next section. Furthermore, as is now classical, memories (processor local memories and memory modules) are locally protected by error detecting and correcting codes.

SYSTEM BEHAVIOR STUDY IN THE PRESENCE OF FAULTS

Failure Assumptions

Processor and memory modules are considered as non-decomposable entities, and only the total failure of these modules will be considered. In order to account for the degradation of system performance resulting from faults in the network, the latter has been considered at the level of the switching module; only total failure of these modules has been assumed.

This total fault assumption is rather pessimistic, but allows for the definition of a relatively simple degradation strategy that is set forth in the sequel.

Total failure of a processor module leads to its removal from the system; the loss of a processor being easily taken into account by system reconfiguration and reassignment of remaining active processors.

Each memory module holds a part of the information that is essential for task execution in association with one (or more) processor modules. It follows that loss of a memory module due to a total failure may lead to the loss of the currently supported task, and therefore to severe penalties that could prevent processing resumption if adequate reconfiguration is not performed. In this study, it will be assumed that such procedures are implemented in order to restore system context even in the case of memory module loss.

According to its functional simplicity, the Omega network provides only one path between two terminal nodes. The failure of a single path means that communication among all nodes is no longer possible and a degradation strategy must be defined, considering the number of active nodes after a failure.

Degradation Strategy

From the organization of the network described in figure 1, it follows that each switching circuit

can be identified using a coordinate triplet (i, j, k), where:

- i = 1, 2, ..., (N/2), characterizes the position of a switching circuit in a stage of a particular slice,
- j = 1, 2, ..., $\log_2 N$, identifies the various stages in a slice,
- k = 1, 2, ..., $\lceil B/b \rceil$, characterizes the network slices (boards).

Failures of switching modules pertaining to the same stage have analogous consequences on system resources for all slices (Figure 2), so the principle of the degradation strategy can be characterized considering index j only.

Denoting the number of stages of a slice by $n=\log_2 N$, the failure of a switching module at stage j leads to the removal of several processor or memory modules; the type of resource, as well as the number of modules to delete, are identified according to the following conditions:

$j \leq \lfloor n/2 \rfloor$: deletion of 2^j processor modules,

$j > \lceil n/2 \rceil$: deletion of 2^{n-j+1} memory modules,

where $\lfloor x \rfloor$ is defined as the largest integer less than or equal to x. It should be noted that for odd values of n, the case where $j = \lceil n/2 \rceil$ has to be considered and one should explicitly specify the type of module to delete; however, the number of modules to remove is always: $2^j = 2^{n-j+1}$.

As an example, let consider figure 2 where switching modules are identified using previously defined i and j indices: failure of switching module $S_{1,2}$ leads to the removal of processors P_0, P_4, P_8, and P_{12}; if these processors were not deleted, they could have only reached half of the memory modules (M_0-M_7).

Although pessimistic, it can be implemented easily and moreover benefits from the two following points: (i) it ensures that the resulting configuration can be fully exploited, and (ii) it tends to minimize the risk of conflicts in the network.

DERIVATION OF A REPRESENTATIVE SIMPLIFIED MODEL

A birds-eye view of the system could be developed according to the previously presented degradation strategy, where all possible degradations of system resources are considered until complete exhaustion. However, such an approach is of a poor practical interest for the two main following reasons: (i) in practice, there exists a limit for degradation beyond which the processing power of the system is too low to ensure useful work, (ii) the number of system resources to be considered results in an exponential number of states and a practical study can be performed only when the most significant states are taken into account.

We summarize first the general assumptions, related to fault and maintenance processes, that support the model construction. Derivation of the model, carried out according to the remarks stated above, is then presented and discussed in detail.

