IEC 62429

Edition 1.0 2007-11

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Reliability growth – Stress testing for early failures in unique complex systems

Croissance de fiabilité – Essais de contraintes pour révéler les défaillances précoces d'un système complexe et unique

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub

The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:
Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

A propos de la CEI

La Commission Electrotechnique Internationale (CEI) est la première organisation mondiale qui élabore et publie des normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications CEI

Le contenu technique des publications de la CEI est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

- Catalogue des publications de la CEI: www.iec.ch/searchpub/cur_fut-f.htm

Le Catalogue en-ligne de la CEI vous permet d'effectuer des recherches en utilisant différents critères (numéro de référence, texte, comité d'études,…). Il donne aussi des informations sur les projets et les publications retirées ou remplacées.

- Just Published CEI: www.iec.ch/online_news/justpub

Restez informé sur les nouvelles publications de la CEI. Just Published détaille deux fois par mois les nouvelles publications parues. Disponible en-ligne et aussi par email.

- Electropedia: www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International en ligne.

- Service Clients: www.iec.ch/webstore/custserv/custserv_entry-f.htm

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions, visitez le FAQ du Service clients ou contactez-nous:
Email: csc@iec.ch
Tél.: +41 22 919 02 11
Fax: +41 22 919 03 00

IEC 62429

Edition 1.0 2007-11

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Reliability growth – Stress testing for early failures in unique complex systems

Croissance de fiabilité – Essais de contraintes pour révéler les défaillances précoces d'un système complexe et unique

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

PRICE CODE
CODE PRIX

W

ICS 03.120.01; 03.120.99

ISBN 2-8318-9427-1

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

RELIABILITY GROWTH – STRESS TESTING FOR EARLY FAILURES IN UNIQUE COMPLEX SYSTEMS

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62429 has been prepared by IEC technical committee 56: Dependability.

The text of this standard is based on the following documents:

FDIS	Report on voting
56/1232/FDIS	56/1249/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

RELIABILITY GROWTH – STRESS TESTING FOR EARLY FAILURES IN UNIQUE COMPLEX SYSTEMS

1 Scope

This International Standard gives guidance for reliability growth during final testing or acceptance testing of unique complex systems. It gives guidance on accelerated test conditions and criteria for stopping these tests. “Unique” means that no information exists on similar systems, and the small number of produced systems means that information deducted from the test has limited use for future production.

This standard concerns reliability growth of repairable complex systems consisting of hardware with embedded software. It can be used for describing the procedure for acceptance testing, "running-in", and to ensure that reliability of a delivered system is not compromised by coding errors, workmanship errors or manufacturing errors. It only covers the early failure period of the system life cycle and neither the constant failure period, nor the wear out failure period. It can also be used when a company wants to optimize the duration of internal production testing during manufacturing of prototypes, single systems or small series.

It is applicable mainly to large hardware/software systems, but does not cover large networks, for example telecommunications and power networks, since new parts of such systems cannot usually be isolated during the testing.

It does not cover software tested alone, but the methods can be used during testing of large embedded software programs in operational hardware, when simulated operating loads are used.

It addresses growth testing before or at delivery of a finished system. The testing can therefore take place at the manufacturer's or at the end user's premises.

If the user of a system performs reliability growth by a policy of updating hardware and software with improved versions, this standard can be used to guide the growth process.

This standard covers a wide field of applications, but is not applicable to health or safety aspects of systems.

This standard does not apply to systems that are covered by IEC 62279[39].

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60050-191:1990, *International Electrotechnical Vocabulary – Chapter 191: Dependability and quality of service*

IEC 60300-3-5, *Dependability management – Part 3-5: Application guide – Reliability test conditions and statistical test principles*

IEC 60605-2, *Equipment reliability testing – Part 2 Design of test cycles*

IEC 61163-1:2006, *Reliability stress screening – Part 1: Repairable assemblies manufactured in lots*

IEC 61163-2, *Reliability stress screening – Part 2: Electronic components*

IEC 61164, *Reliability growth – Statistical test and estimation methods*

IEC 61710, *Power law model – Goodness-of-fit and estimation methods*

3 Terms, definitions, abbreviations and symbols

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 60050-191, as well as the following, apply.

3.1.1
time compression
reducing test time by testing with higher use time than in the field

NOTE An example is testing a system that is used 8 h a day for 24 h a day.

3.1.2
accelerated test
test in which the applied stress level is chosen to exceed that stated in the reference conditions in order to shorten the time duration required to observe the stress response of the item, or to magnify the response in a given time duration

NOTE To be valid, an accelerated test should not alter the basic fault modes and failure mechanisms, or their relative prevalence.

[IEV 191-14-07]

3.1.3
(time) acceleration factor
ratio between the time durations necessary to obtain the same stated number of failures or degradations in two equal size samples, under two different sets of stress conditions involving the same failure mechanisms and fault modes and their relative prevalence.

NOTE One of the two sets of stress conditions should be a reference set.

[IEV 191-14-10]

3.1.4
execution time
time to perform a stated number of transactions

3.1.5
fault
state of an item characterized by inability to perform a required function, excluding the inability during preventive maintenance or other planned actions, or due to lack of external resources.

NOTE 1 A fault is often the result of a failure of the item itself, but may exist without prior failure.

[IEV 191-05-01]

NOTE 2 In English, the term “fault” is also used in the field of electric power systems with the meaning as given in IEV 604-02-01[42] [1]; then, the corresponding term in French is “défaut”.

NOTE 3 In this standard, the term “latent fault” is used to emphasize that the fault has not yet caused a failure.

NOTE 4 Software alone is deterministic. But this standard considers software embedded in hardware where the software can have latent faults relating to the hardware and the environment, e.g. insufficient protection against double keying, no checksum in communication, or no sanity check of input data or output data.

3.1.6
bug
popular name for a software latent fault

3.1.7
reliability indicator
non-functional parameter that points to a probable failure in a short time

3.1.8
success ratio test
test repeated a number of times of which all have to be passed without failures

3.1.9
system
set of interrelated or interacting elements

[ISO 9000:2005, 3.2.1] [41]

NOTE 1 In the context of dependability, a system will have

– a defined purpose expressed in terms of intended functions,

– stated conditions of operation/use, and

– defined boundaries.

NOTE 2 The structure of a system may be hierarchical [IEC 60300-1, 3.6] [43].

NOTE 3 For some systems, such as information technology products, data is an important part of the system elements.

[Future IEC 60300-3-15, modified] [44].

3.1.10
transaction
set of input parameters and preconditions selected from operating loads for the system

3.1.11
root cause analysis
activity to identify the cause of a fault or failure, so it can be removed by design or process changes

3.1.12
error
discrepancy between a computed, observed or measured value or condition and the true, specified or theoretically correct value or condition

NOTE 1 An error can be caused by a faulty item, e.g. a computing error made by faulty computer equipment.

NOTE 2 The French term “erreur” may also designate a mistake (see IEV 191-05-25).

[IEV 191-05-24]

1 References in square brackets refer to the bibliography.

3.1.13
mistake
human error
human action that produces an unintended result

[IEV 191-05-25]

3.1.14
failure
termination of the ability of an item to perform a required function

NOTE 1 After failure the item has a fault

NOTE 2 "Failure" is an event, as distinguished from "fault", which is a state.

NOTE 3 This concept as defined does not apply to items consisting of software only

[IEV 191-04-01]

NOTE 4 Software alone is deterministic. But this standard considers software embedded in hardware where the software can have latent faults relating to the hardware and the environment, e.g. insufficient protection against double keying, no checksum in communication, or no validity check of input data or output data.

3.1.15
failure intensity
failure intensity; instantaneous failure intensity
$z(t)$

limit, if this exists, of the ratio of the mean number of failures of a repaired item in a time interval (t, $t + \Delta t$), and the length of this interval, Δt, when the length of the time interval tends to zero

NOTE 1 The instantaneous failure intensity is expressed by the formula as

formula as

$$z(t) = \lim_{\Delta t \to 0+} \frac{E[N(t + \Delta t) - N(t)]}{\Delta t}$$

[IEV 191-12-04]

NOTE 2 To avoid confusion this standard will use “instantaneous failure intensity” since a system is repaired when it fails, and a latent fault is repaired (removed) when precipitated as a failure.

3.2 Abbreviations

CPU	Central processor unit
EMC	Electro magnetic compatibility
ESD	Electro static discharge
FMEA	Failure mode and effect analysis
MTBF	Mean operating time between failures
RAM	Random access memory

3.3 Symbols

C	total number of transactions
$D(t)$	the number of faults detected by time t
F_u	unacceptable number of failed transactions out of C transactions

i	fault number
M	probability that a system with an unacceptable reliability passes N tests without a failure
m	number of latent faults in the system
N	number of transactions to be performed without failure
p	unacceptable probability of failure per transaction
RCM $r(T_t)$	risk criterion metric for remaining latent faults at total test time T_t
r_c	the estimated number of remaining latent faults in the system
$r(T_t)$	remaining (undetected) latent faults predicted at accumulated test time T_t
s	number of test time intervals used in the Schneidewind model to estimate the model parameters
t	actual test time
t_{status}	test time at status
$T_{D(t)}$	the accumulated test time by which $D(t)$ faults were detected
T_i	the accumulated test time when fault i was detected T_{min}
T_{min}	the minimum test time that shall be accumulated by the system for 0 failures
T_t	accumulated test time measured in time units of the Schneidewind model
z	the acceptable instantaneous failure intensity
z_i	the instantaneous failure intensity of fault i
θ_i	cumulative mean operating time between failures (MTBF) when fault i was detected
	NOTE The term "cumulative MTBF" is used to be in line with other reliability growth models described in the literature. It is instructive in displaying a growth in reliability due to defect root cause elimination. The cumulative MTBF (θ_i) for each fault i is determined as $\theta_i = T_i / i$.
α	empirical constant in the Schneidewind model – failure intensity at test time = 0
β	empirical constant in the Schneidewind model – proportionality constant for failure intensity over time – Unit: $(\text{time})^{-1}$
δ	the probability of no failure occurring by T_{min} for a given acceptable instantaneous failure intensity

4 General

This standard is one of a series of standards under the application guide IEC 61014 [34].

This standard applies to large hardware-software systems when tested using a simulated operating load. Therefore, it is not known during the test if a failure is caused by hardware, software, operating load, or a combination of these. A failure may be caused by a hardware failure, e.g. a random access memory (RAM) failure, a change of timing causing data collision, or an electromagnetic disturbance, changing data transmitted. The failure may also be caused by a software latent fault or by illegal data. How the failed item is repaired or the software is changed is, for this standard, only relevant to the extent that it influences the test decisions, e.g. through the assumptions of the statistical model.

Nearly all modern systems contain embedded software. The software is typically tested on development hardware using transactions derived from the system specifications. Often the software is finished late so that the time for testing the software in the actual hardware is limited. It is usually not acceptable that the customer is the first to operate the software in the real hardware. Therefore, there is a need for a standard to guide testing and reliability growth of hardware with the embedded software.

With hardware, it is assumed that early failures are caused by a latent fault in the hardware. Depending on the stress type and stress level, these latent faults can be precipitated into permanent or intermittent failures after some time. An example could be a crack in a component. Under dry operating conditions without vibration or shocks, the latent fault may remain a latent fault. But under moist operating conditions, moisture and contaminants may penetrate the crack and cause corrosion, ending in a permanent fault. Similarly, vibration or shock can cause crack propagation that may cause a permanent fault after some time.

Software alone is deterministic. This means that a latent fault in the software (commonly called a software bug) will not result in a failure until the part of the code containing the latent fault is activated. The moment when this occurs depend on the operating conditions (e.g. input parameters and the internal states of the program, e.g. memory content). Therefore, there is a similarity between hardware latent faults and software latent faults. The software latent fault, once activated, may cause a permanent fault but will often only cause an intermittent failure.

Logical failures are systematic (i.e. they can be reproduced at will once the trigger for the associated fault is known). Since the trigger for any latent fault is encountered at random in the operating environment of the system, logical failures are observed as a stochastic process. Therefore, the usual measures of reliability can be applied (probability of time to next failure, failure intensity, etc.) Reliability growth will normally occur as latent faults are removed.

In this standard the term "latent fault" will therefore be used to cover weaknesses in hardware as well as bugs in software [10].

A failure caused by a combination of hardware and software could be, for example, that a hardware latent fault causes insufficient cooling of a component. The temperature rise changes the time delays in the circuit, causing data collision that results in a software failure. Another combination could be that a hardware design error causes insufficient shielding of signal wires. The increased level of electromagnetic noise corrupts the data in the signal wires causing a software failure, given that the software does not have an error correction feature, and the operating environment has a high electromagnetic noise level.

This standard covers repairable systems that are produced in a very small number of copies, so that experience from tests of previous similar systems is limited or non-existent. It can be used when a manufacturer wants to optimize the duration of internal acceptance testing and running-in. It addresses growth testing before or at delivery of a finished system. The testing can therefore take place at the manufacturer's or at the end user's premises. It can also be used when a company wants to optimize the duration of final production testing during manufacturing of single items, small series or during testing of a prototype.

It can also be used by the owner of only one, or a few, large systems to improve those systems only. If the user of a system performs reliability growth by a policy of updating hardware and software with improved versions, this standard can be used to control the growth process.

This standard does not cover software alone, but it can be used when embedded software is tested in a hardware system using test strategies that give a diminishing number of failures as a function of test time, for example a software test with simulated operational load. The methods described are well suited to test and improve the robustness of a software program against transients and disturbances caused by the operational load and by the hardware

system. It addresses large hardware/software systems, but does not cover large networks, for example, telecommunications and power networks, since the new parts of these are difficult to isolate during the testing process.

Reliability growth is a method aimed at improving quality by identifying and removing latent faults, but should not be used as the primary means of achieving the intended quality and reliability of the systems produced. Large systems are often produced in a small number of copies. Often only one or a few systems are produced. The remaining latent faults introduced through the design and manufacturing processes therefore shall be identified via growth testing of the finished system. However, an appropriate process control should be used and preventive methods such as an FMEA process (see IEC 60812) [33], fault tree analysis (see IEC 61025 [35]) and design reviews (see IEC 61160 [37]) should be used to reduce the number of latent faults in the produced system(s). Further, the manufacturing processes and assembly processes should be controlled, for example using statistical process control.

In some cases, it may be possible to divide a large system into a number of similar modules on which the methods of IEC 61163-1 can be used. The similar modules are then regarded as a lot consisting of similar items. This will cover latent faults in the modules but not failures caused by the interaction of the modules and interactions between the modules and the embedded software.

The failures caused by the interaction between the modules can be found only by growth testing the finished system. In modern systems, many failures are caused by an interaction between hardware and software. These failures cannot be found before the whole system is finished and functional. When the prototype is the only system produced, prototype testing and growth testing merge into one activity.

This standard covers only the early failure period of the system life cycle. This means that it does not cover the random failure period or the wear out failure period of the bathtub curve, as illustrated in Figure 1.

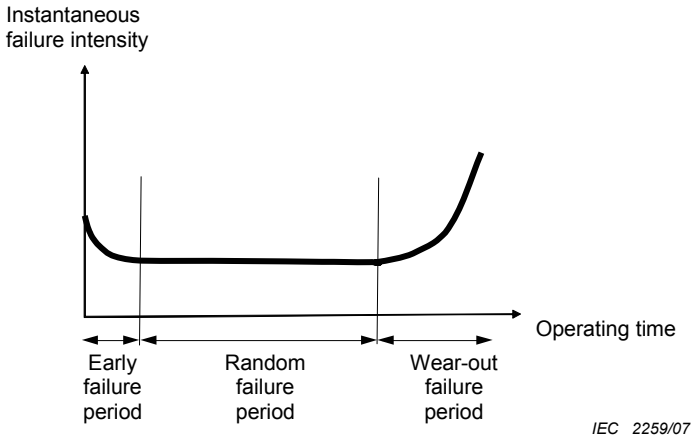

Figure 1 – The bathtub curve

NOTE This standard applies to the early failure period. Due to increased stress or time compression, this part of the operating time may be covered by a shorter period of growth testing.

When planning a reliability growth testing process, the decision makers should carefully consider time and cost against the performance of the system including the risks and costs associated with early failures in the system after delivery. All failures identified during testing shall be carefully analysed in order to find the root cause, and to ensure that the experiences are used to prevent similar problems in other systems. The finished system(s) shall be repaired or updated, re-tested for normal operation, and the system documentation shall be updated as appropriate.

If discrepancies arise between this standard and the relevant contract or specification(s), the latter shall apply.

5 Planning and performing a reliability growth test

5.1 Step 1 – Should a reliability growth test be used?

A reliability growth test is relevant in the following cases:

- the savings in costs due to reduction of early failures is larger than the cost of the test including the necessary monitoring and test equipment;
- where no previous test data exist for the whole system, since only one or a few systems have been produced, or only one system requires testing;
- where early failures are expected due to latent faults introduced in the assembly processes and the components or due to tolerance interference between components in the system;
- where relevant early failures in modules and components should be screened out by reliability stress screening before the start of the system test (see IEC 61163-1 and IEC 61163-2);
- where early failures are expected due to interaction between the hardware of the system and the embedded software;
- when using a test strategy where reliability growth is expected, i.e. the failure intensity should decrease with test time;
- when tests are performed using simulated operating loads, when possible higher than average loads can be used, and where relevant abnormal loads (noisy data, illegal data or overload conditions) can be added; or
- where possible hardware latent faults are precipitated into permanent or intermittent failures by increasing environmental stresses, i.e. by increasing temperature, temperature changes, vibration, shock, etc.

5.2 Step 2 – Failure definitions and data collection

A practical approach is to list the system requirements and check which requirements should be monitored. Then determine how the system can be monitored during the test. The test specification shall define relevant and non-relevant failures.

Relevant failures are sudden failures (function missing) as well as gradual failures (degradation). Further software related failures, i.e. no answer, wrong answer, system locked or excessive response time, should be defined. The failures may be caused by hardware, the embedded software or the interaction between the hardware and the software, e.g. shift in time delays causing data collision or electromagnetic noise changing data.

Non-relevant failures are failures caused by the test equipment, the monitoring equipment or by the test operators. If robustness testing of the system against human errors (mistakes made by the operator) is to be included in the growth test, these errors shall be defined as relevant failures.

If possible, the system should be monitored continuously for function and performance. To the extent that this is not possible, a functional test, including check of function of redundant units, should be made at fixed intervals. When stress cycles are used, the system should be checked for function after each cycle. The status of redundancy and automatic reconfiguration as well as other relevant internal system parameters should be monitored during the testing.

System changes such as replacing a module or switching operating modes shall also be recorded. A practical procedure is to report all events, e.g. start, stop, failure, upgrade, change of configuration, i.e. operating mode, etc., in the test protocol. It is recommended to

invite the test team and user operators to comment and make suggestions on the operation of the system.

For methods 1, 2, 4, 5 and 6, the test time to failure shall be registered. The time reference shall be defined. It can, for example, be test time in hours or minutes, operating time or central processor unit time (CPU time). To reduce test time, time compression or increased stresses (accelerated testing) can be used. For method 3, the number of transactions to failure shall be registered.

5.3 Step 3 – Stress levels

5.3.1 General

A detailed testing procedure shall be made before the reliability growth process starts. This plan shall list the method(s) used for the testing as well as decision procedures and confidence levels. The failure analysis and reporting procedures should also be described. The processes should be tailored to the specific system as well as to the available stress equipment, and the possible means of stressing the system (see IEC 61163-1 for guidance).

In order to precipitate the latent faults as failures as fast as possible, the systems under test should be stressed in a manner that is appropriate for the appearance of relevant failures without introducing failure modes unrelated to field failures, and without reducing the lifetime of the system significantly, i.e. wearing out solder joints or life limited components. The test conditions may lie beyond the specified operating conditions but shall still be kept within design capabilities. The purpose is to prevent system damage and avoid introducing failures that would not occur in the field.

The size of most large systems limit the stress that can be applied. Therefore low acceleration factors are usually used. Since the tests look for early failures, this is seldom a problem. Time compression only accelerates the failure modes influenced by the increased stress(es). The consequence may be that some failure modes, e.g. corrosion, are not accelerated or are even reduced. In most cases, however, this is less of a problem since the tests are looking for early failures and not wear out failures.

Increased stress is used in this test to precipitate latent faults as failures faster than in the field. For the methods that are based directly on diminishing return of the test time, e.g. methods 1.2, 2, 3, 6 and 7, there is no need to estimate the acceleration factor. For methods 1.1, 4 and 5, the acceleration factor needs to be estimated if the reliability target is specified for operation in the field. Methods to estimate the acceleration factor can be found in IEC 61163-2.

5.3.2 Increased operating load

The stress type that is most easy to increase is usually the operating load. Operating and usage profiles should be the basis for defining the operating load during the test. A very useful method is time compression, e.g. increasing the number of operating loads per time unit. In this case the acceleration factor on the operating load can easily be estimated as the ratio between the transactions in the test over the transactions in the field during the same time period.

For software, the operating load can often be increased by using real or simulated input data with a higher occurrence or volume than in normal operation. It should be decided if the operating load should simulate normal operating loads or also include unusual operating conditions, e.g. unbalanced load, load surge or extreme operating conditions such as illegal, noisy or corrupted data.

Normally the highest specified operational load should be used. In a contract situation, the parties may agree that the load can be increased above the specified maximum load. Outside a contract situation, the load shall not be increased above the specification limit except based on a management decision.

In the case of redundant or protective devices which are normally not in operation in a system, conditions should be created for activating these devices at regular time intervals.

5.3.3 Increased environmental stress

5.3.3.1 General

In principle, the stress types described in IEC 61163-1 may be used for small systems. For large systems, the possible stress types are restricted by the limitations caused by their large size, e.g. the system may be too large to fit into a climatic chamber or on vibration test equipment. Certain parts of the system may be inaccessible when the system is assembled and in operation. Furthermore, the presence of operating personnel may reduce the possibilities for increasing the stress level, for example the ambient temperature.

Indirect methods, for example reliability indicator testing (see 5.5.8 and IEC 60706-5 [32]), should be considered as a supplement to, or as a replacement for, increased stresses (see also [3]).

Stress cycles can be designed using IEC 60605-2.

The test plan shall list the chosen stress types as well as the stress levels and their duration. Reduction of lifetime for life-limited items due to the test shall be estimated when relevant.

5.3.3.2 Thermal stress

The operating temperature of the system can often be increased by raising the temperature in the room or by restricting the cooling (i.e. cover inlets or outlets, or by reducing speed of fans). The flow rate of cooling air or cooling water flow can be decreased. Furthermore, the temperature can be cycled (thermal cycling). Temperature cycling should include a cold start as this will often cause maximum thermal gradients in the system.

5.3.3.3 Moisture level

Corrosion testing is usually conducted on component level, but high relative humidity may cause increased leakage currents.

Electrostatic discharge (ESD) is usually a separate test, but low relative humidity may cause ESD discharge from persons or from movable parts. Therefore, it may in some cases be relevant to increase or decrease the relative humidity for the system or part of the system during the test.

5.3.3.4 Mechanical stress

Mechanical vibrations can be introduced by using vibration equipment or a pneumatic hammer on the chassis of the system[1].

5.3.3.5 Voltage and electrical transients

Voltage from power supplies can be increased or decreased as relevant. Transients can be introduced to the voltage supply and to signal cables (see IEC 60605-2).

5.4 Step 4 – Failure analysis and classification of failures

5.4.1 General

When a failure is observed, the first action shall be to note the test time or number of transactions to failure. Thereafter it shall be decided if the system has to be stopped, if it is not already stopped by the failure. It can be necessary to stop the operation of the system for the following reasons:

- for safety reasons;
- in order for the failure not to cause secondary failures, destroying the system or part of the system;
- in order to conduct a failure analysis; or
- in order to repair the failed item.

As soon as evidence for the failure classification has been collected, it should be decided if the failed item should be repaired immediately, or if the repair should be postponed. In some cases it may be possible to continue the testing without repairing the failed item. The condition is that an analysis shows that it is probable that the failure will not cause secondary failures and that it will still be possible to test the major part of the remaining system. This judgement will require engineering knowledge of the system. In the test protocol, it should be recorded if part of the system does not function or is not monitored due to non-repaired faults.

If it is decided to postpone the repair, the influence on the continued test shall be considered and documented, e.g. that a part of the system will not be operating, or redundancy may be reduced in further testing. For hardware, the failed item can usually be repaired by changing a module, a component or by making an adjustment. Exchanged modules and components shall be kept for later failure analysis. As soon as possible, a thorough root cause analysis [27], [28] of each failure shall be made in order to implement corrective actions to the system (component changes, process and/or design changes).

For software-related failures, the failure mode will often be removed by changing the code. Usually these changes are introduced as a new software version, but it may often be possible to continue the testing as follows:

- register the failure, and test time to each failure, but not stopping the test when the failure occurs again;
- register the failure and test time to each failure, and resetting the software when the failure occurs again; or
- register the failure, and test time to each failure, and inserting a patch in the software that neutralizes the failure when it occurs. This may cause part of the software to be non-operating during the rest of the test, and shall be documented in the test report. Where possible, a counter should be added to register the test time each instance the failure reoccurs.

NOTE Verifying that the changes made actually removes the fault, and activities when this is not the case, is described in 5.6.

When continuing the testing, failures observed elsewhere in the system should be taken into account. The reconfiguration and redundancy in the system should be considered. This can make it possible for the test to continue, but it may also mean that the observed failure may be caused by a fault somewhere else in the system. Furthermore, the possibility that a local fault, due to redundancy and automatic reconfiguration, does not cause a failure of the system should be considered. Therefore, the status of redundancy and automatic reconfiguration as well as other relevant internal system parameters should be monitored during the testing.

The failure documentation should follow the guidance given in IEC 60300-3-5. All failures shall be classified as relevant or non-relevant. Only relevant failures shall be used in the estimates and decisions.

5.4.2 Relevant failures

Relevant failures are caused by weak components, inadequate design, inadequate manufacturing processes, software latent faults or interactions between hardware and software. If the robustness of the system against mistakes made by the operator is to be included in the test, such errors shall be defined as relevant failures.

5.4.3 Non-relevant failures

Non-relevant failures are typically caused by human errors (mistakes), maintenance errors or by the test equipment. It should be considered whether non-relevant failures could be avoided by changing the procedures, e.g. for operation and maintenance.

5.5 Step 5 – Stop criteria

5.5.1 General

Since the test is based on reliability growth, i.e. a diminishing number of failures per test hour, it is important to have criteria for stopping the test. Such criteria may be:

a) fixed testing programs (5.5.2, methods 1.1 and 1.2);

b) graphical analysis (5.5.3, method 2);

c) success ratio test (5.5.4, method 3);

d) estimation of reliability (5.5.5, method 4);

e) comparing with acceptable instantaneous failure intensity (5.5.6, method 5);

f) estimation of remaining latent faults (5.5.7, method 6); or

g) reliability indicator testing (5.5.8, method 7).

When no failure has been observed for some predetermined time, and each time a failure has been identified, a decision shall be made whether to stop or continue the test. For methods 2, 5 and 6, where the stop decision depends on diminishing return, each failure mode is only included in the stop criteria the first time it occurs. But all failures shall be recorded, with the time to failure, so they can be reported and included in the updated reliability estimate, especially if the change (repair) does not effectively remove that particular fault in the future operation of the system (see 5.6).

Methods 1.1, 1.2, 2, 4, 5 and 7 are useful for systems where hardware-related failures are expected to dominate, and methods 3 and 6 are recommended where software-related failures are expected to dominate.

Methods 1.2, 2, 3, 6 and 7 do not depend on the acceleration factor used in the test, since the methods are based directly on diminishing return. For methods 1.1, 4 and 5, an estimate of the acceleration factor is needed if the users want to base the test result on a reliability target specified under field conditions (see 5.3.1).

When repairs are postponed please refer to 5.6.

Decision procedures and confidence levels are discussed in 5.5.2 to 5.5.8 for the different methods.

5.5.2 Method 1 – Fixed testing programs

5.5.2.1 Method 1.1: Fixed number of test cycles

In this method, the testing is stopped after the predetermined number of test cycles. The test cycles can be designed using the method described in IEC 60605-2. The number of cycles can be determined in one of the following two ways:

- based on previous experience (e.g. from the field), the duration of the early failure period in operating time is estimated. For the cycle developed according to IEC 60605-2, the number of operating hours equivalent to one test cycle is known. The required number of cycles can be found by dividing the duration of the early failure period in hours by the number of equivalent operating hours for one test cycle. The advantage is that the method is very simple, but it requires knowledge of the duration of the early failure period from similar previous systems; or

- in publications such as [2], curves are published for the efficiency of different stress methods. The number of cycles can be determined, taking into account the diminishing return of one extra stress cycle. The advantage is that no previous knowledge of the duration of the early failure period is required. The disadvantage is that the curves are of a general nature and can only give rough guidance for a particular system and stress conditions.

5.5.2.2 Method 1.2: Fixed number of failure-free test cycles

In this method, the test is stopped after a predetermined number of test cycles (see IEC 60605-2), but the last cycle, or a specified number of cycles, shall be without a failure. In the case of a relevant failure, the testing is therefore prolonged until the required number of failure-free cycles has been passed. The failure-free period can be computed using the methods of IEC 61163-1, but this requires some assumptions or previous knowledge of the times to failure of the weak components as described in Annex J of IEC 61163-1. The advantage is that the failure-free cycle(s) gives increased confidence that the early failures in the system have been detected. The disadvantage is that the method requires previous experience with the components and modules used in the system.

5.5.3 Method 2 – Graphical analysis

In this method, the accumulated number of relevant failures observed is plotted as a function of the test time for each individual system. As soon as a relevant failure is observed, the curve is updated. The difficulty is to estimate how the curve can be expected to continue in the future (see Figure 2). When no failures have been observed for some time or at fixed status times, a decision curve is made. This decision curve contains the plotted curve with one fictitious failure at the end of the present test time or at the decision time (see Figure 3).

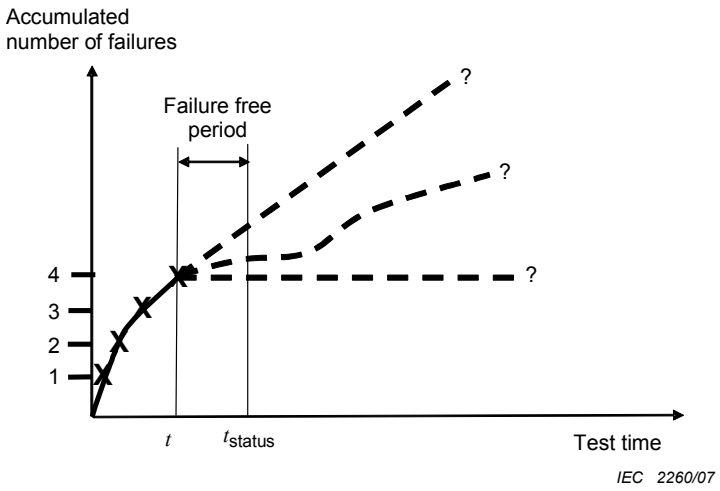

Figure 2 – Evaluating whether the cumulative failure curve has levelled out

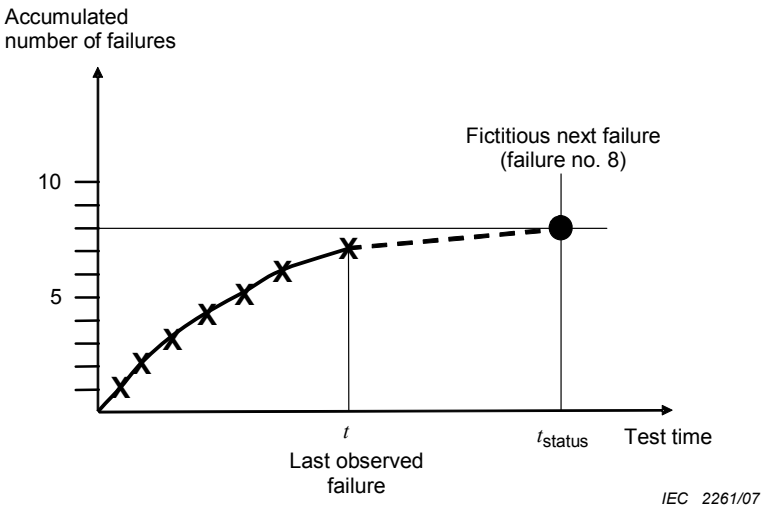

Figure 3 – Method 2

This decision curve can then be evaluated. The decision criterion is based on the cost caused by failures after delivery compared to the cost of the test.

Before the test begins, the minimum number of failures per unit of test time that justify continuation of the test is determined (e.g. 1 failure per 24 h of test time). This number can be determined based on the cost of continuing the test, the cost of correcting a failure during the growth test, or the cost of correcting a failure during operation of the system, taking into account the loss in operation caused by the failure. When the slope of the decision curve during a predetermined time (for example 24 h test time) is lower than the agreed value, the test is stopped.

If the test conditions are changed in a major way during the test, a new plot should be made based on the failures found under the new conditions only.

5.5.4 Method 3 – Success ratio test

5.5.4.1 General

The success ratio test is a special case of the failure-free period criterion. The purpose is to confirm that the failure probability is very likely to be below a specified upper boundary. A number of tests with random operating loads are made on the system. If a failure is observed, the failed item is repaired before the test is continued. The test is continued until a sequence of successive successful transactions gives sufficient confidence that the probability of failure is acceptably low. This method allows the number of transactions to be determined, given that the maximum acceptable probability of failure is specified. The method assumes that each transaction represents an independent trial from the same population, simulating real applications.

This method is appropriate for systems with embedded software. The nature of the transactions depends upon the nature of the software. The transactions shall simulate a normal but high operation load of the system [4].

5.5.4.2 Number of transactions required

In this method, simulated operating loads are used as transactions. This will ensure that the conditions for the system under test vary from transaction to transaction, i.e. initial condition, state of buffers, registers, menu path, etc. Therefore, latent faults that may be missed in a

test program structured after the requirement specifications, can be detected with this method.

Because only a very small fraction of the conceivable combinations between input parameters, output parameters and menu paths can be executed, one can never be sure that the probability of failure is small enough. However, it is possible to calculate the probability that a system with an unacceptable reliability would have passed the performed tests. The transactions are assumed to be independent:

$$M = (1 - p)^N \quad (1)$$

where M is the probability that a system with an unacceptable failure probability p would have passed the N tests successfully. Tables 1 and 2 provide some values of M for two values of p and various values of N.

If C is the total number of transactions from which the test is selected, and if it is considered unacceptable for F_u or more of those to result in an error, then the probability M that a system with an unacceptable failure probability per transaction of F_u/C would have passed the N tests successfully can be estimated as:

$$M = (1 - F_{uu}/C)^N \quad (2)$$

The system under test shall pass N transactions without failures, before the test can be stopped as passed.

Table 1 – Probability that a system with failure probability of 0,001 will pass N successive tests

p = 0,001	
N	$M = (1 - p)^N$
500	0,606 38
600	0,548 65
700	0,496 41
800	0,449 15
900	0,406 39
1 000	0,367 70
1 500	0,222 96
2 000	0,135 20
2 500	0,081 98
3 000	0,049 71
3 500	0,030 14
4 000	0,018 28
4 500	0,011 08
4 700	0,009 07
5 000	0,006 72

Table 2 – Probability that a system with failure probability of 0,000 001 will pass N successive tests

p = 0,000 001	
N	$M = (1 - p)^N$
1 000 000	0,367 88
2 000 000	0,135 34
3 000 000	0,049 79
4 000 000	0,018 32
5 000 000	0,006 74
6 000 000	0,002 48
7 000 000	0,000 91
8 000 000	0,000 34
9 000 000	0,000 12
10 000 000	0,000 05

5.5.5 Method 4 – Estimation of reliability

In this method, the statistical tools of IEC 61164 or IEC 61710 are used to estimate the reliability growth and to compare the estimate with the reliability target for the system. For further information, see IEC 61164 and IEC 61710.

5.5.6 Method 5 – Comparison with acceptable instantaneous failure intensity

5.5.6.1 Background for method

This method involves performing a reliability growth programme on a system consisting of both hardware and software. During this test, continual stress tests are applied to the system with a view to identifying weak points in the design and manufacturing process. The tests involve applying both environmental and operational stresses including extreme data traffic rates and conditions to precipitate latent faults. As latent faults in the design and manufacturing process are identified and corrected, the reliability of the system begins to grow. Due to feedback mechanisms inherent in the methodology, the ability of the processes to produce reliable systems in the future also grows. This growth is modelled by means of the reliability growth model that allows the development team to regularly monitor the reliability improvement. Finally, a mathematical stopping rule determines the optimum time for release of the system(s).

Features of the test programme include:

- a design process that is fully controlled such that it is systematic, predictable, reliable and follows defined methodologies and procedures;
- all system components/materials are procured from approved vendors;
- all system units are manufactured via the final process;
- an acceptable instantaneous failure intensity under the test conditions is chosen for the system. This represents a predefined instantaneous failure intensity that the system should attain before testing is terminated and the system is released to the customer. An estimated acceleration factor may be used to transfer the instantaneous failure intensity from use conditions to test conditions (see 5.3.1 and 5.5.1);
- this acceptable instantaneous failure intensity is then used to determine a minimum test time based on the probability of not encountering any failure during the test;
- stress test commences as soon as the system is available and continues until the final system is ready for shipment;
- the test involves the application of environmental and operational stresses including extreme data traffic rates and conditions to identify weak points in new designs. This includes testing the software for robustness when specified;
- as changes are made to the design, these are incorporated into the systems undergoing test;
- the cumulative test hours are recorded for each system undergoing test; and
- the test monitoring ensure that all failures of the system that become visible in the output are detected.

As latent faults are identified and corrected in the design, components and the manufacturing process, the system's reliability begins to grow. References for this methodology are contained in [11], [12] and [13].