General Assumptions

Fault Process

Constant failure rates, have been considered for all modules. This is acceptable for processor and switching modules, but is more questionable with respect to memory, where error correcting codes have been implemented. However, equating the failure rate to the inverse of the memory module mean up time (MUT) is a pessimistic assumption, as long as the time values considered are not too large with respect to the latter. Quantitative values selected are listed below for processor, memory and switching modules:

$\lambda_P = 1.1\ 10^{-3}\ h^{-1}$, $\lambda_M = 3\ 10^{-4}\ h^{-1}$, $\lambda_S = 10^{-6}\ h^{-1}$

λ_P and λ_S are constructor specified values, λ_M is the estimated value as indicated above.

Only total module failures are considered, (i.e. every fault leads to an attempt to system reconfiguration by means of resource removal).

Coverage factors [BOU 69, ARN 72], c_X, $0 \leq c_X \leq 1$, $X \in \{P, M, S\}$ are allocated to the degradation following the fault of the modules; they characterize the ability of the system to cope with the fault. Should the reconfiguration fail, a global system failure is assumed (factor $1-c_X$).

Maintenance Process

On-line maintenance is considered for processor and memory modules only, with rates:

$$\mu_P = 0.5\ h^{-1} \quad \text{and} \quad \mu_M = 0.5\ h^{-1}.$$

Due to the physical organization of the network (figure 1), switching modules cannot be maintained on-line as their repair needs the system to be stopped. When the system is stopped (safe down), it is assumed that all failed modules are repaired, the system being brought back into operation, with rate $\mu_S = 0.5\ h^{-1}$, when all modules have been repaired.

A single repairman policy is assumed and priority is given to the first failed module (processor or memory) for on-line maintenance.

As shown in [LAP 75, HEL 80], perfect maintenance is clearly optimistic, we thus consider here a "successful repair" factor r_Y, $0 \leq r_Y \leq 1$, $Y \in \{P, M\}$ for on-line maintenance operations in order to rate both system susceptibility and maintenance crew stress during such phases. Less informally, this factor can be defined as the **conditional probability of successful repair | on-line repair is initiated.** If repair is unsuccessful, total system failure is assumed (factor $1-r_Y$).

Global system maintenance, that puts all faulty modules back into operation is considered in the case of total system failure (failed down), with rate, $\mu_F = 0.1\ h^{-1}$.

Definition of the Stopping Strategy

The actual relationship between system configuration and performance level is complex and depends on conflicting influences (basic processing power, overhead, conflicts, etc.).

In the lack of more precise information on actual system performance, we present here a voluntarily simple attempt to characterize system performance that is devoted to specify the stopping strategy. It has been assumed that tasks are processed through creation of Processor-Memory (P-M) pairs, which allows for the use of a simplified queueing

model in order to evaluate the variation of the average processing time with respect to the number of available P-M pairs, denoted i.

The nominal structure corresponds to i=N=16 and the workload has been assumed to be equally distributed among the i available active P-M pairs, each of them forming an M/M/1 queueing model with parameters λ=L/i and μ=1/t, where L is the system workload expressed as the average number of FLOPS, and t is the average time to process 1 FLOP on a P-M pair.

Analysis of such a system by standard techniques [KLE 75] is straightforward, and leads to the following expression for the average FLOP processing time:

$$T(i) = i/(i\mu - L)$$

In the case where L= 20 MFLOPS and $\bar{t}$ = .5 μs, which represent respectively the expected average performance of the system and of the considered array processor, it can be determined that more than 50% of performance is lost when 4 P-M pairs are removed. This leads to the selection of the value 12 as the bound for the number of available P-M pairs in the stopping decision of the degradation strategy.

Limitation to the Most Significant States

In spite of the adopted degradation strategy, the model construction still requires that a large number of states be considered; in the worst case, one may have to consider the states corresponding to 56 successive faults (module failures), before reaching the stopping bound.