5.5.6.2 The reliability growth model

The reliability growth model involves plotting the cumulative MTBF (θ_t) on the *y*-axis and the square root of the cumulative test time (T_i) on the *x*-axis. As failures occur, the reliability growth curve is updated. Reliability improvements will be observed by an increasing vertical slope while a deterioration in reliability will be observed as a decreasing slope. Unlike other graphical models, the influence of early failures on the overall growth model is minimized as the test time increases due to the scale on the *x*-axis expanding as the test progresses. This reliability growth model allows the development team to monitor and display the reliability improvements at status meetings. It is not the intention to use the reliability growth model to quantify or extrapolate any reliability data or statistics, but rather to act as a graphical tool to display continual reliability test progress to the development team. Consequently, it is

acceptable to combine data from the various stress and operational tests. An example of the reliability growth model is shown in Annex B.

5.5.6.3 The stopping rule

The stopping rule allows one to assess when the system has achieved a predefined level of reliability that allows testing to terminate and delivery of the system to proceed. As each failure is encountered, the stopping time is recalculated in order to achieve the minimum predefined level of reliability. All types of latent faults can be accommodated within this stopping rule, and it is applicable to the total system incorporating both hardware and software.

Each latent fault is assumed to have a latent fault instantaneous failure intensity. The latent faults with the highest failure intensity are expected to be precipitated first. At a certain point, all latent faults have been precipitated, or the remaining latent faults have a failure intensity lower than the acceptable instantaneous failure intensity z.

The stopping rule uses the concept of a latent fault instantaneous failure intensity, whereby a system has an unknown number of latent faults present, each of which has its own fault instantaneous failure intensity. A system that initially contained m latent faults will observe a decreasing instantaneous failure intensity as the number of latent faults is reduced from m due to detection and correction. If faults are not corrected, or if a failing component is replaced with one containing the same latent fault, then clearly that fault and its associated instantaneous failure intensity remains within the system (see 5.6). In the following, the latent fault instantaneous failure intensity will, for short, just be called instantaneous failure intensity.

An example of this method is shown in Annex B.

The assumptions of the stopping rule can be stated as follows:

- the system has an unknown number of latent faults m;
- each latent fault i (i = 1,…,m) is independent and has an associated instantaneous failure intensity z_i which occurs in accordance with a Poisson process; and
- when a failure occurs, its cause is investigated and the fault is found and corrected. During correction of one fault, no other latent fault is introduced.

The stopping rule advocates a minimum test time (T_{min}) to avoid terminating the test too early (see [11], [12] and [13]):

$$T_{\text{min}} = -\frac{\ln \delta}{z} \tag{3}$$

where z is the acceptable instantaneous failure intensity and δ represents the probability of no failure occurring by T_{min}. It is preferable to have δ small as there is no wish to terminate the test prior to encountering failures. δ = 0,05 indicates that there is only a 5 % chance that T_{min} will have been reached without encountering any failure. This value of δ is generally recommended.

Once the minimum test time is accumulated, the stopping rule shown below indicates that testing shall terminate at the earliest time t such that:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2 \left(1 - e^{-t/T_i}\right)^2}} \leq z \tag{4}$$

The acceptable instantaneous failure intensity, z represents a predefined instantaneous failure intensity that the system should attain before testing is terminated and the system is released to the customer. The $1/\left(t - T_{D(t)}\right)$ term in the stopping rule equation represents the point estimate of the system instantaneous failure intensity in the period since the last failure. The stopping rule is also quite robust to the time of failure occurrence and it is not necessarily adversely affected by a small number of failures, provided of course that they do not occur at the end of the test.

The test results are meaningful only so far as the demand profile of the intended application is equal to the demand profile of the test. In case of deviating between these, the results shall be re-calculated to the new profile. This is the reason why it is recommended to use simulated operating loads during the test.

5.5.7 Method 6 – Estimation of remaining latent faults

In this method, the parameters for a statistical model of the failures of the hardware-software system as a function of test time are estimated. Thereafter, the remaining latent faults in the system can be estimated as well as the estimated test time to remove the remaining latent faults (see [5], [6], [7] and [8]).

An appropriate statistical model is fitted to the observed number of failures as a function of test time. The statistical model should be estimated under the specific test conditions (e.g. stress level) and the specific system. This has the advantage that the probability of a specified number of remaining latent faults (e.g. less than one latent fault) can be estimated. The disadvantage is a more complicated procedure that includes the fitting of a statistical model. In some cases, the statistical model does not describe the data sufficiently well, or the estimation of the parameters of the model does not converge. In such cases, other statistical models from the literature may be used (see BS 5760-8 [40] and [5])), provided their suitability is justified. However, no model will fit the data exactly, so it will often be necessary to use the model that fits the data best based on a goodness-of-fit test (see [9]). In the example in Annex C, the model of Dr. Schneidewind is used, but there exist several software failure models, for example Goel and Okumoto ([22] and [23]), Jelinski and Moranda ([24]), Rushforth, Staffanson and Crawford ([25]), and Langberg and Singpurwallah ([26]).

The steps for using the method are as follows:

a) starting at the beginning of the test, record test time to failure for each failure;

b) select an appropriate statistical failure model that allows estimation of the number of remaining latent faults (see [40], [5], [6], [7], [8] and the references listed above). Some models also give a prediction of remaining test time (see [40]);

c) estimate the parameters of the model selected in step b), based on the observed failures;

d) determine the stop criterion for the test expressed as probability of less than r_c remaining latent faults in the system (r_c may be 1);

e) compute the probability of the remaining latent faults in the system being lower than r_c ; and

f) continue the test until the probability is below the specified r_c.

For an example of the method, see Annex C.

5.5.8 Method 7 – Reliability indicator testing

Reliability indicator testing can be used to detect latent faults before they have precipitated into failures. A reliability indicator is a parameter that is not one of the functional parameters of the system, but which may be monitored during the test process (see also IEC 60706-5).

Examples of such parameters are electrical noise measurements, e.g.1/f noise measurement (see [3]) or temperature increase monitored by an infrared camera [3]. Furthermore,

mechanical noise or power consumption may be used. For software, response time may be used.

In modern microprocessor systems, a number of self-test features or boundary scan features can often be built-in and monitored in order to verify the health of the system.

The use of such reliability indicators should be agreed in the contract and specified in the test plan. All indications by the reliability indicators should be investigated thoroughly and if a latent fault is found, a root cause analysis should be made. When no latent fault is found, the operation of the system should, if possible, be followed for an appropriate time in order to see if a failure that develops later was indicated by the deviating indicator value.

The advantage of the indicator testing method is that it can detect latent faults before they develop into failures. Furthermore, it can be easier to monitor secondary parameters than primary (functional) parameters. The disadvantage is that the indicator test can detect only specific failure modes, so that either a large number of indicators has to be used or the system is screened for only one or a few major failure modes. More research is needed in reliability indicators before they can be widely used in industry. However, they are very promising for testing and preventive maintenance (see IEC 60706-5 and [17] and [18].

All correct as well as incorrect indications shall be logged and a matrix produced to summarize the correct indications as well as the number of types of error for each reliability indicator used (see Table 3). The percentage of observations in each cell of Table 3 shall be recorded.

In order to be efficient, each reliability indicator should point to an area or some specific component where the potential problem is found. Selecting reliability indicators therefore requires engineering knowledge of the system as well as a good knowledge of failure modes, their development, and early indications.

Table 3 – Correct and incorrect decisions using reliability indicators

	Reliability indicator testing result	
	Reliability indicator indicates latent fault present	**Reliability indicator indicates no latent fault present**
The latent fault was real	Correct decision	Indicator not sensitive enough
The latent fault could not be found	Indicator too sensitive	Correct decision

5.6 Step 6 – Verification of repairs and reliability growth

When, upon a failure occurrence, changes are made in the system to remove latent faults and improve the system, the result of those change shall be evaluated in subsequent or extended testing for their effectiveness and to make sure that they have not introduced yet another failure mode, not experienced previously. The extent of additional test time will depend on the nature of those changes. The changes may also be verified by simulation or by a separate test specifically prepared to address the failure mode which is being mitigated, for example a specific accelerated test. Such a test will often not be able to detect possible interaction of the changed part of the system with other part of the system. Therefore the test on system level will often have to be extended, especially when the repairs are postponed so that a number of changes are made to the system at the same time, e.g. new hardware version or new software version.

Based on the verification of the effectiveness of the changes, the estimated reliability growth shall be updated. Any observed failure modes that were not removed by the system changes, the number of their repetitions in test as well as any additional failures attributed to the change shall be included in this estimate.

5.7 Step 7 – Reporting and feedback

The final report should contain the following information as relevant:

– system description including revision of hardware and software;
– tailoring to specific contract and system conditions (e.g. reconfiguration and redundancy);
– monitored parameters and failure definitions (see 5.2);
– operating and usage profiles – operating loads during test (see 5.3);
– testing conditions and equipment (see 5.3), stress types, stress level, stress duration and stress cycles (see IEC 60605-2 and IEC 61163-1);
– reduction of lifetime for life-limited items;
– time to failures and failure classification (see 5.4);
– failure analysis procedures to find root causes (see 5.4);
– stop criteria and confidence levels (see 5.5);
– termination of testing (see 5.5);
– repairs and changes made during the test;
– repeated growth test(s) when relevant;
– changes made during the test (new revision of hardware and software) and how the system was repaired or updated and re-tested for normal operation;
– changes made after the test – the system documentation should be updated as appropriate; and
– conclusion of the test – achieved reliability where possible with confidence level.

The resulting reliability growth can be stated as follows:

Method 1.1: The fixed test program was passed without failures/with the following failures;

Method 1.2: The last cycle of the fixed test program was passed without failures;

Method 2: No failures were observed from test time A until test time B (the report shall give the numbers A and B – see Figure 3);

Method 3: N transactions were performed without failures. The probability that the system has a failure intensity of F_u/C is M or lower (see 5.5.4.2);

Method 4: The MTBF of the system after the reliability growth test is estimated to be (number to be stated in report);

Method 5: The instantaneous failure intensity after the reliability growth is estimated to be $\leq z$ (see Equation (4));

Method 6: The estimated number of remaining failures are $<C$; or

Method 7: No abnormal levels of the following reliability indicators were observed during the test: (the report shall list the failure indicators used).

Annex A
(informative)

Practical example of method 3 – Success ratio test

A system containing a “batch type” embedded software program shall be tested.

Experience shows that the test time will be determined by the time it takes to identify the relevant latent faults in the software.

For testing, 25 000 transactions have been recorded from the real operating load of the previous system. They are deemed to cover typical operating cases when the system is used in practice.

It is decided that the system can be transferred from the test to normal operation if the probability of more than five of these transactions causing failure is less than or equal to 10 % (i.e. F_u = 5).

The number of transactions needing to be run during the test is determined by the equation:

$$M = (1 - F_u/C)^N \quad \text{(A.1)}$$

or

$0{,}10 = (1 - 5/25\,000)^N = (0{,}999\,8)^N$

N = 11 500 transactions

Therefore 11 500 of the 25 000 transactions need to be run selected randomly.

If no failure is detected, the test is passed and can be stopped.

Annex B
(informative)

Practical example of method 5 – Comparison with acceptable instantaneous failure intensity

B.1 Reliability growth plot

An example of the method is shown using the data contained in Table B.1.The cumulative test time to failure for each fault is represented in column 2. After each failure observed, this information is then transformed in columns 4 and 5 and used to develop the *y*- and *x*-axes respectively of the reliability growth plot, as well as determining the stopping time (column 5). The resultant reliability growth plot is shown in Figure B.1. It can be observed from Table B.1 and Figure B.1 that sustained reliability growth occurs after approximately 42 000 cumulative test minutes (20th failure).

NOTE In software testing test time is often measured in minutes.

Table B.1 – Reliability growth and stopping times for the practical example

Fault number i	**Cumulative time to failure** min T_i	**Cumulative MTBF** min $\theta_i = \frac{T_i}{i}$	$\sqrt{T_i}$ min ^{0,5}	**Stopping time** min t
1	1	1	1	10 010
2	60	30	8	10 070
3	8 230	2 743	91	31 130
4	8 300	2 075	91	35 030
5	8 350	1 670	91	37 570
6	12 568	2 095	112	42 510
7	15 556	2 222	125	47 260
8	19 876	2 485	141	52 590
9	19 900	2 211	141	56 150
10	19 910	1 991	141	59 220
11	27 200	2 473	165	64 570
12	27 210	2 268	165	67 770
13	27 700	2 131	166	70 850
14	28 660	2 047	169	73 840
15	34 450	2 297	186	78 080
16	37 400	2 338	193	81 660
17	37 410	2 201	193	84 350
18	41 250	2 292	203	87 920
19	42 000	2 211	205	90 660
20	42 100	2 105	205	93 130
21	44 020	2 096	210	95 940
22	48 600	2 209	220	99 360
23	51 600	2 243	227	102 430
24	55 100	2 296	235	105 600
25	82 100	3 284	287	117 120
26	108 300	4 165	329	133 740

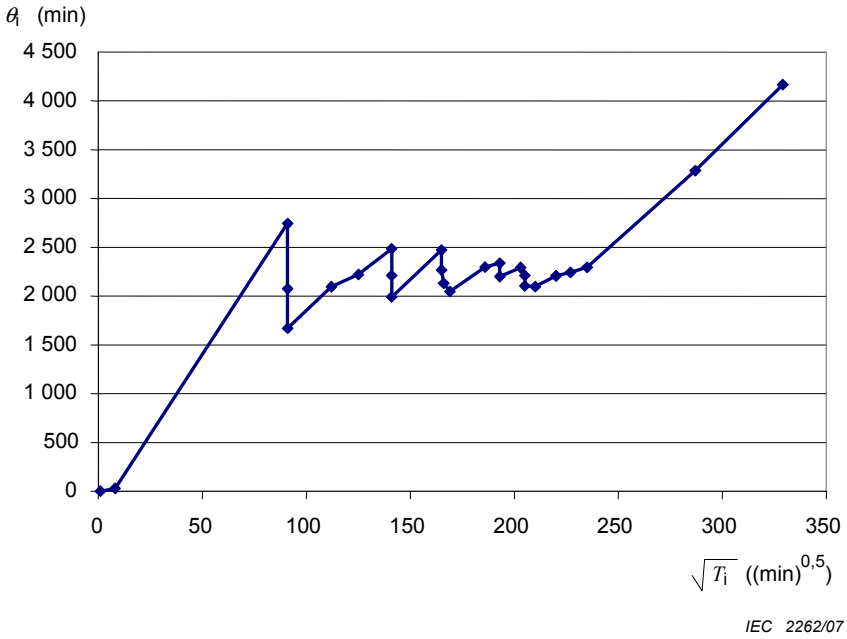

Figure B.1 – Reliability growth plot from data from Table B.1

B.2 The stopping rule

An acceptable instantaneous failure intensity was considered to be $z = 1$ x 10^{-4} failures per test minute and δ was chosen as 0,05. The minimum test time ($T_{\min}$) is calculated as follows:

$$T_{\min} = -\frac{\ln \delta}{z} = -\frac{\ln(0{,}05)}{1\times 10^{-4}} = 30\ 000 \text{ min} \qquad \text{(B.1)}$$

The test should not terminate before 30 000 min even if no failure has been encountered.

As each failure is encountered, the stopping time is recalculated.

Column 5 of Table B.1 indicates the proposed stopping times based on the stopping rule. As the first failure occurs after only 1 min $T_{D(t)}$ is in this case 1, the proposed stopping time after failure number 1 is shown below:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2\left(1-e^{-t/T_i}\right)^2}} = \frac{1}{t-1} + 3\sqrt{\frac{e^{-t/1}}{1^2\left(1-e^{-t/1}\right)^2}} \le z = 1\times 10^{-4} \qquad \text{(B.2)}$$

The smallest value of t (to the nearest 10 min) that satisfies this equation is 10 010 min. Note that this value is less than the proposed minimum test time.

The proposed stopping time for failure number 3 is 31 130 cumulative test min. The calculations are shown below:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2\left(1-e^{-t/T_i}\right)^2}} = \frac{1}{t-8\,230} + 3\sqrt{\frac{e^{-t/1}}{1^2\left(1-e^{-t/1}\right)^2} + \frac{e^{-t/60}}{60^2\left(1-e^{-t/60}\right)^2} + \frac{e^{-t/8\,230}}{8\,230^2\left(1-e^{-t/8\,230}\right)^2}} \le z = 1\times 10^{-4} \qquad \text{(B.3)}$$

The smallest value of t that satisfies this equation is 31 130 min.

This result indicates that testing shall continue until 31 130 cumulative test minutes have elapsed before testing is terminated. However, it is observed that before this time period has been reached, another defect has been detected, pushing the stopping time out to 35 030 min. Testing terminates if this stopping time has been reached without any further failures.

In this example the testing was stopped after 133 740 min (approximately 3 months of testing) since no failure was observed between 108 300 min and 133 740 min.

Annex C
(informative)

Practical example of method 6 – Estimation of remaining latent faults

C.1 Method 6 – Estimation of remaining latent faults

The system is tested and the number of failures counted in each time interval. The time is measured with the time interval as the unit, i.e. s = 3 means 3 time intervals (time units).

When five failures have been obtained, the parameters α and β of the Schneidewind software reliability model are estimated (see [5] and [6]).

Then the following steps are performed:

Step 1: Decide the required critical number of remaining latent faults remaining in the system after the test r_c (i.e. less than one latent fault). If the expected number of failures is greater than or equal to r_c, then the growth test has to continue.

Step 2: Compute:

$$\text{RCM}\, r(T_t) = \left(r(T_t) - r_c\right) / r_c = \left(r(T_t) / r_c\right) - 1 \qquad \text{(C.1)}$$

Step 3: RCM $r(T_t)$ can be plotted as a function of T_t for the selected r_c or listed as in Table C.1.

The test can be stopped as soon as the value of RCM $r(T_t)$ is below r_c.

The number of remaining latent faults in the system can be estimated using the equation:

$$r(T_t) = \left(\frac{\alpha}{\beta}\right)\exp\left(-\beta\left(T_t - (s-1)\right)\right) \qquad \text{(C.2)}$$

The expected total test time to achieve the specified number of remaining latent faults in the system can be estimated using the equation:

$$T_t = \frac{\ln\left[\alpha / \left(\beta\left[r(T_t)\right]\right)\right]}{\beta} + (s-1) \qquad \text{(C.3)}$$

EXAMPLE

After test time (test interval) s = 9, the following parameters have been estimated for the system under test:

$$\alpha = 0{,}534 \ \left[(\text{time-units})^{-1}\right] \text{ and } \beta = 0{,}061 \ \left[(\text{time-units})^{-1}\right]$$

$r(T_t)$ and the estimated remaining number of latent faults RCM $r(T_t)$ are computed for less than one remaining latent fault, i.e. critical number of remaining latent faults r_c = 1 for test time T_t =18 time units and 52 time units.

Table C.1 – Determinination for stopping test

T_t [time – units]	α $\left[(\text{time – units})^{-1}\right]$	β $\left[(\text{time – units})^{-1}\right]$	s [time – units]	$r(T_t)$ number of latent faults	RCM$r(T_t)$ number of latent faults
18	0,534	0,061	9	4,76	3,76
52	0,534	0,061	9	0,60	–0,40

The expected total test time to achieve the specified number of remaining latent faults $r(T_t)$ = 1 in the system can be estimated using the equation:

$$T_t = \frac{\ln\left[\alpha / (\beta[r(T_t)])\right]}{\beta} + (s-1) = \frac{\ln\left[0{,}534 / (0{,}061[1])\right]}{0{,}061} + (9-1) = 43{,}56 \text{ time units} \qquad \text{(C.4)}$$

C.2 Background of Schneidewind's model

Background information about the Schneidewind model can be found in [5] and [40].