An important simplification can be obtained when considering the physical organization of the network that is described in figure 1. It can be noted that the failure of a single switching module in a slice ($i=i_1$, $j=j_1$, $k=k_1$), will render functionally useless all the switching modules in the same "column" ($i=i_1$, $j=j_1$, $k \in \{1, \ldots, \lceil B/b \rceil\}$, $k \neq k_1$). This consideration allows to account for at most 8 successive faults in the description of the system; however, in this case, the size of the model is still prohibitive (20582 distinct states).

Further simplification can be achieved noting that (i) the failure of a single switching module will prevent use of the terminal nodes to which it is connected, and (ii) the failure of the modules, which are directly connected to a switching module, make it functionally useless. This leads to a model with only 6 successive fault active states, but featuring 1308 states.

Having reached that point, one must recall that the considered system is maintainable: in such a system, according to previous experience in the study of complex systems [LAP 80], the influence of multiple fault active states decreases rapidly with the number of faults: the probability of an n-fault state being of the order n with respect to the ratio of the failure rates to the on-line repair rates.

The above remarks, introduced in order to simplify model construction, were carefully investigated. Part of this investigation is presented here; it consists of a quantitative sensitivity study of the system with respect to (i) the number of successive faults to consider, (ii) the failure rates of the functionally useless terminal modules as a consequence of the failure of a switching module. For clarity of the presentation, more detailed discussion of the validation process is postponed until the simplified model is introduced; we only list here the main results obtained.

The number of successive faults to consider can be limited to 2 with negligible influence on model accuracy. Also, neglecting the failure rates of inactive modules appeared to have no impact.

These conclusions allowed for the construction of a simplified model, with only 25 distinct states; the 30-state model shown in figure 3 is however more readable.

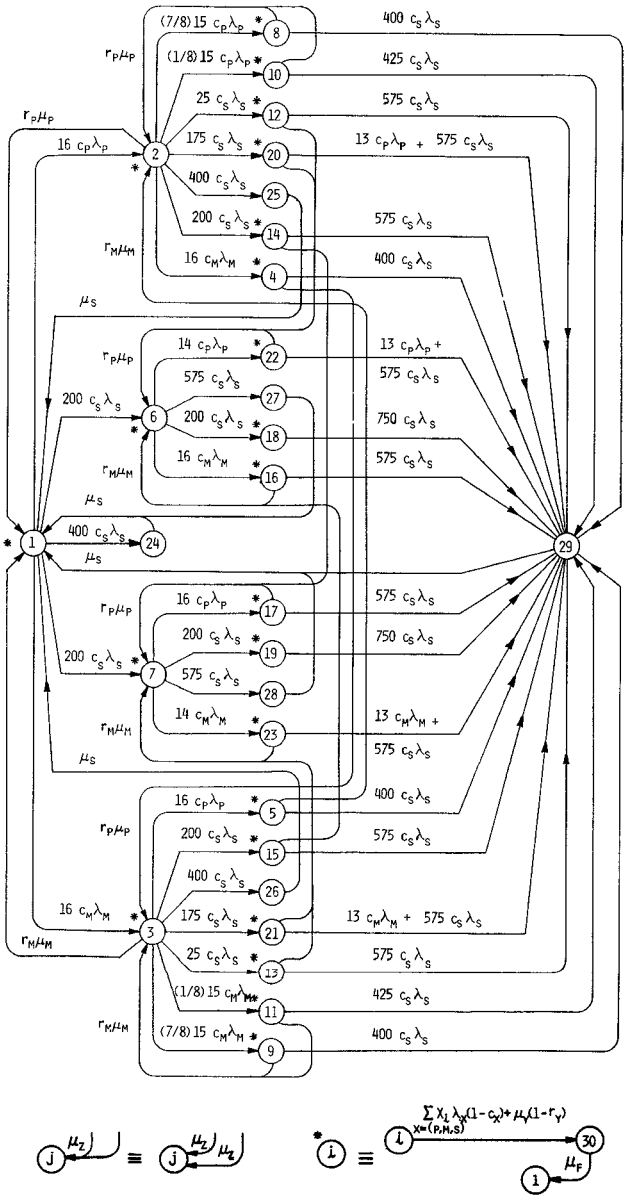

FIGURE 3: Simplified Model

State 1 represents the fault-free state of the system, the other states corresponding to various levels of performance. In particular, all states characterized by a number of active P-M pairs less than or equal to the Stopping Bound SB=12, are considered as safe-down states. Table 1 indicates the classification of states according to their level of performance; index k_i corresponds to the associated number of active P-M pairs; a null index has been considered for all configurations featuring a number of P-M pairs less than or equal to the SB.

States	1	2-5	6-19	20-23	24-30
Index k_i	16	15	14	13	0

TABLE 1: Performance Indices.

State 29 is a safe-down state which covers all triple-fault configurations corresponding to a number of active P-M pairs less than or equal to SB. State 30 is the system failure state resulting from a non covered fault or an unsuccessful on-line repair; no useful work is performed, thus it can be assumed as a failed-down state.

For clarity of the presentation, some transitions were merged as indicated at the bottom left of figure 3. Also, transitions to state 30 were omitted, and thus every active state (*) should have been actually represented as described at the bottom right of figure 3. The X_i, $X \in \{P, M, S\}$, correspond to factors related to the configuration of the active state i, $i \in \{1, 2, \ldots, 23\}$.

Validation of the Model

The validation process will be presented using the simplified model of figure 3 as a reference. The evaluation was actually carried out considering life-cycle reliability LCR(t), MUT, and asymptotic availability (A) as measures. However, for sake of conciseness, we restrict the presentation here to the results obtained for MUT and A concerning a limited number of simplifications that were investigated to derive the final model. The restriction to MUT still provides a representative measure of the continuous operating time, due to the strong connectivity of the graph of the transitions between the active states [PAG 80].

Furthermore, it should be noted that the validation study has been carried out considering ideal values for both coverage and successful repair factors; this in fact corresponds to the most pessimistic condition for establishment of the equivalence between simplified and actual models.

The study of the validity of the **limitation to double-fault active states** consists of comparing the results obtained when all triple-fault active states characterized by a number of P-M pairs greater than SB are included with those obtained when the latter are neglected.

In this case, the model is modified in order to take account of these triple-fault active states; as an example, figure 4 shows the modification related to state 18 from figure 3.

In practice, not all double-fault active states need to be expanded as indicated for state 18, but

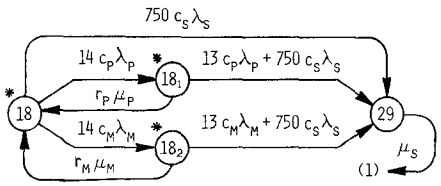

FIGURE 4: Extension of State 18.

only the most significant ones. They correspond to the states that are reached by transitions presenting poor competition between on-line maintenance and failure processes; in this particular case, the states to consider correspond to states 16, 17, 18 and 19, the other ones being much less significant.

Results presented in Table 2 indicate the difference observed when the triple-fault states are neglected or not. Unavailability (UA = 1-A) figures are given for better readability of the results. As can be seen, the discrepancy is very small, about 3 % for both UA and MUT, allowing the exploitation of the model with double-fault active states only.

Model	UA (x 10^3)	MUT (h)
Simplified	1.046	1910
Expanded	1.078	1853

TABLE 2: Influence of the Triple-Fault States.

Further simplifications consisting of limiting to single-fault active states only would not be accurate enough as indicated by the results shown in Table 3.

Simplified Model	UA (x 10^3)	MUT (h)
Double-Fault States	1.046	1910
Single-Fault States	2.750	725

TABLE 3: Limitation to Single-Fault States.