Bibliography

[1] NIKOLSKY, George N. and TUSTIN, Wayne: “Using a Pneumatic Hammer for ESS”, TEST, December/January, 1988/89

[2] “Technical Guidelines for the ESS process” IEST-RP-PR001.1 Published 1/1/1999 – ISBN: 1-877862-70-3 Document number: PR 01 - USA

[3] JENSEN, Finn: “*Electronic Component Reliability*”, Wiley, 1995

[4] PARNAS, David L., VAN SCHOUWEN, John A. and SHU PO KWAN: “Evaluating of Safety Critical Software”, Communications of the ACM, June 1990, Volume 33, No.6

[5] “Recommended Practice for Software Reliability R-013-1992”, ANSI, American Standards Institute / American Institute of Aeronautics and Astronautics, Washington DC, 1993

[6] SCHNEIDEWIND, Norman F. : “Introduction to Software Reliability with Space-Shuttle Example”, RAMS Tutorial Notes, 2001

[7] SCHNEIDEWIND, Norman F. : “Reliability Modelling for Safety Critical Software”, IEEE Transactions on Reliability, Vol. 46, No.1, March 1997, pp. 88-98

[8] SCHNEIDEWIND, Norman F.: “Software Reliability Model with Optimal Selection of Failure Data”, IEEE Transactions on Software Engineering, Vol. 19, No. 11, November 1993, pp. 1095-1104

[9] LITTLEWOOD, B. and VERAL, J.L.:: ”Likelihood Function of a Debugging Model for Computer Software Reliability”, IEEE Transactions on Reliability, Vol. R-30, No. 2, June 1981

[10] Michael Pecht (Editor): ”Product Reliability, Maintainability and Supportability Handbook”, ARINC Research Corporation, CRC Press, 1995

[11] DONOVAN, J. and MURPHY, E.: “Reliability Growth – A New Graphical Model”, Quality and Reliability Engineering International, 1999, 15: pp. 167-174

[12] DONOVAN, J. and MURPHY, E.: “An Infrequently Used Stopping Rule Revisited”, Quality Engineering, 2001, 13: pp. 367-376

[13] DONOVAN, J. and MURPHY, E.: “Total System Reliability: Integrated Model for Growth and Test Termination”, Quality and Reliability Engineering International, 2005, 21: pp. 329-344

[14] MIL-HDBK-2164A, Environmental Stress Screening Process for Electronic Equipment - Revision A

[15] MIL-HDBK 344A, Environmental Stress Screening (ESS) of Electronic Equipment - Revision A

[16] Def-Stan 00-40, Reliability and Maintainability

[17] SALFNER, Felix and MALEK, Miroslaw : ”Predicting Failures of Computer Systems: A Case Study for a Telecommunication System”, IEEE Proceedings of IDPS 2006

[18] MALEK, Miroslaw : “Tutorial on Predictive Algorithms and Technologies for Availability Enhancement”, International Service Availability Symposium (ISAS 2006), Helsinki 14 May 2006

[19] ETO, Hiroyuki and DOHI, Tadashi: “Analysis of a Service Degradation Model with Preventive Rejuvenation”, International Service Availability Symposium (ISAS 2006), Helsinki 14 May 2006

[20] Re-Aéro-703-06-A: “Guide pour le Pilotage de la Croissance de Fiabilité” (Guidelines for the Reliability Growth Management) BNAe 1995/94 Ref.186

[21] CROW, L.H.: ”Reliability Analysis for Complex, Repairable Systems” AD-A020 296 - AMSAA-TR 138 (Army Material Systems Analysis Activity)December 1975

[22] GOEL, A.L. and OKUMOTO, K.: ”An Imperfect Debugging Model for Reliability and Other Quantitative Measures of Software Systems” in: Bayesian Software Prediction Models, Rome Air Development Center Report RADC-TR-78-155, Issue 5, Vol. 1, 1978

[23] GOEL, A.L. and OKUMOTO, K.:”Time Dependent Error Detection Rate Model for Software Reliability and Other Performance Measures”, IEEE Trans. Reliability, 1979 pp 206-211

[24] JELINSKI, Z. and MORANDA, P.B.: ”Software Reliability Research”, in: W. Freiberger (ed.), “Statistical Computer Performance Evaluation”, New York: Academic Press, 1972, pp. 465-484

[25] RUSHFORTH, C.K., STEFFANSON, F.L. and CRAWFORD, A.E.: ”Software Reliability Estimation under Conditions of Incomplete Information”, Rome Air Development Center Report RADCTR-79-230, 1979

[26] LANGBERG, N. and SINGPURWALLAH, N.D.: ”A Unification of Some Software Reliability Models via the Bayesian Approach”, George Washington University Technical Memo TM-66571, 1981

[27] “Root Cause Analysis Guidance Document”, DOE-NE-STD-1404-92, U.S. Department of Energy, Office of Nuclear Energy, Office of Nuclear Safety Policy and Standards, Washington D.C.20585, February 1992

[28] ROONEY, J.J. and VANDEN HEUVEL, L.N.: ”Root Cause Analysis for Beginners”, Quality Progress, July 2004

[29] IEC 60300-2, *Dependability management – Part 2: Guidelines for dependability management*

[30] IEC 60300-3-1, *Dependability management – Part 3-1: Application guide – Analysis techniques for dependability – Guide on methodology*

[31] IEC 60605-4, *Equipment reliability testing – Part 4: Statistical procedures for exponential distribution – Point estimates, confidence intervals, prediction intervals and tolerance intervals*

[32] IEC 60706-5, *Guide on maintainability of equipment – Part 5-4: Diagnostic testing*

[33] IEC 60812, Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA)

[34] IEC 61014, *Programmes for reliability growth*

[35] IEC 61025, *Fault tree analysis (FTA)*

[36] IEC 61078, *Analysis techniques for dependability – Reliability block diagram and boolean methods*

[37] IEC 61160, Design review

[38] Void.

[39] IEC 62279, Railway applications – Communications, signalling and processing systems – Software for railway control and protection systems

[40] BS 5760-8: *Reliability of systems, equipment and components. Guide to assessment of of reliability of systems containing software*

[41] ISO 9000:2005: *Quality management systems – Fundamentals and vocabulary*

[42] IEC 60050-604, *International Electrotechnical Vocabulary – Chapter 604: Generation, transmission and distribution of electricity – Operation*

[43] IEC 60300-1:2003, *Dependability management – Part 1: Dependability management systems*

[44] IEC 60300-3-15, *Dependability management – Part 3-15: Guidance to engineering of system dependability (in preparation)*

SOMMAIRE

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

CROISSANCE DE FIABILITÉ – ESSAIS DE CONTRAINTES POUR RÉVÉLER LES DÉFAILLANCES PRÉCOCES D'UN SYSTÈME COMPLEXE ET UNIQUE

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI n'a prévu aucune procédure de marquage valant indication d'approbation et n'engage pas sa responsabilité pour les équipements déclarés conformes à une de ses Publications.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

La Norme internationale CEI 62429 a été établie par le comité d'études 56 de la CEI: Sûreté de fonctionnement.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
56/1232/FDIS	56/1249/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de maintenance indiquée sur le site web de la CEI sous "http://webstore.iec.ch" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

CROISSANCE DE FIABILITÉ – ESSAIS DE CONTRAINTES POUR RÉVÉLER LES DÉFAILLANCES PRÉCOCES D'UN SYSTÈME COMPLEXE ET UNIQUE

1 Domaine d'application

La présente Norme internationale donne des recommandations applicables à la croissance de fiabilité au cours des essais finaux ou des essais d'acceptation d'un système complexe et unique. Elle donne des indications relatives aux conditions d'essais accélérés et des critères pour l'arrêt de ces essais. « Unique » signifie qu'aucune information n'existe sur des systèmes similaires, et le faible nombre de systèmes produits implique que les informations déduites des essais seront un usage limité pour une production future.

La présente norme concerne la croissance de fiabilité de systèmes complexes réparables composés de matériels incorporant des logiciels embarqués. Elle peut être utilisée pour décrire la procédure pour les essais d'acceptation, « le rodage », et pour s'assurer que la fiabilité d'un système fourni n'est pas compromise par des erreurs de codage, des erreurs de qualité d'exécution ou des erreurs de fabrication. Elle ne traite que de la période de défaillance précoce du cycle de vie du système et pas de la période de défaillance constante, ni de la période de défaillance par usure. Elle peut aussi être utilisée lorsqu'une entreprise veut optimiser la durée des essais de production interne au cours de la fabrication de prototypes, de systèmes uniques ou de petites séries.

Elle s'applique principalement aux grands systèmes matériels/logiciels, mais ne couvre pas les grands réseaux, par exemple les réseaux de télécommunications et d'alimentation, étant donné que les nouvelles parties de systèmes de ce type ne peuvent pas en général, être isolées au cours des essais.

Elle ne traite pas des logiciels soumis aux essais isolément, mais les méthodes peuvent être utilisées au cours d'essais de grands logiciels embarqués dans des matériels opérationnels, lorsque des charges en service simulées sont utilisées.

Elle traite des essais de croissance réalisés avant ou lors de la livraison d'un système fini. Les essais peuvent par conséquent avoir lieu dans les locaux du fabricant ou de l'utilisateur final.

Si l'utilisateur d'un système traite la croissance de fiabilité par une politique de mise à jour des matériels et des logiciels avec des versions améliorées, la présente norme peut être utilisée en tant que lignes directrices dans le processus de croissance.

La présente norme couvre un large champ d'applications, mais n'est pas applicable aux aspects sanitaires ou de sécurité des systèmes.

La présente norme ne s'applique pas aux systèmes couverts par la CEI 62279[39].

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI 60050-191:1990, *Vocabulaire Electrotechnique International – Chapitre 191: Sûreté de fonctionnement et qualité de service*

CEI 60300-3-5, *Gestion de la sûreté de fonctionnement – Partie 3-5: Guide d'application – Conditions des essais de fiabilité et principes des essais statistiques*

CEI 60605-2, *Essai de fiabilité des équipements – Partie 2: Conception des cycles d'essai*

CEI 61163-1:2006, *Déverminage sous contraintes – Partie 1: Assemblages réparables fabriqués en lots*

CEI 61163-2 :2006, *Déverminage sous contraintes – Partie 2: Composants électroniques*

CEI 61164, *Reliability growth – Statistical test and estimation methods* (disponible seulement en anglais)

CEI 61710, *Modèle de loi en puissance – Test d'adéquation et méthodes d'estimation des paramètres*

3 Termes, définitions, abréviations et symboles

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions donnés dans la CEI 60050-191, ainsi que les suivants, s'appliquent:

3.1.1
compression temporelle
réduction de la durée d'essai en effectuant les essais avec des durées d'utilisation plus élevée qu'en exploitation

NOTE Par exemple, essai d'un système qui est utilisé 8 h par jour pendant 24 h.

3.1.2
essai accéléré
essai au cours duquel le niveau des contraintes appliquées à une entité est choisi au-delà du niveau qui correspond aux conditions de référence, afin de réduire la durée nécessaire à l'observation des réponses de l'entité aux contraintes ou en vue d'accentuer ces réponses pour une durée donnée

NOTE Pour être valable, il convient qu'un essai accéléré n'altére ni les mécanismes de défaillance, ni les modes de panne, ni leur fréquence relative.

[VEI 191-14-07]

3.1.3
facteur d'accélération (temporelle)
rapport entre les durées nécessaires pour obtenir le même nombre fixé de défaillances ou de dégradations dans deux échantillons de taille identique sous deux ensembles de contraintes différents entraînant les mêmes mécanismes de défaillance et les mêmes modes de panne avec les mêmes fréquences relatives

NOTE Il convient que l'un des deux ensembles des contraintes corresponde à des conditions de référence.

[VEI 191-14-10]

3.1.4
temps d'exécution
temps pour réaliser un nombre établi de transactions

3.1.5
panne
état d'une entité inapte à accomplir une fonction requise, à l'exclusion de son inaptitude due à la maintenance préventive ou à d'autres actions programmées, ou due à un manque de moyens extérieurs

NOTE 1 Une panne est souvent la conséquence d'une défaillance de l'entité elle-même, mais elle peut exister sans défaillance préalable.

[VEI 191-05-01]

NOTE 2 En anglais, le terme « fault » est également employé dans le domaine des réseaux d'énergie électrique avec le sens donné en VEI 604-02-01[42]1; le terme correspondant en français est alors « défaut ».

NOTE 3 Dans la présente norme, le terme « panne latente » est utilisé pour insister sur le fait que la panne n'a pas encore provoqué de défaillance.

NOTE 4 Le logiciel seul est déterministe. Néanmoins, la présente norme prend en compte les logiciels embarqués dans les matériels, où le logiciel peut présenter des pannes latentes liées au matériel et à l'environnement, par exemple une protection insuffisante contre la double frappe, pas de somme de contrôle en communication, ou pas de contrôle de l'état des données d'entrée ou de sortie.

3.1.6
bogue
appellation courante pour désigner une panne latente de logiciel

3.1.7
indicateur de fiabilité
paramètre non fonctionnel pour signaler une défaillance probable dans un temps réduit

3.1.8
essai de taux de succès
essai répété un certain nombre de fois, ne devant présenter à chaque fois, aucune défaillance

3.1.9
système
ensemble d'éléments corrélés ou interactifs

[ISO 9000:2005, 3.2.1] [41]

NOTE 1 Dans le contexte de la sûreté de fonctionnement, un système présentera

– un but défini exprimé en termes de fonctions prévues,

– des conditions spécifiées de fonctionnement/d'utilisation, et

– des limites définies.

NOTE 2 La structure d'un système peut être hiérarchique [CEI 60300-1, 3.6] [43].

NOTE 3 Pour certains systèmes, tels que les produits de traitement de l'information, les données représentent une part importante des éléments du système.

[Future CEI 60300-3-15, modifiée] [44].

3.1.10
transaction
ensemble de paramètres d'entrée et de conditions préalables sélectionnés à partir des charges en service pour le système

1 Les références entre crochets se réfèrrent à la bibliographie.

3.1.11
analyse de cause initiale
activité visant à identifier la cause d'une panne ou d'une défaillance, de telle sorte qu'elle puisse être éliminée par des modifications de conception ou de processus

3.1.12
erreur
écart ou discordance entre une valeur ou une condition calculée, observée ou mesurée, et la valeur ou la condition vraie, prescrite ou théoriquement correcte

NOTE 1 Une erreur peut être causée par une entité en panne, par exemple une erreur de calcul faite par un ordinateur en panne.

NOTE 2 Le terme français « erreur » peut aussi désigner une erreur humaine (voir VEI 191-05-25).

[VEI 191-05-24]

3.1.13
faute
erreur (humaine)
action humaine qui produit un résultat différent de celui qui est recherché

[VEI 191-05-25]

3.1.14
défaillance
cessation de l'aptitude d'une entité à accomplir une fonction requise

NOTE 1 Après défaillance d'une entité, cette entité est en état de panne.

NOTE 2 Une « défaillance » est un passage d'un état à un autre, par opposition à une « panne », qui est un état.

NOTE 3 La notion de défaillance, telle qu'elle est définie, ne s'applique pas à une entité constituée seulement de logiciel.

[VEI 191-04-01]

NOTE 4 Le logiciel seul est déterministe. Néanmoins, la présente norme prend en compte les logiciels embarqués dans les matériels, où le logiciel peut présenter des pannes latentes liées au matériel et à l'environnement, par exemple une protection insuffisante contre la double frappe, pas de somme de contrôle en communication, ou pas de contrôle de l'état des données d'entrée ou de sortie.

3.1.15
intensité de défaillance
intensité de défaillance; intensité instantanée de défaillance
$z(t)$

limite, si elle existe, du quotient de l'espérance mathématique du nombre de défaillances d'une entité réparée, pendant un intervalle de temps $(t, t + \Delta t)$, par la durée Δt de l'intervalle de temps lorsque cette durée tend vers zéro

NOTE 1 L'intensité instantanée de défaillance s'exprime par la formule:

$$z(t) = \lim_{\Delta t \to 0+} \frac{E[N(t + \Delta t) - N(t)]}{\Delta t}$$

[VEI 191-12-04]

NOTE 2 Afin d'éviter toute confusion, l'expression « intensité instantanée de défaillance » sera utilisée dans la présente norme, étant donné qu'un système est réparé lorsqu'il est défaillant, et qu'une panne latente est réparée (éliminée) lorsqu'elle entraîne une défaillance.

3.2 Abréviations

CPU	Processeur de l'unité centrale (*central processor unit*)
CEM	Compatibilité électromagnétique
DES	Décharge électrostatique
AMDE	Analyse des modes de défaillance et de leurs effets
MTBF	Moyenne des temps de bon fonctionnement (*mean operating time between failures)*
RAM	Mémoire vive (*random access memory*)

3.3 Symboles

C	nombre total de transactions
$D(t)$	nombre de pannes détectées par le temps t
F_u	nombre inacceptable de transactions défaillantes sur C transactions
i	nombre de pannes
M	probabilité selon laquelle un système avec une fiabilité inacceptable subit avec succès N essais sans aucune défaillance
m	nombre de pannes latentes dans le système
N	nombre de transactions à effectuer sans défaillance
p	probabilité inacceptable de défaillance par transaction
RCM $r(T_t)$	métrique pour le critère de risque (*risk criterion metric*) pour les pannes latentes restantes à la durée d'essai totale T_t
r_c	nombre estimé de pannes latentes restantes dans le système
$r(Tt)$	pannes latentes restantes (non détectées) prévues à la durée d'essai cumulée T_t
s	nombre d'intervalles de temps d'essais utilisés dans le modèle de Schneidewind pour estimer les paramètres du modèle
t	durée d'essai réelle
t_{statut}	durée d'essai au statut
$T_{D(t)}$	durée d'essai cumulée de laquelle $D(t)$ pannes ont été détectées
T_i	durée d'essai cumulée lors de la détection de la panne i
T_{min}	durée d'essai minimale devant être cumulée par le système pour 0 défaillance
T_t	durée d'essai cumulée mesurée en unités de temps du modèle de Schneidewind
z	intensité instantanée de défaillance acceptable
z_i	intensité instantanée de défaillance de la panne i
θ_i	moyenne cumulée des temps de bon fonctionnement (MTBF) lors de la détection de la panne i

NOTE Le terme « MTBF cumulée » est utilisé pour être conforme à d'autres modèles de croissance de fiabilité décrits dans les publications. Il est informatif dans l'affichage d'une croissance en fiabilité en raison de l'élimination de la cause initiale de la défaillance. La MTBF cumulée (θ_t) pour chaque panne i est déterminée par $\theta_i = T_i / i$.

α	constante empirique dans le modèle de Schneidewind – intensité de défaillance à la durée d'essai = 0

β constante empirique dans le modèle de Schneidewind – constante de proportionnalité pour l'intensité de défaillance dans le temps – Unité: $(\text{temps})^{-1}$

δ probabilité selon laquelle aucune défaillance ne se produit par $T_{\min}$ pour une intensité instantanée de défaillance acceptable donnée

4 Généralités

La présente norme fait partie de la série de normes entrant dans le cadre du guide d'application CEI 61014 [34].