Consideration of failure rates of inactive modules, as a consequence of the failure of a switching module, leads to the modification of states 5 and 6, as well as of their outgoing transitions and states. Figure 5 illustrates the modification of state 6, considering processor module failure rates only for the expansion. It should be noted that this constitutes the major

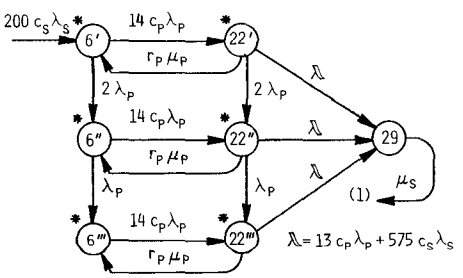

FIGURE 5: Extension of State 6.

Model	UA (x 10^3)	MUT (h)
Simplified	1.046	1910
Expanded	1.046	1910

TABLE 4: Failure of Inactive Modules.

point to account for, due to the symmetry of the study and to the relative influence of module failure rates.

Table 4 presents the corresponding results for the considered set of measures. Here again, great accuracy is preserved when using the simplified model.

EXPLOITATION OF THE MODEL

As representativity and accuracy of the simplified model were validated, thorough exploitation of this limited size model was made possible. For this purpose, measures introduced in section 1 have been used and systematic variation of the values of model parameters has been exercised. Only some of the obtained results [ARL 81] are presented here; attention is paid essentially to the modification of the stopping bound in the degradation strategy along with the variation of the coverage factors.

More precisely, this study presents the modification observed on system behavior when more restrictive SB's are considered, i. e., SB = 13,14,15.

Table 5 presents the observed variation for the MUT, the MDT and the UA, with respect to the considered SB's, for values 1. and .9 of the coverage factors c_X, $X \in \{P, M, S\}$.

SB	15	14	13	12
MUT (h)	43 (43)	719 (284)	1289 (340)	1910 (379)
MDT (h)	2.00 (2.80)	2.00 (7.27)	2.00 (8.29)	2.00 (8.78)
UA (x 10^3)	44.343 (60.998)	2.775 (24.926)	1.550 (23.769)	1.046 (22.668)

TABLE 5: Modification of SB with c_X = 1. (.9).

In the ideal case where c_X=1, a significant improvement of the MUT is noted between the case where no degradation is allowed (SB=15) and the behavior corresponding to the degradation of 1 P-M pair (SB=14); on the other hand, the modification of SB from 13 to 12, results in a more limited improvement.

For c_X=.9, the allowance for degraded operation leads to a less significant variation of the MUT. Another point to stress is the important increase of UA that is observed for each SB; this corresponds to the decrease of the MUT, but also and above all, to the degradation of the MDT, in the case of non ideal coverage factors.

Curves in Figure 6 and values in Table 6 present respectively the variation of the pointwise

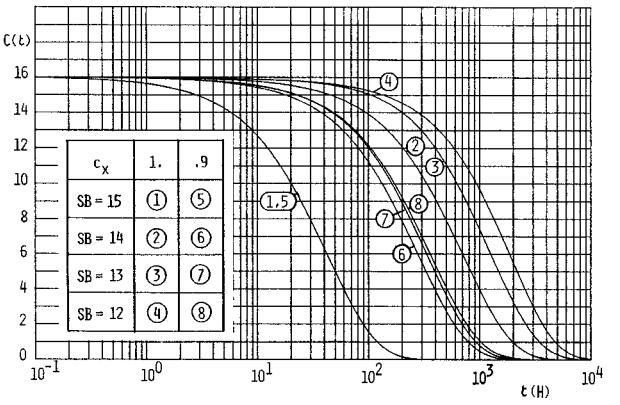

FIGURE 6: Influence of the SB on C(t).

SB	15	14	13	12
MCCTF Pairs x H	689 (689)	11471 (4540)	20493 (5411)	36474 (5945)

TABLE 6: MCCTF - c_X = 1. (.9).

capacity before system down C(t) and of the mean life cycle capacity cumulated before system down (MCCTF); they both confirm the previously stated remark concerning the MUT.

Table 7 presents the impact of the modification of the value of SB on the asymptotic capacity (AC). It can be seen that the allowance for degraded operation does not lead to a uniform improvement of AC, as opposed to the previous results concerning MUT, A, and C(t).

SB	15	14	13	12
AC P-M Pairs	15.290 (15.024)	15.912 (15.564)	15.832 (15.510)	15.258 (15.348)

TABLE 7: Asymptotic Capacity - c_X = 1. (.9).

This is especially significant when c_X=1., because in this case, the decrease of SB results in a large proportion of system operation being spent in a degraded mode. On the other hand, for c_X=.9, a significant compression of the results is observed when SB < 14; this results mainly from the important decrease of the probabilities of the states corresponding to very degraded operation. It has to be noted that further degradation of the c_X tends to counteract the impact of the allowance for degraded operation; all the results converge to those corresponding to SB=15.

From a practical point of view, the above results show that: (i) it is not worth implementing a sophisticated strategy intended for providing continuous operation of the system by means of an optimal use of remaining active resources, (ii) it is essential to implement efficient coverage mechanisms to benefit from the graceful degradation property.

It should be noted that the first remark provides an a posteriori support to the simplified determination of the stopping strategy which underlies the derivation of the model.

CONCLUSION

The paper presented a contribution to the study of supersystem structures. The considered approach was developed in application to an actual supersystem intended for ultrafast scientific computations.

The inherent complexity of the system has been mastered through a detailed sensitivity study during the system model validation phase. This made it possible to construct a simplified, but representative model of limited size. Accordingly, a thorough evaluation of the system, based on this model, could be performed. For this purpose, joint dependability-performance related measures were associated to reliability and availability measures in order to rate system quality of service.

The simplified model constituted a useful tool for system designers in the development process of the system. The results showed the prominence of coverage and successful repair factors characterizing the efficiency of detection and reconfiguration procedures, and of restoration procedures, respectively. Thus, major attention has to be paid to the improvement of such procedures, even if they are used in an elementary fail-safe degradation stategy, rather than to the implementation of a sophisticated degradation strategy dedicated to make optimal use of system resources, but that would be based on poor procedures.

ACKNOWLEDGEMENT

The authors would like to thank C. Michel at SINTRA for his support and confidence which made this work possible. Many thanks go to all members of the research team **Design and Validation of Fault-Tolerant Computer Systems** at LAAS for their friendly assistance and helpful technical suggestions; constant availability of J. E. Doucet is also gratefully aknowledged.

REFERENCES

ALV 64 W.H.VON ALVEN, Editor, **Reliability Engineering**, ARINC Research Corporation, Prentice Hall, 1964.

ARL 81 J.ARLAT and J.C.LAPRIE, "Dependability Evaluation of MARIANE", **Report of SINTRA Contract N° 84255/55295/AA/BS**, December 1981, available from SINTRA, 92602 Asnières, France, (in French).

ARN 72 T.F.ARNOLD, "The Concept of Coverage and its Effect on the Reliability Model of a Repairable System", **Proc. 2nd Int. Sym. Fault-Tolerant Computing**, Newton, Mass., June 19-21, 1972, pp. 200-204.

AVI 78 A.AVIZENIS, "Fault-Tolerance: The Survival Attribute of Digital Systems", **Proc. IEEE**, October 1978, pp. 1109-1125.

BEA 78 M.D.BEAUDRY, "Performance-Related Reliability Measures for Computing Systems", **IEEE Tr. Comp.**, Vol. C-27, June 1978, pp. 540-547.

BOU 69 W.G.BOURRICIUS, W.C.CARTER, P.R.SCHNEIDER, "Reliability Modeling Techniques for Self-Repairing Computer Systems", **Proc. 12th ACM Nat. Conf.**, August 1969, pp. 295-309.