La présente norme s'applique aux grands systèmes matériels-logiciels lorsqu'ils sont soumis aux essais à l'aide d'une charge en service simulée. Par conséquent, on ne sait pas au cours de l'essai si une défaillance est provoquée par le matériel, le logiciel, la charge en service ou une combinaison des trois. Une défaillance peut être causée par une défaillance matérielle, par exemple une défaillance de la mémoire vive (RAM: *random access memory*), une modification du rythme, entraînant une collision de données, ou une perturbation électromagnétique modifiant les données transmises. La défaillance peut également être due à une panne latente du logiciel ou à des données illégales. La façon dont l'entité défaillante est réparée ou dont le logiciel est modifié n'est pertinente, pour la présente norme, que dans la mesure où ceci influence les décisions d'essai, par exemple à travers les hypothèses du modèle statistique.

Presque tous les systèmes modernes se composent de logiciels embarqués. Le logiciel est généralement soumis aux essais sur un matériel de développement à l'aide de transactions déduites des spécifications du système. La réalisation du logiciel est souvent terminée tardivement, de telle sorte que la durée pour essayer le logiciel dans le matériel réel est limitée. Il n'est généralement pas acceptable que le client soit le premier à faire fonctionner le logiciel dans le matériel réel. Il est par conséquent nécessaire qu'une norme fournisse un guide concernant les essais et la croissance de fiabilité du matériel avec le logiciel embarqué.

Avec le matériel, on suppose que les défaillances précoces sont provoquées par une panne latente dans le matériel. En fonction du type de contrainte et du niveau de contrainte, ces pannes latentes peuvent entraîner des défaillances permanentes ou intermittentes après un certain temps, comme par exemple une fissure dans un composant. Dans des conditions de fonctionnement sèches, sans vibrations ni chocs, la panne latente peut rester une panne latente. Néanmoins, dans des conditions de fonctionnement humides, l'humidité et les contaminants peuvent pénétrer dans la fissure et causer de la corrosion, entraînant au final une panne permanente. De même, les vibrations ou les chocs peuvent provoquer une propagation de la fissure, pouvant donner lieu à une panne permanente après un certain temps.

Le logiciel seul est déterministe. Ceci signifie qu'une panne latente dans le logiciel (communément appelée « bogue ») n'entraînera pas de défaillance jusqu'à ce que la partie du code renfermant la panne latente soit activée. L'instant où ceci se produit dépend des conditions de fonctionnement (par exemple paramètres d'entrée et états internes du programme, par exemple contenu de la mémoire). Il existe par conséquent une similitude entre les pannes latentes du matériel et les pannes latentes du logiciel. La panne latente du logiciel, une fois qu'elle est activée, peut entraîner une panne permanente mais ne donnera souvent lieu qu'à une défaillance intermittente.

Les défaillances logiques sont systématiques (c'est-à-dire qu'elles peuvent être reproduites à volonté, une fois que l'élément à l'origine de la panne associée est connu). Etant donné que l'élément à l'origine de toute panne latente se produit de manière aléatoire dans l'environnement de fonctionnement du système, les défaillances logiques sont observées comme un processus stochastique. Par conséquent, les mesures habituelles de fiabilité peuvent être appliquées (probabilité de durée de fonctionnement avant la prochaine

défaillance, intensité de défaillance, etc). La croissance de fiabilité sera normalement effective une fois que les pannes latentes seront supprimées.

Dans la présente norme, le terme « panne latente » sera par conséquent utilisé pour couvrir les faiblesses du matériel ainsi que les bogues du logiciel [10].

Un exemple de défaillance provoquée par une combinaison de matériels et de logiciels pourrait être qu'une panne latente de matériel provoque un refroidissement insuffisant d'un composant. L'échauffement modifie les temps de propagation dans le circuit, provoquant une collision de données qui entraîne une défaillance logicielle. Une autre combinaison pourrait consister en ce qu'une erreur de conception du matériel entraîne une protection insuffisante des fils de transmission. Le niveau croissant de bruit électromagnétique altère les données dans les fils de transmission, provoquant une défaillance logicielle, étant donné que le logiciel ne possède pas de dispositif de correction d'erreur et que l'environnement de fonctionnement présente un niveau de bruit électromagnétique élevé.

La présente norme couvre les systèmes réparables qui sont produits en très faible quantité, de sorte que l'expérience acquise au cours des essais de systèmes similaires précédents est limitée ou inexistante. Elle peut être utilisée lorsqu'un fabricant veut optimiser la durée des essais d'acceptation interne (au fabricant) et le rodage. Elle traite des essais de croissance avant ou lors de la livraison d'un système fini. Les essais peuvent par conséquent avoir lieu dans les locaux du fabricant ou de l'utilisateur final. Elle peut aussi être utilisée lorsqu'une entreprise veut optimiser la durée des essais de production finale au cours de la fabrication d'entités uniques, de petites séries ou au cours des essais d'un prototype.

Elle peut également être utilisée par le propriétaire d'un seul ou de plusieurs grands systèmes, afin d'améliorer ces systèmes. Si l'utilisateur d'un système traite la croissance de fiabilité par une politique de mise à jour des matériels et des logiciels avec des versions améliorées, la présente norme peut être utilisée pour contrôler le processus de croissance.

La présente norme ne couvre pas les logiciels isolément, mais elle peut être utilisée lorsque des logiciels embarqués sont soumis aux essais dans un système matériel utilisant des stratégies d'essai qui entraînent une diminution du nombre de défaillances en fonction de la durée d'essai, par exemple un essai de logiciel avec une charge opérationnelle simulée. Les méthodes décrites sont bien adaptées pour soumettre aux essais et améliorer la robustesse d'un logiciel contre les transitoires et les perturbations provoquées par la charge opérationnelle et par le système matériel. Elles concernent les grands systèmes matériels/logiciels, mais ne couvrent pas les grands réseaux, par exemple les réseaux de télécommunications et d'alimentation, étant donné qu'il est difficile d'isoler les nouvelles parties de ces systèmes au cours du processus d'essais.

La croissance de fiabilité est une méthode visant à améliorer la qualité en identifiant et en supprimant les pannes latentes, mais il convient de ne pas l'utiliser comme moyen principal d'obtention de la qualité prévue et de la fiabilité des systèmes produits. Les grands systèmes sont souvent produits en faible quantité. Souvent, un seul système ou quelques systèmes sont produits. Les pannes latentes restantes introduites au cours des processus de conception et de fabrication doivent par conséquent être identifiées par l'intermédiaire d'essais de croissance du système fini. Toutefois, il convient d'utiliser un contrôle de processus approprié et des méthodes préventives telles qu'un processus AMDE (voir CEI 60812)[33], une analyse par arbre de panne (voir CEI 61025[35]) et des revues de conception (voir CEI 61160[37]), afin de réduire le nombre de pannes latentes dans le(s) système(s) produit(s). De plus, il convient de contrôler les processus de fabrication et d'assemblage, par exemple à l'aide d'un contrôle de processus statistique.

Dans certains cas, il peut être possible de diviser un grand système en un certain nombre de modules similaires, pour lesquels les méthodes de la CEI 61163-1 peuvent être utilisées. Les modules similaires sont ensuite considérés comme un lot se composant d'éléments similaires. Ceci couvrira les pannes latentes dans les modules, mais pas les défaillances provoquées par l'interaction des modules et par les interactions entre les modules et le logiciel embarqué.

Les défaillances causées par l'interaction entre les modules peuvent être trouvées uniquement en soumettant à des essais de croissance le système fini. Dans les systèmes modernes, de nombreuses défaillances sont provoquées par une interaction entre matériel et logiciel. Ces défaillances ne peuvent pas être détectées avant que l'ensemble du système ne soit fini et fonctionnel. Lorsque le prototype est le seul système produit, les essais de prototype et les essais de croissance fusionnent en un seul groupe d'essais.

La présente norme traite uniquement de la période de défaillance précoce du cycle de vie du système. Ceci signifie qu'elle ne couvre pas la période de défaillance aléatoire ni la période de défaillance par usure de la courbe en baignoire, telle qu'illustrée sur la Figure 1.

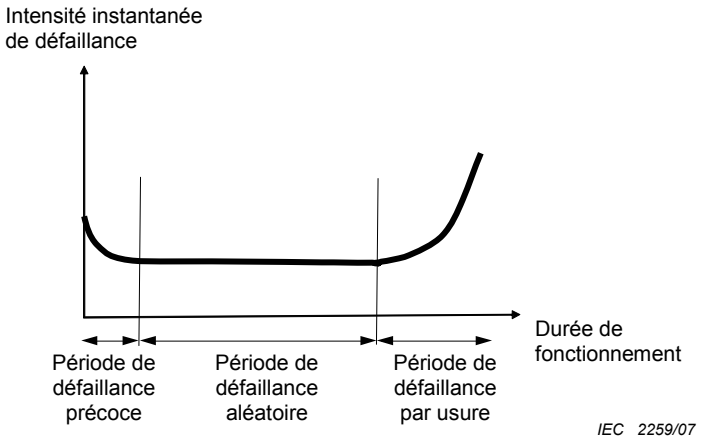

Figure 1 – Courbe en baignoire

NOTE La présente norme s'applique à la période de défaillance précoce. En raison d'une augmentation de la contrainte ou de la compression temporelle, cette partie de la durée de fonctionnement peut être couverte par une période plus courte d'essais de croissance.

Lors de la planification d'un processus d'essais de croissance de fiabilité, il convient que les décideurs prennent attentivement en compte le temps et les coûts par rapport à la performance du système, y compris les risques et les coûts associés aux défaillances précoces dans le système après livraison. Toutes les défaillances identifiées au cours des essais doivent être analysées soigneusement, afin de trouver la cause initiale et de s'assurer que les expériences sont utilisées pour éviter des problèmes similaires dans d'autres systèmes. Le ou les systèmes finis doivent être réparés ou mis à jour, soumis à de nouveaux essais pour voir s'ils fonctionnent normalement, et la documentation du système doit être mise à jour selon le cas.

Si des divergences apparaissent entre la présente norme et le contrat ou la (les) spécification(s) correspondant(e)(s), ces derniers ont préséance.

5 Planification et réalisation d'un essai de croissance de fiabilité

5.1 Etape 1 – Est-il recommandé d'utiliser un essai de croissance de fiabilité?

L'essai de croissance de fiabilité est approprié dans les cas suivants:

- lorsque les économies de coûts dues à la réduction des défaillances précoces sont supérieures au coût de l'essai, incluant le contrôle nécessaire et le matériel d'essai;
- lorsqu'aucune donnée d'essai précédente n'existe pour l'ensemble du système, étant donné qu'un seul ou que quelques systèmes ont été produits, ou qu'un seul système nécessite des essais;
- lorsque des défaillances précoces sont attendues en raison de pannes latentes introduites par les processus d'assemblage et dans les composants ou en raison de tolérance incompatibles entre les composants dans le système ;
- lorsqu'il convient que des défaillances précoces appropriées dans les modules et les composants soient décelées par un déverminage sous contraintes avant le début de l'essai du système (voir CEI 61163-1 et CEI 61163-2);
- lorsque des défaillances précoces sont prévisibles en raison de l'interaction entre le matériel du système et le logiciel embarqué;
- lors de l'utilisation d'une stratégie d'essai dont l'un des objectifs est la croissance de fiabilité, c'est-à-dire que l'intensité de défaillance devrait diminuer avec la durée d'essai;
- lorsque des essais sont effectués à l'aide de charges en service simulées, si possible en utilisant des charges supérieures à la moyenne, et lorsque des charges anormales adaptées (données polluées, données illégales ou conditions de surcharge) peuvent être ajoutées; ou
- lorsque des pannes latentes éventuelles de matériels entraînent des défaillances permanentes ou intermittentes par accroissement des contraintes environnementales, c'est-à-dire par augmentation de la température, des variations de température, des vibrations, des chocs, etc.

5.2 Etape 2 – Définitions des défaillances et recueil de données

Une approche pratique consiste à énumérer les exigences du système et à vérifier parmi celles-ci lesquelles il convient de contrôler. Puis à déterminer la façon dont le système peut être contrôlé au cours de l'essai. La spécification d'essai doit définir les défaillances pertinentes et non pertinentes.

Les défaillances pertinentes sont les défaillances soudaines (absence de fonction) ainsi que les défaillances progressives (dégradation). Il convient de définir les autres défaillances liées au logiciel, c'est-à-dire absence de réponse, mauvaise réponse, blocage du système ou temps de réponse excessif. Les défaillances peuvent être provoquées par le matériel, le logiciel embarqué ou par l'interaction entre le matériel et le logiciel, par exemple un écart dans les temps de propagation entraînant une collision de données ou un bruit électromagnétique modifiant les données.

Les défaillances non pertinentes sont les défaillances causées par le matériel d'essai, le dispositif de surveillance ou par les opérateurs d'essai. Si les essais de robustesse du système aux erreurs humaines (fautes commises par l'opérateur) doivent être inclus dans l'essai de croissance, ces erreurs doivent être définies comme des défaillances pertinentes.

Si possible, il convient de contrôler en permanence le fonctionnement et les performances du système. Dans la mesure où ceci n'est pas possible, il convient d'effectuer, à intervalles réguliers, un essai fonctionnel, comprenant une vérification du fonctionnement des éléments redondants. Lorsque des cycles de contraintes sont utilisés, il convient de vérifier le fonctionnement du système après chaque cycle. Il est recommandé de surveiller au cours de

l'essai l'état de la redondance et la reconfiguration automatique ainsi que d'autres paramètres pertinents du système interne.

Les modifications du système telles que le remplacement d'un module ou le changement des modes de fonctionnement doivent également être enregistrées. Une procédure pratique consiste à consigner tous les événements, par exemple le début, l'arrêt, la défaillance, la mise à niveau, le changement de configuration, c'est-à-dire le mode de fonctionnement, etc., dans le protocole d'essai. Il est recommandé d'inviter l'équipe d'essai et les opérateurs utilisateurs à commenter et à émettre des suggestions sur le fonctionnement du système.

Pour les méthodes 1, 2, 4, 5 et 6, la durée d'essai de fonctionnement avant défaillance doit être enregistrée. La référence temporelle doit être définie. Il peut s'agir, par exemple, de la durée d'essai en heures ou en minutes, de la durée de fonctionnement ou du temps du processeur de l'unité centrale (temps CPU). Pour réduire la durée d'essai, la compression temporelle ou une augmentation des contraintes (essais accélérés) peuvent être utilisées. Pour la méthode 3, le nombre de transactions avant défaillance doit être enregistré.

5.3 Etape 3 – Niveaux de contraintes

5.3.1 Généralités

Une procédure d'essais détaillée doit être définie avant le début du processus de croissance de fiabilité. Cette procédure doit répertorier la ou les méthodes utilisées pour les essais ainsi que les procédures de décision et les niveaux de confiance. Il convient également de décrire l'analyse de défaillance et les procédures de rapport. Il est recommandé que les processus soient adaptés au système spécifique ainsi qu'à l'équipement d'essai disponible, et aux moyens possibles de soumettre le système aux contraintes (voir CEI 61163-1 à titre de guide).

Afin que les pannes latentes entraînent aussi vite que possible des défaillances, il convient que les systèmes soient soumis à des contraintes en essai d'une manière adaptée à l'apparition de défaillances pertinentes sans pour autant introduire de modes de défaillance indépendants des défaillances en exploitation, et sans réduire significativement la durée de vie du système, par exemple, par usure des joints ou des composants à durée de vie limitée. Les conditions d'essais peuvent se situer au-delà des conditions de fonctionnement spécifiées, mais doivent toujours être maintenues à l'intérieur aux aptitudes inhérentes à la conception. Le but est d'empêcher les dommages du système et d'éviter l'introduction de défaillances qui ne se produiraient pas en cours d'exploitation.

La taille de la plupart des grands systèmes limite les contraintes qui peuvent être appliquées. Par conséquent, des facteurs d'accélération faible sont généralement utilisés. Etant donné que les essais consistent en la recherche des défaillances précoces, ceci pose rarement un problème. La compression temporelle accélère uniquement les modes de défaillance influencés par l'augmentation de la ou des contraintes. La conséquence peut être que certains modes de défaillance, par exemple la corrosion, ne sont pas accélérés, et sont même réduits. Dans la plupart des cas, cependant, ceci pose moins de problèmes, puisque les essais consistent en la recherche des défaillances précoces, et non des défaillances par usure.

Une augmentation des contraintes est utilisée dans cet essai pour faire en sorte que les pannes latentes entraînent des défaillances plus rapidement qu'en cours d'exploitation. Pour les méthodes qui sont fondées directement sur une efficacité décroissant avec la durée d'essai, par exemple les méthodes 1.2, 2, 3, 6 et 7, il n'est pas nécessaire d'estimer le facteur d'accélération. Pour les méthodes 1.1, 4 et 5, il est nécessaire d'estimer le facteur d'accélération si l'objectif de fiabilité est spécifié pour l'exploitation. Les méthodes pour estimer le facteur d'accélération figurent dans la CEI 61163-2.

5.3.2 Augmentation de la charge en service

Le type de contrainte le plus simple à augmenter est généralement la charge en service. Il convient que les profils de fonctionnement et d'utilisation constituent la base de la définition de la charge en service au cours de l'essai. La compression temporelle est une méthode très utile, par exemple l'augmentation du nombre de charges en service par unité de temps. Dans ce cas, le facteur d'accélération sur la charge en service peut facilement être estimé comme étant le rapport entre les transactions au cours de l'essai et les transactions au cours de l'exploitation, pendant la même période de temps.

Pour les logiciels, la charge en service peut souvent être augmentée à l'aide de données d'entrée réelles ou simulées avec une occurrence ou un volume plus élevé(e) qu'en fonctionnement normal. Il est recommandé de décider s'il convient que la charge en service simule des charges en service normales ou qu'elle comprenne aussi des conditions de fonctionnement inhabituelles, par exemple une charge déséquilibrée, une pointe de charge ou des conditions de fonctionnement extrêmes telles que des données illégales, polluées ou corrompues.

Il convient en principe d'utiliser la charge opérationnelle la plus élevée spécifiée. Dans une situation contractuelle, les parties peuvent se mettre d'accord sur le fait que la charge peut être augmentée au-dessus de la charge maximale spécifiée. En dehors d'une situation contractuelle, la charge ne doit pas être augmentée au-dessus de la limite de spécification, à moins que l'équipe responsable n'en décide autrement.

Dans le cas de dispositifs redondants ou de protection qui en principe ne fonctionnent pas dans un système, il convient de créer des conditions pour activer ces dispositifs à des intervalles de temps réguliers.

5.3.3 Augmentation des contraintes environnementales

5.3.3.1 Généralités

En principe, les types de contraintes décrits dans la CEI 61163-1 peuvent être utilisés pour les petits systèmes. Pour les grands systèmes, les types possibles de contraintes sont restreints par les limites résultant de leur grande taille, par exemple le système peut être trop grand pour entrer dans une enceinte climatique ou sur un équipement d'essai de vibration. Certaines parties du système peuvent être inaccessibles lorsque le système est assemblé et en cours d'exploitation. De plus, la présence du personnel d'exploitation peut réduire les possibilités d'augmentation du niveau de contrainte, par exemple la température ambiante.

Il convient que les méthodes indirectes, par exemple les essais par indicateur de fiabilité (voir 5.5.8 et CEI 60706-55[32])), soient considérées comme un complément ou comme une alternative à l'augmentation des contraintes (voir aussi [3]).

Les cycles de contraintes peuvent être conçus à l'aide de la CEI 60605-2.

Le programme d'essai doit énumérer les types de contraintes choisis ainsi que les niveaux de contraintes et leur durée. La réduction de la durée de vie pour les éléments à durée de vie limitée à cause de l'essai doit être estimée le cas échéant.

5.3.3.2 Contrainte thermique

La température d'exploitation du système peut souvent être accrue en augmentant la température de la pièce ou en réduisant le refroidissement (c'est-à-dire en recouvrant les entrées ou les sorties, ou en réduisant la vitesse des ventilateurs). Le débit de l'air ou de l'eau de refroidissement peut être diminué. De plus, la température peut être soumise à des cycles (cycles thermiques). Il convient que les cycles de températures comprennent un démarrage à froid, dans la mesure où ceci provoquera souvent des gradients thermiques maximaux dans le système.

5.3.3.3 Niveau d'humidité

Des essais de corrosion sont généralement effectués au niveau du composant, mais une humidité relative élevée peut entraîner une augmentation des courants de fuite.

Les décharges électrostatiques constituent généralement un essai séparé, mais une humidité relative faible peut provoquer des décharges électrostatiques provenant de personnes ou de parties mobiles. Par conséquent, il peut être approprié, dans certains cas, d'augmenter ou de diminuer l'humidité relative pour le système ou une partie du système au cours de l'essai.

5.3.3.4 Contrainte mécanique

Des vibrations mécaniques peuvent être introduites à l'aide d'un matériel de vibration ou d'un marteau pneumatique sur le châssis du système [1].

5.3.3.5 Tension et transitoires électriques

La tension provenant de l'alimentation peut être augmentée ou diminuée selon le cas. Les transitoires peuvent être introduites dans la tension d'alimentation et dans les câbles de signaux (voir CEI 60605-2).

5.4 Etape 4 – Analyse de défaillance et classification des défaillances

5.4.1 Généralités

Lorsqu'une défaillance est observée, la première action doit consister à indiquer la durée d'essai ou le nombre de transactions avant défaillance. Il faut ensuite décider si le système doit être arrêté, s'il n'est pas déjà arrêté par la défaillance. Il peut être nécessaire d'arrêter le fonctionnement du système pour les raisons suivantes:

- pour des raisons de sécurité;
- pour que la défaillance ne provoque pas de défaillances secondaires, détruisant le système ou une partie du système;
- afin de réaliser une analyse de défaillance; ou
- afin de réparer l'entité défaillante.

Dès que des preuves ont été recueillies pour le classement des défaillances, il convient de décider s'il est recommandé de réparer immédiatement l'entité défaillante ou de différer la réparation. Dans certains cas, il peut être possible de poursuivre les essais sans réparer l'entité défaillante. La condition est la suivante: une analyse doit montrer qu'il est probable que la défaillance n'entraînera pas de défaillances secondaires et qu'il sera toujours possible de soumettre aux essais la majeure partie du système restant. Ce jugement exigera une connaissance technique du système. Dans le protocole d'essai, il convient de consigner si une partie du système ne fonctionne pas ou n'est pas surveillée à cause de pannes non réparées.

S'il est décidé de différer la réparation, l'influence sur l'essai poursuivi doit être prise en compte et documentée, par exemple indiquer qu'une partie du système ne fonctionnera pas, ou que la redondance peut être réduite dans les essais supplémentaires. Pour les matériels, l'entité défaillante peut généralement être réparée en changeant un module, un composant ou en effectuant un réglage. Les modules et les composants échangés doivent être conservés pour une analyse de défaillance ultérieure. Dès que possible, une analyse approfondie de cause initiale [27],[28] de chaque défaillance doit être réalisée, afin de mettre en œuvre des actions correctives dans le système (changements de composants, modifications du processus et/ou de la conception).

Pour les défaillances liées au logiciel, le mode de défaillance sera souvent éliminé en modifiant le code. Ces modifications sont généralement introduites comme une nouvelle version de logiciel, mais il peut souvent être possible de poursuivre les essais comme suit:

- en enregistrant la défaillance et la durée d'essai avant chaque défaillance, mais en n'arrêtant pas l'essai lorsque la défaillance se produit de nouveau;
- en enregistrant la défaillance et la durée d'essai avant chaque défaillance, et en réinitialisant le logiciel lorsque la défaillance se produit de nouveau; ou
- en enregistrant la défaillance et la durée d'essai avant chaque défaillance, et en insérant une correction dans le logiciel qui neutralise la défaillance lorsqu'elle se produit. Ceci peut avoir pour conséquence un non-fonctionnement d'une partie du logiciel pendant le reste de l'essai, et doit être documenté dans le rapport d'essai. Si possible, il convient d'ajouter un compteur pour enregistrer la durée d'essai à chaque fois que la défaillance se produit à nouveau.

NOTE La vérification de l'élimination effective de la panne par les modifications apportées, et lorsque ce n'est pas le cas, les activités sont décrites en 5.6.

En poursuivant les essais, il convient de prendre en compte les défaillances observées ailleurs dans le système. Il convient de tenir compte de la reconfiguration et de la redondance dans le système. Ceci peut rendre possible la poursuite de l'essai, mais peut également signifier que la défaillance observée peut être due à une panne située ailleurs dans le système. De plus, il convient de prendre en compte la possibilité qu'une panne locale, due à la redondance et à la reconfiguration automatique, provoque une défaillance du système. Il est par conséquent recommandé de surveiller au cours de l'essai l'état de la redondance et la reconfiguration automatique ainsi que d'autres paramètres pertinents du système interne.

Il convient que la documentation de la défaillance suive les lignes directrices données dans la CEI 60300-3-5. Toutes les défaillances doivent être classées comme pertinentes ou non pertinentes. Seules les défaillances pertinentes doivent être utilisées dans les estimations et les décisions.

5.4.2 Défaillances pertinentes

Les défaillances pertinentes sont celles provoquées par des composants fragiles, une conception inadéquate, des processus de fabrication inadéquats, des pannes latentes du logiciel ou des interactions entre le matériel et le logiciel. Si la robustesse du système contre les fautes commises par l'opérateur doit être incluse dans l'essai, des erreurs de ce type doivent être définies comme étant des défaillances pertinentes.

5.4.3 Défaillances non pertinentes

Les défaillances non pertinentes sont généralement causées par des erreurs humaines (fautes), des erreurs de maintenance ou par le matériel d'essai. Il convient d'envisager si les défaillances non pertinentes pourraient être évitées en modifiant les procédures, par exemple concernant le fonctionnement et la maintenance.

5.5 Etape 5 – Critères d'arrêt

5.5.1 Généralités

Etant donné que l'essai est fondé sur la croissance de fiabilité, c'est-à-dire sur une diminution du nombre de défaillances par heure d'essai, il est important d'avoir des critères pour arrêter l'essai. Ces critères peuvent être les suivants:

a) programmes d'essais fixes (5.5.2, méthodes 1.1 et 1.2);

b) analyse graphique (5.5.3, méthode 2);

c) essai de taux de succès (5.5.4, méthode 3);

d) estimation de fiabilité (5.5.5, méthode 4);

e) comparaison avec l'intensité instantanée de défaillance acceptable (5.5.6, méthode 5);

f) estimation des pannes latentes restantes (5.5.7, méthode 6); ou

g) essais de l'indicateur de fiabilité (5.5.8, méthode 7).

Lorsqu'aucune défaillance n'a été observée pendant une certaine durée prédéterminée, et à chaque fois qu'une défaillance a été identifiée, on doit décider si l'essai doit être arrêté ou poursuivi. Pour les méthodes 2, 5 et 6, pour lesquelles la décision d'arrêter dépend de l'efficacité décrossante, chaque mode de défaillance est uniquement inclus dans les critères d'arrêt la première fois qu'il se produit. Néanmoins, toutes les défaillances doivent être enregistrées, avec la durée de fonctionnement avant défaillance, afin qu'elles puissent être consignées et incluses dans l'estimation de fiabilité mise à jour, et surtout si la modification (réparation) ne supprime pas réellement cette panne particulière dans le fonctionnement futur du système (voir 5.6).

Les méthodes 1.1, 1.2, 2, 4, 5 et 7 sont utiles pour les systèmes dans lesquels il est prévu que les défaillances liées au matériel soient prédominantes, et les méthodes 3 et 6 sont recommandées lorsqu'il est prévu que les défaillances liées au logiciel soient prédominantes.

Les méthodes 1.2, 2, 3, 6 et 7 ne dépendent pas du facteur d'accélération utilisé au cours de l'essai, étant donné que les méthodes sont fondées directement sur l'efficacité décroissante. Pour les méthodes 1.1, 4 et 5, une estimation du facteur d'accélération est nécessaire si les utilisateurs veulent fonder le résultat d'essai sur une cible de fiabilité spécifiée dans les conditions d'exploitation (voir 5.3.1).

Lorsque les réparations sont différées, se reporter à 5.6.

Les procédures de décision et les niveaux de confiance sont traités en 5.5.2 à 5.5.8 pour les différentes méthodes.

5.5.2 Méthode 1 – Programmes d'essais fixes

5.5.2.1 Méthode 1.1 : Nombre fixe de cycles d'essais

Dans cette méthode, les essais sont arrêtés après le nombre prédéterminé de cycles d'essais. Les cycles d'essais peuvent être conçus à l'aide de la méthode décrite dans la CEI 60605-2. Le nombre de cycles peut être déterminé par l'une des deux façons suivantes:

- la durée de la période de défaillance précoce en durée de fonctionnement est estimée. d'après les expériences antérieures (par exemple en exploitation), Pour le cycle élaboré conformément à la CEI 60605-2, le nombre d'heures de fonctionnement équivalent à un cycle d'essai est connu. Le nombre requis de cycles peut être trouvé en divisant la durée de la période de défaillance précoce en heures par le nombre d'heures de fonctionnement équivalentes pour un cycle d'essai. L'avantage de cette méthode réside dans le fait qu'elle est très simple, mais elle exige une connaissance de la durée de la période de défaillance précoce à partir de systèmes antérieurs similaires; ou
- dans des publications telles que [2], des courbes sont publiées pour le rendement de différentes méthodes de contraintes. Le nombre de cycles peut être déterminé en prenant en compte l'efficacité décroissante d'un cycle de contrainte supplémentaire. L'avantage réside dans le fait qu'aucune connaissance précédente de la durée de la période de défaillance précoce n'est exigée. L'inconvénient est le suivant: les courbes sont de nature générale et ne peuvent donner que des lignes directrices grossières pour un système particulier et des conditions de contraintes particulières.

5.5.2.2 Méthode 1.2 : Nombre fixe de cycles d'essais sans défaillance

Dans cette méthode, l'essai est arrêté après un nombre prédéterminé de cycles d'essais (voir CEI 60605-2), mais le dernier cycle, ou un nombre spécifié de cycles, doit se dérouler sans défaillance. Dans le cas d'une défaillance pertinente, l'essai est par conséquent prolongé jusqu'à obtention du nombre exigé de cycles sans défaillance. La période sans défaillance peut être calculée à l'aide des méthodes de la CEI 61163-1, mais ceci exige certaines hypothèses ou une connaissance préalable des durées de fonctionnement avant défaillance des composants fragiles décrits en Annexe J de la CEI 61163-1. L'avantage réside dans le

fait que le ou les cycles sans défaillance augmentent la certitude que les défaillances précoces dans le système ont été détectées. L'inconvénient est que la méthode exige une expérience antérieure avec les composants et les modules utilisés dans le système.

5.5.3 Méthode 2 – Analyse graphique

Dans cette méthode, le nombre cumulé de défaillances pertinentes observées est tracé en fonction de la durée d'essai pour chaque système individuel. Dès qu'une défaillance pertinente est observée, la courbe est mise à jour. La difficulté consiste à estimer la forme de la courbe dans le futur (voir Figure 2). Lorsqu'aucune défaillance n'a été observée pendant un certain temps ou à intervalles fixes, une courbe de décision est réalisée. Cette courbe de décision contient la courbe tracée avec une défaillance fictive à la fin de la durée d'essai actuelle ou lors de la durée de décision (voir Figure 3).

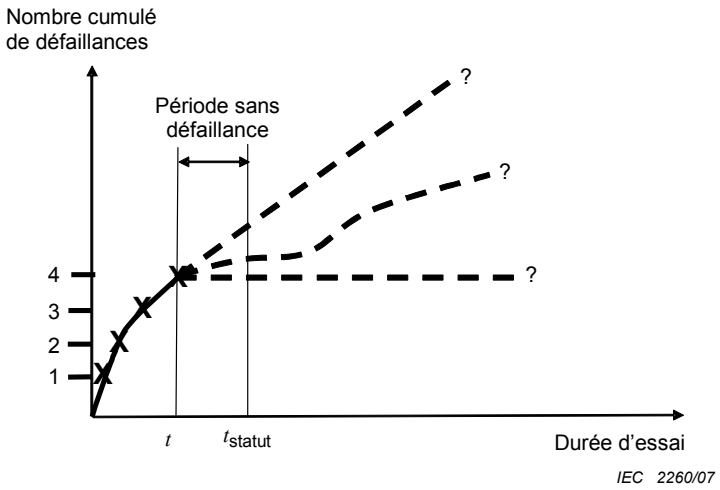

Figure 2 – Evaluation de la hausse ou de la baisse de la courbe de défaillance cumulée

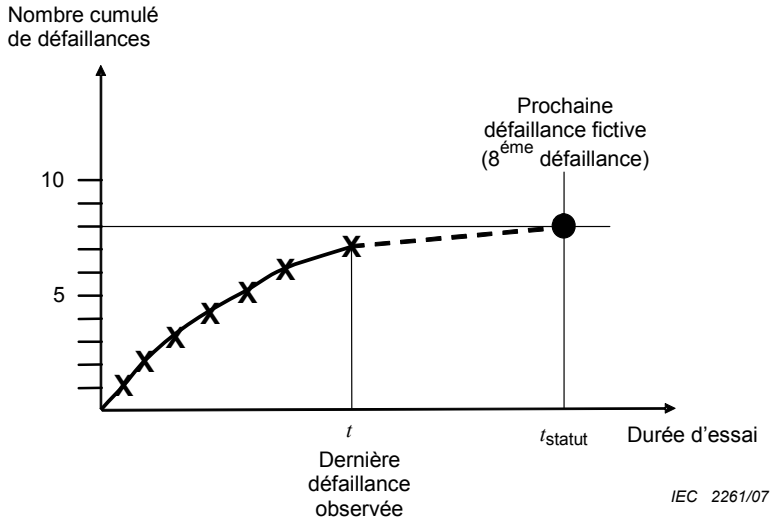

Figure 3 – Méthode 2

Cette courbe de décision peut ensuite être évaluée. Le critère de décision est fondé sur le coût entraîné par les défaillances après livraison par rapport au coût de l'essai.

Avant le début de l'essai, le nombre minimal de défaillances par unité de durée d'essai justifiant la poursuite de l'essai est déterminé (par exemple, 1 défaillance par 24 h de durée d'essai). Ce nombre peut être déterminé d'après le coût impliqué par la poursuite de l'essai, le coût lié à la correction d'une défaillance au cours de l'essai de croissance, ou le coût relatif à la correction d'une défaillance au cours du fonctionnement du système, en prenant en compte les pertes de fonctionnement causées par la défaillance. Lorsque la pente de la courbe de décision pendant une durée prédéterminée (par exemple durée d'essai de 24 h) est inférieure à la valeur convenue, l'essai est arrêté.

Si les conditions d'essais sont modifiées de façon importante au cours de l'essai, il convient de réaliser un nouveau tracé d'après les défaillances trouvées dans les nouvelles conditions uniquement.

5.5.4 Méthode 3 – Essai de taux de succès

5.5.4.1 Généralités

L'essai de taux de succès est un cas particulier du critère de la période sans défaillance. Le but est de confirmer qu'il est très probable que la probabilité de défaillance est inférieure à une limite supérieure spécifiée. Un certain nombre d'essais avec des charges en service aléatoires sont effectués sur le système. Si une défaillance est observée, l'entité défaillante est réparée avant que l'essai ne soit poursuivi. L'essai est poursuivi jusqu'à ce qu'une séquence de transactions réussies successives soit suffisamment concluante pour s'assurer que la probabilité de défaillance a un niveau faible acceptable. Cette méthode permet une détermination du nombre de transactions, étant donné que la probabilité acceptable maximale de défaillance est spécifiée. La méthode suppose que chaque transaction représente un essai indépendant provenant de la même population, simulant des applications réelles.

Cette méthode est appropriée pour les systèmes possédant des logiciels embarqués. La nature des transactions dépend de la nature du logiciel. Les transactions doivent simuler une charge en service normale mais élevée du système [4].

5.5.4.2 Nombre de transactions requises

Dans cette méthode, les charges en service simulées sont utilisées comme des transactions. Ceci assurera que les conditions pour le système en essai varient d'une transaction à l'autre, c'est-à-dire condition initiale, état des mémoires tampons, registres, chemin d'accès, etc. Par conséquent, les pannes latentes qui peuvent être non détectées dans un programme d'essai structuré après les spécifications des exigences, peuvent être détectées avec cette méthode.

Dans la mesure où seule une très faible part des combinaisons envisageables entre les paramètres d'entrée, les paramètres de sortie et les chemins d'accès peut être exécutée, il n'est jamais certain que la probabilité de défaillance soit suffisamment faible. Il est toutefois possible de calculer la probabilité selon laquelle un système avec une fiabilité inacceptable aurait subi, avec succès, les essais réalisés. Les transactions sont supposées être indépendantes:

$$M = (1 - p)^N \quad (1)$$

où M est la probabilité selon laquelle un système avec une probabilité de défaillance inacceptable p peut subir avec succès les N essais. Les Tableaux 1 et 2 donnent certaines valeurs de M pour deux valeurs de p et diverses valeurs de N.

Si C est le nombre total de transactions à partir desquelles l'essai est choisi, et s'il est considéré comme inacceptable pour F_u ou plus d'entraîner une erreur, alors la probabilité M selon laquelle un système avec une probabilité de défaillance inacceptable par transaction de F_u/C aurait subi avec succès les N essais peut être estimée ainsi:

$$M = (1 - F_u/C)^N \quad (2)$$

Le système en essai doit subir N transactions sans défaillance, avant que l'essai ne puisse être arrêté car concluant.