COR 75 M.CORRAZA, **Mathematical Techniques of Reliability Evaluation**, Toulouse, France: Cepadues-Edition, 1975, (in French).

COS 81 A.COSTES, J.E.DOUCET, C.LANDRAULT and J.C.LAPRIE, "SURF: A Program for Dependability Evaluation of Complex Fault-Tolerant Systems", **Proc. 11th Int. Symp. Fault-Tolerant Computing**, Portland, Maine, June 24-26, 1981, pp. 72-78.

FEN 81 T.Y.FENG, "A Survey of Interconnection Networks", **Computer**, Dec. 1981, pp. 12-27.

GAY 79 F.A.GAY and M.L.KETELSEN, "Performance Evaluation for Gracefully Degrading Systems", **Proc. 9th Int. Symp. Fault-Tolerant Computing**, Madison, Wisconsin, June 20-22, 1979, pp. 51-58.

HEL 80 B.HELVIK, "Periodic Maintenance, on the Effect of Imperfectness", **Proc. 10th Int. Symp. Fault-Tolerant Computing**, Kyoto, Japan, October 1-3, 1980, pp. 204-206.

HUS 81 R.HUSLENDE, "A Combined Evaluation of Performance and Reliability for Degradable Systems", **ACM/SIGMETRICS Conf. on Measurement and Modeling of Comp. Systems**, Las Vegas, Nevada, September 14-16, 1981, pp. 157-164.

KLE 75 L.KLEINROCK, **Queueing Systems, vol. I**,New York: Wiley , 1975.

LAP 75 J.C.LAPRIE, "Reliability and Availability of Repairable Structures", **Proc. 5th Int. Symp. Fault-Tolerant Computing**, Paris, June, 1975, pp. 87-92.

LAP 79 J.C.LAPRIE, "Dependability Modeling of Computing Systems", **IFIP Working Conf.: Reliable Computing and Fault-Tolerance in the 80's**, London, England, Sept. 26-29, 1979.

LAP 80 J.C.LAPRIE, K.MEDHAFFER-KANOUN, "Dependability Modeling of Safety Systems", **Proc. 10th Int. Symp. Fault-Tolerant Computing**, Kyoto, Japan, October 1-3, 1980, pp. 245-250.

LAP 81 J.C.LAPRIE, A.COSTES, C.LANDRAULT, "Parametric Analysis of 2-Unit Redundant Computer Systems with Corrective and Preventive Maintenance", **IEEE Trans. Reliability**, Vol. R-30, June 1981, pp.139-144.

LAP 82 J.C.LAPRIE and A.COSTES, "Dependability: A Unifying Concept for Reliable Computing", **Proc. 12th Int. Symp. Fault-Tolerant Computing**, Santa Monica, California, June 22-24, 1982, pp. 18-21.

LAR 81 F.LARBEY, "Crossbar and Transcoder Circuits: Technical Specifications", **Report N° 26.069**, October 1981, available from SINTRA, 92602 Asnières, France, (in French).

LAW 75 D.H.LAWRIE, "Access and Alignment of Data in an Array Processor", **IEEE Tr. Comp.**, Vol. C-24, December 1975, pp. 1145-1155.

MEY 80 J.F.MEYER, "On Evaluating the Performability of Degradable Computing Systems", **IEEE Tr. Comp.**, Vol. C-29, August 1980, pp. 720-731.

PAG 80 A.PAGES, M.GONDRAN, **Reliability of Systems**, Paris: Eyrolles, 1980, (in French).

PLA 79 P.PLANCKE, "Definition of a Specialized Machine for the Management of a Federation of Processors", **Report of DRET Contract N° 79/422**, 1979, available from SINTRA, 92602 Asnières, France (in French).

SHE 80 J.P.SHEN and J.P.HAYES, "Fault Tolerance of a Class of Connecting Networks", **Proc. Int. Symp. Computer Architecture**, La Baule, France, May 1980, pp. 61-71.