Tableau 1 – Probabilité selon laquelle un système avec une probabilité de défaillance de 0,001 subira avec succès N essais successifs

p = 0,001	
N	$M = (1 - p)^N$
500	0,606 38
600	0,548 65
700	0,496 41
800	0,449 15
900	0,406 39
1 000	0,367 70
1 500	0,222 96
2 000	0,135 20
2 500	0,081 98
3 000	0,049 71
3 500	0,030 14
4 000	0,018 28
4 500	0,011 08
4 700	0,009 07
5 000	0,006 72

Tableau 2 – Probabilité selon laquelle un système avec une probabilité de défaillance de 0,000 001 subira avec succès N essais successifs

p = 0,000 001	
N	$M = (1 - p)^N$
1 000 000	0,367 88
2 000 000	0,135 34
3 000 000	0,049 79
4 000 000	0,018 32
5 000 000	0,006 74
6 000 000	0,002 48
7 000 000	0,000 91
8 000 000	0,000 34
9 000 000	0,000 12
10 000 000	0,000 05

5.5.5 Méthode 4 – Estimation de fiabilité

Dans cette méthode, les outils statistiques de la CEI 61164 ou de la CEI 61710 sont utilisés pour estimer la croissance de fiabilité et pour comparer les estimations avec l'objectif de fiabilité pour le système. Pour de plus amples informations, voir la CEI 61164 et la CEI 61710.

5.5.6 Méthode 5 – Comparaison avec l'intensité instantanée de défaillance acceptable

5.5.6.1 Bases de la méthode

Cette méthode implique la réalisation d'un programme de croissance de fiabilité sur un système se composant à la fois de matériels et de logiciels. Au cours de cet essai, des essais de contraintes continus sont appliqués au système, visant à identifier les points faibles des processus de conception et de fabrication. Les essais impliquent l'application de contraintes environnementales et opérationnelles, y compris des taux de transmission de données extrêmes et des conditions pour précipiter les pannes latentes. Dans la mesure où les pannes latentes dans les processus de conception et de fabrication sont identifiées et corrigées, la fiabilité du système commence à croître. En raison de mécanismes de rétroaction propres à la méthodologie, la capacité des processus à produire des systèmes fiables dans le futur augmente également. Cette croissance est modélisée à l'aide du modèle de croissance de fiabilité qui permet à l'équipe de développeurs de contrôler régulièrement l'amélioration de la fiabilité. Enfin, une règle d'arrêt mathématique détermine la durée optimale pour la livraison du ou des systèmes.

Les caractéristiques du programme d'essais sont les suivantes:

- un processus de conception qui est entièrement contrôlé, de telle sorte qu'il soit systématique, prévisible, fiable et qu'il suive des méthodologies et des procédures définies;
- tous les composants/matériaux du système proviennent de vendeurs agréés;
- toutes les entités du système sont fabriquées par l'intermédiaire du processus final;
- une intensité instantanée de défaillance acceptable dans les conditions d'essais est choisie pour le système. Ceci représente une intensité instantanée de défaillance prédéfinie qu'il convient que le système atteigne avant la fin des essais et avant qu'il soit livré au client. Un facteur d'accélération estimé peut être utilisé pour faire passer l'intensité instantanée de défaillance des conditions d'utilisation aux conditions d'essais (voir 5.3.1 et 5.5.1);
- cette intensité instantanée de défaillance acceptable est ensuite utilisée pour déterminer une durée d'essai minimale en se basant sur la probabilité qu'aucune défaillance ne sera rencontrée au cours de l'essai;
- l'essai de contrainte débute dès que le système est disponible et se poursuit jusqu'à ce que le système final soit prêt pour livraison;
- l'essai implique l'application de contraintes environnementales et opérationnelles, y compris des taux de transmission de données extrêmes et des conditions pour identifier les points faibles dans les nouvelles conceptions. Ceci inclut des essais de robustesse du logiciel lorsque cela est stipulé;
- lorsque des modifications sont apportées à la conception, celles-ci sont intégrées dans les systèmes soumis à l'essai;
- les heures d'essais cumulées sont enregistrées pour chaque système soumis à l'essai; et
- le contrôle de l'essai assure que toutes les défaillances du système qui deviennent visibles à la sortie sont détectées.

Dans la mesure où les pannes latentes sont identifiées et corrigées au cours de la conception, dans les composants et au cours du processus de fabrication, la fiabilité du système commence à croître. Les références concernant cette méthodologie sont indiquées en [11], [12] et [13].

5.5.6.2 Modèle de croissance de fiabilité

Le modèle de croissance de fiabilité implique le tracé du MTBF cumulé (θ_t) sur l'axe *y* et la racine carrée de la durée d'essai cumulée (T_i) sur l'axe *x*. A mesure que des défaillances apparaissent, la courbe de croissance de fiabilité est mise à jour. Les améliorations de la

fiabilité seront observées par une augmentation de la pente, tandis que la détérioration de la fiabilité sera représentée par une diminution de la pente. Contrairement à d'autres modèles graphiques, l'influence des défaillances précoces sur le modèle de croissance global est réduite, dans la mesure où la durée d'essai augmente à cause de l'échelle sur l'axe *x*, s'agrandissant à mesure que l'essai progresse. Ce modèle de croissance de fiabilité permet à l'équipe de développeurs de contrôler et d'afficher les améliorations de la fiabilité lors des réunions. Le but n'est pas d'utiliser le modèle de croissance de fiabilité pour quantifier ou extrapoler les données de fiabilité ou les statistiques, mais plutôt d'être un outil graphique pour l'équipe de développeur constate la progression continue de l'essai de fiabilité. Par conséquent, il est acceptable de combiner les données provenant des diverses contraintes et des divers essais opérationnels. Un exemple du modèle de croissance de fiabilité est présenté en Annexe B.

5.5.6.3 Règle d'arrêt

La règle d'arrêt permet d'évaluer le moment où le système a atteint un niveau prédéfini de fiabilité, permettant d'arrêter les essais et de procéder à la livraison du système. Dès qu'une défaillance est rencontrée, le temps d'arrêt est recalculé afin d'atteindre le niveau prédéfini minimal de fiabilité. Tous les types de pannes latentes peuvent être adaptés à cette règle d'arrêt, et celle-ci est applicable à l'ensemble du système, intégrant à la fois le matériel et le logiciel.

Chaque panne latente est supposée présenter une intensité instantanée de défaillance de panne latente. Il est prévu que les pannes latentes avec l'intensité de défaillance la plus élevée apparaissent en premier. Arrivé à un certain point, toutes les pannes latentes ont été précipitées, ou les pannes latentes restantes ont une intensité de défaillance inférieure à l'intensité instantanée de défaillance acceptable z.

La règle d'arrêt utilise le concept d'une intensité instantanée de défaillance de la panne latente, dans lequel un système présente un nombre inconnu de pannes latentes, chacune d'entre elles ayant sa propre intensité instantanée de défaillance de panne. Un système qui renfermait initialement m pannes latentes présentera une intensité instantanée de défaillance décroissante, dans la mesure où le nombre de pannes latentes est réduit de m, à cause de la détection et de la correction. Si les pannes ne sont pas corrigées, ou si un composant défaillant est remplacé par un composant contenant la même panne latente, alors cette panne et son intensité instantanée de défaillance associée restent clairement dans le système (voir 5.6). Dans la suite du texte, dans un souci de concision, l'intensité instantanée de défaillance de la panne latente sera appelée intensité instantanée de défaillance.

Un exemple de cette méthode est présenté en Annexe B.

Les hypothèses de la règle d'arrêt peuvent être établies comme suit:

- le système présente un nombre inconnu de pannes latentes m;
- chaque panne latente i (i = 1,…,m) est indépendante et présente une intensité instantanée de défaillance associée z_i, qui se produit conformément à un processus de Poisson; et
- lorsqu'une défaillance se produit, sa cause est recherchée et la panne est trouvée et corrigée. Au cours de la correction d'une panne, aucune autre panne latente n'est introduite.

La règle d'arrêt prône une durée d'essai minimale (T_{min}), afin d'éviter l'arrêt trop précoce de l'essai (voir [11], [12] et [13]):

$$T_{\text{min}} = -\frac{\ln \delta}{z} \tag{3}$$

où z est l'intensité instantanée de défaillance acceptable et δ représente la probabilité d'apparition d'aucune défaillance par T_{min}. Il est préférable que la valeur de δ soit faible, dans la mesure où il n'est pas souhaitable de terminer l'essai avant de rencontrer des défaillances. δ = 0,05 indique qu'il y a seulement 5 % de probabilité pour que T_{min} soit atteinte sans avoir rencontré de défaillances. Cette valeur de δ est généralement recommandée.

Une fois que la durée d'essai minimale est cumulée, la règle d'arrêt présentée ci-dessous indique que l'essai doit être arrêté au plus tôt, c'est-à-dire au temps t, de telle sorte que:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2 \left(1 - e^{-t/T_i}\right)^2}} \leq z \tag{4}$$

L'intensité instantanée de défaillance acceptable z représente une intensité instantanée de défaillance prédéfinie qu'il convient que le système atteigne avant la fin des essais et avant que le système ne soit livré au client. Le terme $1/\left(t - T_{D(t)}\right)$ dans l'équation de la règle d'arrêt représente l'estimation ponctuelle de l'intensité instantanée de défaillance du système dans la période suivant la dernière défaillance. La règle d'arrêt est également assez robuste par rapport à la durée d'apparition de la défaillance et elle n'est pas nécessairement altérée par un faible nombre de défaillances, à condition bien sûr qu'elles ne se produisent pas à la fin de l'essai.

Les résultats d'essais ne sont significatifs que dans la mesure où le profil de la demande de l'application prévue est égal au profil de la demande de l'essai. S'il y a un écart entre ces profils, les résultats doivent être recalculés par rapport au nouveau profil. C'est la raison pour laquelle il est recommandé d'utiliser des charges en service simulées au cours de l'essai.

5.5.7 Méthode 6 – Estimation des pannes latentes restantes

Dans cette méthode, les paramètres pour un modèle statistique des défaillances du système matériel-logiciel en fonction de la durée d'essai sont estimés. Les pannes latentes restantes dans le système peuvent ensuite être estimées ainsi que la durée d'essai estimée, afin d'éliminer les pannes latentes restantes (voir [5], [6], [7] et [8]).

Un modèle statistique approprié est adapté au nombre observé de défaillances en fonction de la durée d'essai. Il convient d'estimer le modèle statistique dans les conditions d'essais spécifiques (par exemple, niveau de contrainte) et avec le système spécifique. Ceci présente l'avantage que la probabilité d'un nombre spécifié de pannes latentes restantes (par exemple moins d'une panne latente) peut être estimée. L'inconvénient est une procédure plus compliquée qui comprend l'adaptation d'un modèle statistique. Dans certains cas, le modèle statistique ne décrit pas suffisamment correctement les données, ou l'estimation des paramètres du modèle ne converge pas. Dans de tels cas, d'autres modèles statistiques provenant des publications peuvent être utilisés (voir [40] et [5]), à condition que leur adéquation soit justifiée. Cependant, aucun modèle ne correspondra exactement aux données, et il sera donc souvent nécessaire d'utiliser le modèle qui correspond le mieux aux données, en se basant sur un essai d'adéquation (voir [9]). Dans l'exemple de l'Annexe C, le modèle du Dr. Schneidewind est utilisé, mais il existe plusieurs modèles de défaillances logicielles, par exemple Goel et Okumoto ([22] et [23]), Jelinski et Moranda ([24]), Rushforth, Staffanson et Crawford ([25]), et Langberg et Singpurwallah ([26]).

Les étapes d'utilisation de la méthode sont les suivantes:

a) dès le début de l'essai, enregistrer la durée d'essai de fonctionnement avant défaillance pour chaque défaillance;

b) choisir un modèle de défaillance statistique approprié, permettant une estimation du nombre de pannes latentes restantes (voir [40], [5], [6], [7], [8] et les références mentionnées ci-dessus). Certains modèles donnent également une prévision de la durée d'essai restante (voir [40],);

c) estimer les paramètres du modèle choisi à l'étape b), d'après les défaillances observées;

d) déterminer le critère d'arrêt pour l'essai, exprimé comme la probabilité que les pannes latentes restantes dans le système soient inférieures à r_C (r_C peut être égal à 1);

e) calculer la probabilité selon laquelle les pannes latentes restantes dans le système sont inférieures à r_C ; et

f) poursuivre l'essai jusqu'à ce que la probabilité soit inférieure à la valeur de r_C spécifiée.

Un exemple de cette méthode est donné en Annexe C.

5.5.8 Méthode 7 – Essais de l'indicateur de fiabilité

Les essais de l'indicateur de fiabilité peuvent être utilisés pour détecter les pannes latentes avant qu'elles n'aient entraîné des défaillances. Un indicateur de fiabilité est un paramètre qui n'est pas l'un des paramètres fonctionnels du système, mais qui peut être surveillé au cours du processus d'essai (voir aussi CEI 60706-5).

Parmi ces paramètres, on peut citer par exemple les mesures du bruit électrique, par exemple mesure de bruit $1/f$ (voir [3]), ou l'augmentation de la température contrôlée par une caméra infrarouge [3]. De plus, le bruit d'origine mécanique ou la consommation de puissance peuvent être utilisés. Pour les logiciels, le temps de réponse peut être utilisé.

Dans les systèmes à microprocesseurs modernes, un certain nombre de caractéristiques d'autotest ou de caractéristiques du « boundary scan » (test par scrutation de connexions périphériques placées en configuration de registres à décalage) peuvent souvent être intégrées et contrôlées afin de vérifier l'état du système.

Il convient que l'utilisation d'indicateurs de fiabilité de ce type soit convenue dans le contrat et spécifiée dans le programme d'essais. Il convient que toutes les indications fournies par les indicateurs de fiabilité soient vérifiées de façon approfondie et, si une panne latente est trouvée, il est recommandé d'effectuer une analyse de cause initiale. Lorsqu'aucune panne latente n'est trouvée, il convient que le fonctionnement du système, si possible, soit poursuivi pendant une durée appropriée, afin de voir si une défaillance se déclarant plus tard était prévisible à partir de la dérive de la valeur de l'indicateur.

L'avantage de la méthode d'essai de l'indicateur est qu'elle peut détecter des pannes latentes avant qu'elles n'entraînent des défaillances. De plus, il peut être plus facile de surveiller les paramètres secondaires que les paramètres primaires (fonctionnels). L'inconvénient est que l'essai de l'indicateur ne peut détecter que des modes de défaillance spécifiques, de sorte que soit un nombre élevé d'indicateurs doit être utilisé, soit le système est déverminé pour un seul ou quelques modes de défaillance principaux. Des recherches supplémentaires sont nécessaires concernant les indicateurs de fiabilité, avant qu'ils ne puissent être utilisés largement dans l'industrie. Cependant, ils sont très prometteurs pour les essais et la maintenance préventive (voir CEI 60706-5 et [17] et [18]).

Toutes les indications correctes et incorrectes doivent être consignées et une matrice doit être réalisée pour résumer les indications correctes ainsi que le nombre de types d'erreurs pour chaque indicateur de fiabilité utilisé (voir Tableau 3). Le pourcentage d'observations dans chaque cellule du Tableau 3 doit être enregistré.

Pour être efficace, il convient que chaque indicateur de fiabilité se concentre sur une zone ou un composant spécifique où le problème potentiel est trouvé. Le choix des indicateurs de fiabilité exige par conséquent une connaissance technique du système ainsi qu'une bonne connaissance des modes de défaillance, de leur évolution et des indications précoces.

Tableau 3 – Décisions correctes et incorrectes en utilisant les indicateurs de fiabilité

	Résultat d'essai de l'indicateur de fiabilité	
	L'indicateur de fiabilité indique la présence d'une panne latente	**L'indicateur de fiabilité indique la présence d'aucune panne latente**
La panne latente était réelle	Décision correcte	Indicateur pas suffisamment sensible
La panne latente n'a pas pu être trouvée	Indicateur trop sensible	Décision correcte

5.6 Etape 6 Vérification des réparations et de la croissance de fiabilité

Lorsqu'en cas d'apparition d'une défaillance, des modifications sont apportées au système pour éliminer les pannes latentes et améliorer le système, le résultat de ces modifications doit être évalué lors d'essais ultérieurs ou prolongés, pour vérifier leur efficacité et pour s'assurer qu'elles n'ont pas encore introduit un autre mode de défaillance, qui n'avait pas été rencontré jusqu'alors. L'étendue de la durée d'essai supplémentaire dépendra de la nature de ces modifications. Les modifications peuvent également être vérifiées par simulation ou par un essai distinct, préparé spécifiquement pour traiter du mode de défaillance qui est atténué, par exemple un essai accéléré spécifique. Un essai de ce type ne permettra souvent pas de détecter une interaction possible entre la partie modifiée du système et l'autre partie du système. Par conséquent, l'essai au niveau du système devra souvent être prolongé, en particulier lorsque les réparations sont différées, de sorte qu'un certain nombre de modifications soient apportées au système en même temps, par exemple une nouvelle version de matériel ou une nouvelle version de logiciel.

En se basant sur la vérification de l'efficacité des modifications, la croissance de fiabilité estimée doit être mise à jour. Tous les modes de défaillance observés qui n'ont pas été éliminés par les modifications du système, le nombre de leurs répétitions au cours de l'essai ainsi que toutes les défaillances supplémentaires attribuées aux modifications doivent être inclus dans cette estimation.

5.7 Etape 7 – Rapport et rétroaction

Il convient que le rapport final contienne les informations suivantes, selon le cas:

– description du système, y compris révision du matériel et du logiciel;
– adaptation à un contrat et à des conditions de système spécifiques (par exemple reconfiguration et redondance);
– surveillance des paramètres et définitions des défaillances (voir 5.2);
– profils de fonctionnement et d'utilisation – charges en service au cours de l'essai (voir 5.3);
– conditions et matériel d'essai (voir 5.3), types de contraintes, niveau de contrainte, durée des contraintes et cycles de contraintes (voir CEI 60605-2 et CEI 61163-1);
– réduction de la durée de vie des éléments à durée de vie limitée;
– durée de fonctionnement avant défaillance et classification des défaillances (voir 5.4);
– procédés d'analyse de défaillance pour trouver les causes initiales (voir 5.4);
– critères d'arrêt et niveaux de confiance (voir 5.5);
– arrêt des essais (voir 5.5);

– réparations et modifications apportées au cours de l'essai;
– essai(s) de croissance répété(s) le cas échéant;
– modifications apportées au cours de l'essai (nouvelle révision du matériel et du logiciel) et façon dont le système a été réparé ou mis à jour et soumis à de nouveaux essais pour vérifier s'il fonctionne normalement;
– modifications apportées après l'essai – il convient de mettre à jour la documentation du système, si approprié; et
– conclusion de l'essai – fiabilité obtenue si possible avec un niveau de confiance.

La croissance de fiabilité résultante peut être présentée ainsi:

Méthode 1.1: Le programme d'essais fixe s'est déroulé sans défaillance/avec les défaillances suivantes;

Méthode 1.2: Le dernier cycle du programme d'essais fixe s'est déroulé sans défaillance;

Méthode 2: Aucune défaillance n'a été observée entre la durée d'essai A et la durée d'essai B (le rapport doit donner les nombres A et B – voir Figure 3);

Méthode 3: N transactions ont été réalisées sans défaillance. La probabilité que le système ait une intensité de défaillance de F_u/C est *M* ou moins (voir 5.5.4.2);

Méthode 4: La MTBF du système après l'essai de croissance de fiabilité est estimée être égale à (nombre à indiquer dans le rapport);

Méthode 5: L'intensité instantanée de défaillance après l'essai de croissance de fiabilité est estimée être $\leq z$ (voir Equation (4));

Méthode 6: Le nombre estimé de défaillances restantes est <C; ; ou

Méthode 7: Aucun niveau anormal des indicateurs de fiabilité suivants n'a été observé au cours de l'essai: (le rapport doit énumérer les indicateurs de défaillance utilisés).

Annexe A
(informative)

Exemple pratique de la méthode 3 – Essai de taux de succès

Un système contenant un logiciel embarqué de type « traitement par lot » doit être soumis aux essais.

L'expérience montre que la durée d'essai sera déterminée par la durée nécessaire pour identifier les pannes latentes pertinentes dans le logiciel.

Pour les essais, 25 000 transactions ont été enregistrées à partir de la charge réelle en service du système précédent. Elles sont supposées couvrir des cas de fonctionnement typiques lorsque le système est utilisé en pratique.

Il est décidé que le système peut passer de l'essai à une exploitation normale si la probabilité de plus de cinq de ces transactions entraînant une défaillance est inférieure ou égale à 10 % (c'est-à-dire, F_u = 5).

Le nombre de transactions devant être effectuées au cours de l'essai est déterminé par l'équation:

$$M = (1 - F_u/C)^N \quad \text{(A.1)}$$

ou

$0{,}10 = (1 - 5/25\,000)^N = (0{,}999\,8)^N$

N = 11 500 transactions

Par conséquent, il est nécessaire d'effectuer 11 500 transactions sur les 25 000, choisies de manière aléatoire.

Si aucune défaillance n'est détectée, l'essai est réussi et peut être arrêté.

Annexe B
(informative)

Exemple pratique de la méthode 5 – Comparaison avec l'intensité instantanée de défaillance acceptable

B.1 Tracé de croissance de fiabilité

Un exemple de la méthode est présenté à l'aide des données contenues dans le Tableau B.1. La durée d'essai cumulée de fonctionnement avant défaillance pour chaque panne est représentée dans la colonne 2. Après avoir observé chaque défaillance, ces informations sont ensuite transformées dans les colonnes 4 et 5 et utilisées pour élaborer respectivement les axes *x* et *y* du tracé de croissance de fiabilité, ainsi que pour déterminer le temps d'arrêt (colonne 5). Le tracé de croissance de fiabilité qui en résulte est représenté sur la Figure B.1. On peut observer, à partir du Tableau B.1 et de la Figure B.1, qu'une croissance de fiabilité durable se produit après environ 42 000 min d'essai cumulées (20e défaillance).

NOTE Lors des essais de logiciels, la durée d'essai est souvent mesurée en minutes.

Tableau B.1 – Croissance de fiabilité et temps d'arrêt pour l'exemple pratique

Nombre de pannes *i*	**Durée cumulée de fonctionnement avant défaillance** min T_i	**MTBF cumulée** min $\theta_i = \frac{T_i}{i}$	$\sqrt{T_i}$ min ^{0,5}	**Durée d'arrêt** min *t*
1	1	1	1	10 010
2	60	30	8	10 070
3	8 230	2 743	91	31 130
4	8 300	2 075	91	35 030
5	8 350	1 670	91	37 570
6	12 568	2 095	112	42 510
7	15 556	2 222	125	47 260
8	19 876	2 485	141	52 590
9	19 900	2 211	141	56 150
10	19 910	1 991	141	59 220
11	27 200	2 473	165	64 570
12	27 210	2 268	165	67 770
13	27 700	2 131	166	70 850
14	28 660	2 047	169	73 840
15	34 450	2 297	186	78 080
16	37 400	2 338	193	81 660
17	37 410	2 201	193	84 350
18	41 250	2 292	203	87 920
19	42 000	2 211	205	90 660
20	42 100	2 105	205	93 130
21	44 020	2 096	210	95 940
22	48 600	2 209	220	99 360
23	51 600	2 243	227	102 430
24	55 100	2 296	235	105 600
25	82 100	3 284	287	117 120
26	108 300	4 165	329	133 740

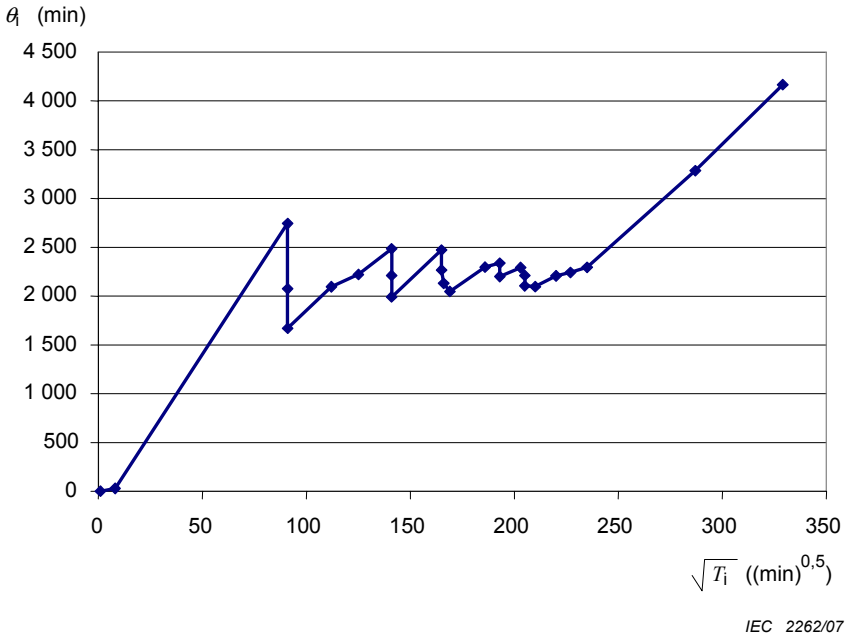

Figure B.1 – Tracé de croissance de fiabilité des données du Tableau B.1

B.2 Règle d'arrêt

On a considéré que l'intensité instantanée de défaillance acceptable était égale à $z = 1$ x 10^{-4} défaillances par minute d'essai, et une valeur de δ de 0,05 a été choisie. La durée d'essai minimale ($T_{\min}$) est calculée de la façon suivante:

$$T_{\min} = -\frac{\ln \delta}{z} = -\frac{\ln(0{,}05)}{1\times 10^{-4}} = 30\ 000 \text{ min} \qquad \text{(B.1)}$$

Il convient de ne pas arrêter l'essai avant 30 000 min, même si aucune défaillance n'a été rencontrée.

Dès qu'une défaillance se produit, le temps d'arrêt est recalculé.

La colonne 5 du Tableau B.1 indique les temps d'arrêt proposés en se basant sur la règle d'arrêt. Dans la mesure où la première défaillance se produit après seulement 1 min, $T_{D(t)}$ est, dans ce cas 1, le temps d'arrêt proposé après la première défaillance, présenté ci-dessous:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2\left(1-e^{-t/T_i}\right)^2}} = \frac{1}{t-1} + 3\sqrt{\frac{e^{-t/1}}{1^2\left(1-e^{-t/1}\right)^2}} \le z = 1\times 10^{-4} \qquad \text{(B.2)}$$

La plus petite valeur de t (à plus ou moins 10 min) satisfaisant à cette équation est 10 010 minutes. Noter que cette valeur est inférieure à la durée d'essai minimale proposée.

Le temps d'arrêt proposé pour la troisième défaillance est 31 130 min d'essai cumulées. Les calculs sont présentés ci-dessous:

$$\frac{1}{t - T_{D(t)}} + 3\sqrt{\sum_{i=1}^{D(t)} \frac{e^{-t/T_i}}{T_i^2\left(1 - e^{-t/T_i}\right)^2}} = \frac{1}{t - 8\,230} + 3\sqrt{\frac{e^{-t/1}}{1^2\left(1 - e^{-t/1}\right)^2} + \frac{e^{-t/60}}{60^2\left(1 - e^{-t/60}\right)^2} + \frac{e^{-t/8\,230}}{8\,230^2\left(1 - e^{-t/8\,230}\right)^2}} \leq z = 1 \times 10^{-4} \quad \text{(B.3)}$$

La plus petite valeur de t satisfaisant à cette équation est 31 130 min.

Ce résultat indique que les essais doivent se poursuivre jusqu'à ce que 31 130 min d'essai cumulées se soient écoulées avant la fin des essais. Cependant, on observe qu'avant que cette période de temps n'ait été atteinte, un autre défaut a été détecté, portant le temps d'arrêt à 35 030 min. Les essais sont arrêtés si ce temps d'arrêt a été atteint sans aucune autre défaillance.

Dans cet exemple, les essais ont été arrêtés après 133 740 min (environ 3 mois d'essais), étant donné qu'aucune défaillance n'a été observée entre 108 300 min et 133 740 min.

Annexe C
(informative)

Exemple pratique de la méthode 6 – Estimation des pannes latentes restantes

C.1 Méthode 6 – Estimation des pannes latentes restantes

Le système est soumis aux essais et le nombre de défaillances est compté dans chaque intervalle de temps. La durée est mesurée avec l'intervalle de temps comme unité, c'est-à-dire, s = 3 signifie 3 intervalles de temps (unités de temps).

Lorsque cinq défaillances ont été obtenues, les paramètres α et β du modèle de fiabilité du logiciel de Schneidewind sont estimés (voir [5] et [6]).

Les étapes suivantes sont ensuite effectuées:

Etape 1: Décider le nombre critique requis de pannes latentes restantes demeurant dans le système après l'essai r_c (c'est-à-dire moins d'une panne latente). Si le nombre prévu de défaillances est supérieur ou égal à r_c, l'essai de croissance doit alors être poursuivi.

Etape 2: Calculer:

$$\text{RCM}\, r(T_t) = (r(T_t) - r_c)/r_c = (r(T_t)/r_c) - 1 \qquad \text{(C.1)}$$

Etape 3: RCM $r(T_t)$ peut être tracé en fonction de T_t pour la valeur de r_c choisie ou être indiqué comme dans le Tableau C.1.

L'essai peut être arrêté dès que la valeur de RCM $r(T_t)$ est inférieure à r_c.

Le nombre de pannes latentes restantes dans le système peut être estimé à l'aide de l'équation:

$$r(T_t) = \left(\frac{\alpha}{\beta}\right)\exp(-\beta(T_t - (s-1))) \qquad \text{(C.2)}$$

La durée d'essai totale prévue pour atteindre le nombre spécifié de pannes latentes restantes dans le système peut être estimée à l'aide de l'équation:

$$T_t = \frac{\ln[\alpha/(\beta[r(T_t)])]}{\beta} + (s-1) \qquad \text{(C.3)}$$

EXEMPLE

Après la durée d'essai (intervalle d'essai) s = 9, les paramètres suivants ont été estimés pour le système en essai:

$$\alpha = 0{,}534 \ \left[(\text{temps}-\text{unités})^{-1}\right] \text{ et } \beta = 0{,}061 \ \left[(\text{temps}-\text{unités})^{-1}\right]$$

$r(T_t)$ et le nombre restant estimé de pannes latentes RCM $r(T_t)$ sont calculés pour moins d'une panne latente restante, c'est-à-dire que le nombre critique de pannes latentes restantes r_c = 1 pour la durée d'essai T_t =18 unités de temps et 52 unités de temps.

Tableau C.1 – Détermination de l'instant où l'essai doit être arrêté

T_t [temps – unités]	α $[(\text{temps – unités})^{-1}]$	β $[(\text{temps – unités})^{-1}]$	s [temps – unités]	$r(T_t)$ nombre de pannes latentes	RCM $r(T_t)$ nombre de pannes latentes
18	0,534	0,061	9	4,76	3,76
52	0,534	0,061	9	0,60	–0,40

La durée d'essai totale prévue pour atteindre le nombre spécifié de pannes latentes restantes $r(T_t)$ = 1 dans le système peut être estimée à l'aide de l'équation:

$$T_t = \frac{\ln[\alpha/(\beta[r(T_t)])]}{\beta} + (s-1) = \frac{\ln[0{,}534/(0{,}061[1])]}{0{,}061} + (9-1) = 43{,}56 \text{ unités de temps} \quad \text{(C.4)}$$

C.2 Bases du modèle de Schneidewind

Des informations de base sur le modèle de Schneidewind figurent en [5] et [40].

Bibliographie

[1] NIKOLSKY, George N. and TUSTIN, Wayne: « Using a Pneumatic Hammer for ESS », TEST, December/January, 1988/89

[2] « Technical Guidelines for the ESS process » IEST-RP-PR001.1 Published 1/1/1999 – ISBN: 1-877862-70-3 Document number: PR 01 - USA

[3] JENSEN, Finn: « Electronic Component Reliability », Wiley, 1995

[4] PARNAS, David L., VAN SCHOUWEN, John A. and SHU PO KWAN: « Evaluating of Safety Critical Software », Communications of the ACM, June 1990, Volume 33, No.6

[5] « Recommended Practice for Software Reliability R-013-1992 », ANSI, American Standards Institute / American Institute of Aeronautics and Astronautics, Washington DC, 1993

[6] SCHNEIDEWIND, Norman F.: « Introduction to Software Reliability with Space-Shuttle Example », RAMS Tutorial Notes, 2001

[7] SCHNEIDEWIND, Norman F.: « Reliability Modelling for Safety Critical Software », IEEE Transactions on Reliability, Vol. 46, No.1, March 1997, pp. 88-98

[8] SCHNEIDEWIND, Norman F.: « Software Reliability Model with Optimal Selection of Failure Data », IEEE Transactions on Software Engineering, Vol. 19, No. 11, November 1993, pp. 1095-1104

[9] LITTLEWOOD, B. and VERAL, J.L.: « Likelihood Function of a Debugging Model for Computer Software Reliability », IEEE Transactions on Reliability, Vol. R-30, No. 2, June 1981

[10] Michael Pecht (Editor): « Product Reliability, Maintainability and Supportability Handbook », ARINC Research Corporation, CRC Press, 1995

[11 DONOVAN, J. and MURPHY, E.: « Reliability Growth – A New Graphical Model », Quality and Reliability Engineering International, 1999, 15: pp. 167-174

[12] DONOVAN, J. and MURPHY, E.: « An Infrequently Used Stopping Rule Revisited », Quality Engineering, 2001, 13: pp. 367-376

[13] DONOVAN, J. and MURPHY, E.: « Total System Reliability: Integrated Model for Growth and Test Termination », Quality and Reliability Engineering International, 2005, 21: pp. 329-344

[14] MIL-HDBK-2164A, Environmental Stress Screening Process for Electronic Equipment - Revision A

[15] MIL-HDBK 344A, Environmental Stress Screening (ESS) of Electronic Equipment - Revision A

[16] Def-Stan 00-40, Reliability and Maintainability

[17] SALFNER, Felix and MALEK, Miroslaw : « Predicting Failures of Computer Systems: A Case Study for a Telecommunication System », IEEE Proceedings of IDPS 2006

[18] MALEK, Miroslaw : « Tutorial on Predictive Algorithms and Technologies for Availability Enhancement », International Service Availability Symposium (ISAS 2006), Helsinki 14 May 2006

[19] ETO, Hiroyuki and DOHI, Tadashi: « Analysis of a Service Degradation Model with Preventive Rejuvenation », International Service Availability Symposium (ISAS 2006), Helsinki 14 May 2006

[20] Re-Aéro-703-06-A: « Guide pour le Pilotage de la Croissance de Fiabilité » (Guidelines for the Reliability Growth Management) BNAe 1995/94 Ref.186

[21] CROW, L.H.: « Reliability Analysis for Complex, Repairable Systems » AD-A020 296 - AMSAA-TR 138 (Army Material Systems Analysis Activity)December 1975

[22] GOEL, A.L. and OKUMOTO, K.: « An Imperfect Debugging Model for Reliability and Other Quantitative Measures of Software Systems » in: Bayesian Software Prediction Models, Rome Air Development Center Report RADC-TR-78-155, Issue 5, Vol. 1, 1978

[23] GOEL, A.L. and OKUMOTO, K.: « Time Dependent Error Detection Rate Model for Software Reliability and Other Performance Measures », IEEE Trans. Reliability, 1979 pp 206-211

[24] JELINSKI, Z. and MORANDA, P.B.: « Software Reliability Research », in: W. Freiberger (ed.), “Statistical Computer Performance Evaluation”, New York: Academic Press, 1972, pp. 465-484

[25] RUSHFORTH, C.K., STEFFANSON, F.L. and CRAWFORD, A.E.: Software Reliability Estimation under Conditions of Incomplete Information », Rome Air Development Center Report RADCTR-79-230, 1979

[26] LANGBERG, N. and SINGPURWALLAH, N.D.: « A Unification of Some Software Reliability Models via the Bayesian Approach », George Washington University Technical Memo TM-66571, 1981

[27] « Root Cause Analysis Guidance Document », DOE-NE-STD-1404-92, U.S. Department of Energy, Office of Nuclear Energy, Office of Nuclear Safety Policy and Standards, Washington D.C.20585, February 1992

[28] ROONEY, J.J. and VANDEN HEUVEL, L.N.: « Root Cause Analysis for Beginners », Quality Progress, July 2004

[29] CEI 60300-2, *Gestion de la sûreté de fonctionnement – Partie 2: Lignes directrices pour la gestion de la sûreté de fonctionnement*

[30] CEI 60300-3-1, *Gestion de la sûreté de fonctionnement – Partie 3-1: Guide d’application – Techniques d’analyse de la sûreté de fonctionnement – Guide méthodologique*

[31] CEI 60605-4, *Essai de fiabilité des équipements – Partie 4: Méthodes statistiques de distribution exponentielle – Estimateurs ponctuels, intervalles de confiance, intervalles de prédiction et intervalles de tolérance*

[32] CEI 60706-5, *Guide de maintenabilité de matériel – Partie 5-4: Essais pour diagnostic*

[33] CEI 60812, *Techniques d'analyse de la fiabilité du système – Procédure d'analyse des modes de défaillance et de leurs effets (AMDE)*

[34] CEI 61014, *Programmes de croissance de fiabilité*

[35] CEI 61025, *Analyse par arbre de panne (AAP)*

[36] CEI 61078, *Techniques d'analyse pour la sûreté de fonctionnement – Bloc-diagramme de fiabilité et méthodes booléennes*

[37] CEI 61160, *Revue de conception*

[38] Disponible.

[39] CEI 62279, Applications ferroviaires – Systèmes de signalisation, de télécommunication et de traitement – Logiciels pour systèmes de commande et de protection ferroviaire

[40] BS 5760-8: *Reliability of systems, equipment and components. Guide to assessment of reliability of systems containing software*

[41] ISO 9000:2005: *Systèmes de management de la qualité – Principes essentiels et vocabulaire*

[42] IEC 60050-604, *Vocabulaire Electrotechnique International – Chapitre 604: Production, transport et distribution de l'energie electrique – Exploitation*

[43] CEI 60300-1:2003, *Gestion de la sûreté de fonctionnement – Partie 1: Gestion du programme de sûreté de fonctionnement*

[44] CEI 60300-3-15, *Dependability management – Part 3-15: Guidance to engineering of system dependability (en préparation, titre actuellement seulement disponible en anglais)*

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

3, rue de Varembé
P.O. Box 131
CH-1211 Geneva 20
Switzerland

Tel: + 41 22 919 02 11
Fax: + 41 22 919 03 00
info@iec.ch
www.iec.ch