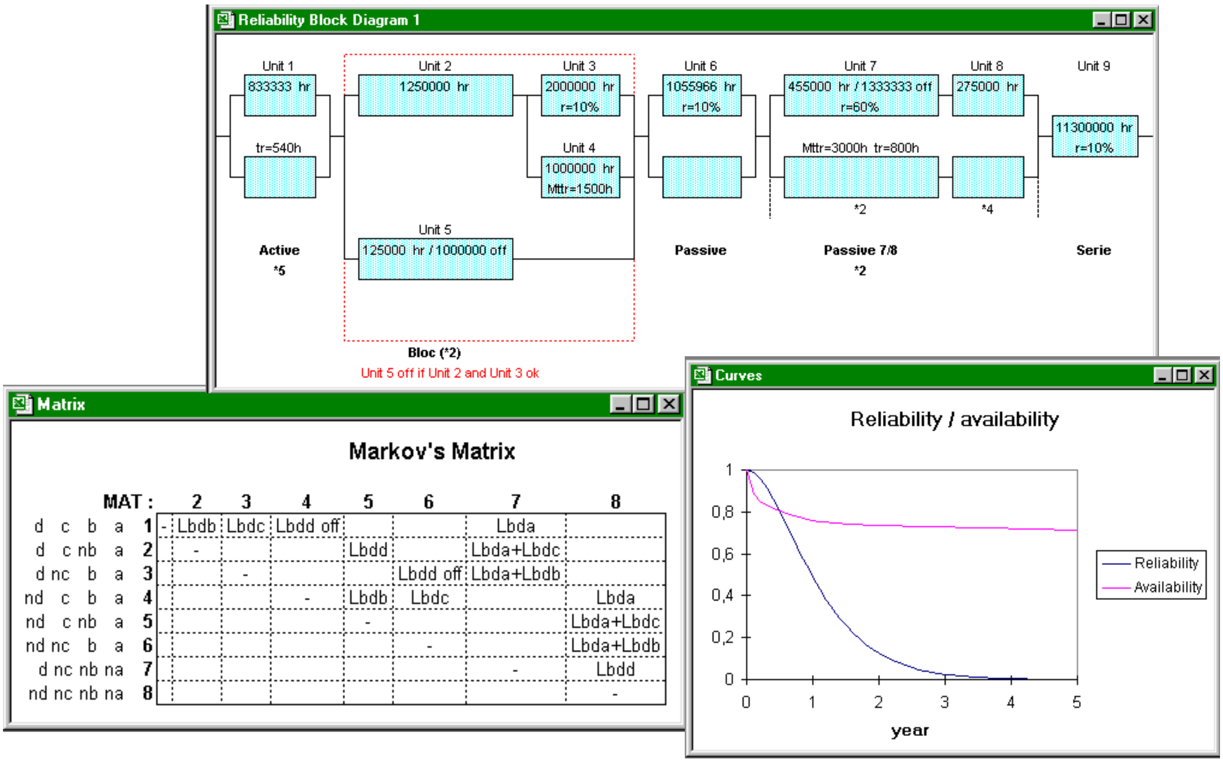
Cab Innovation

3 rue de la Coquille
31500 Toulouse
Tel. 33 (0)5 61 54 68 08
Fax. 33 (0)5 61 54 33 32
Mail : Contact@cabinnovation.com
Web : www.cabinnovation.com

SUPERCAB *Version 14*

using Microsoft EXCEL®



Markov's Matrix

	MAT :		2	3	4	5	6	7	8
d c b a	1	-	Lbdb	Lbdc	Lbdd off			Lbda	
d c nb a	2		-			Lbdd		Lbda+Lbdc	
d nc b a	3			-			Lbdd off	Lbda+Lbdb	
nd c b a	4				-	Lbdb	Lbdc		Lbda
nd c nb a	5					-			Lbda+Lbdc
nd nc b a	6						-		Lbda+Lbdb
d nc nb na	7							-	Lbdd
nd nc nb na	8								-

Reliability / availability & markovian processing

User's Manual

FOREWORD

The software SUPERCAB BASIC version 4 includes some of the SUPERCAB version 14 features. It is not the subject of a specific user manual

The copyright law and international conventions protect the ***SUPERCAB*** software and its User's Manual. Their reproduction or distribution, either wholly or partly, through any means whatsoever, is strictly prohibited. Any person who does not comply with such provisions is committing an offence of forgery and is liable to prosecution and can be sentenced under the provisions prescribed by the law.

The Programming Protection Agency (A.P.P.) references ***SUPERCAB*** at the I.D.D.N. (Inter Deposit Digital Number) index, with the following reference:

IDDN.FR.001.070017.00.R.P.2000.000.20600

CONTENTS :

1 The *SUPERCAB* Software

1.1 General Presentation

SUPERCAB makes available to users, whether they are reliability experts or unspecialised designers, efficient reliability and availability assessment tools :

. An architecture assessment program allowing to compute the reliability and availability of systems from features of their components, draw evolution curves, and draw Block-Diagram automatically.

. A program for processing Markov matrixes, capable of assuming uncontinuous transition rates using the fictitious-state method. In addition to the availability in transient and steady state, this program computes MTTF, MUT, MDT and MTBF.

. A program for assessing evolutionary systems whose state or behaviour is modified at determined times. Such program enables to model limited lifetime elements, maintenance being periodical or limited to working hours and any other type of determinist evolution.

. A logic analyzer enabling to build automatically the Markov matrix of a complex system from logic expressions featuring its operation.

. An approximation function allowing to obtain a Markovian model (or various laws of probabilities) typical of whatsoever transition law. Such a model may be used for representing the behaviour of an element undergoing a wear (curve of bath-type fault rate), or to replace a subassembly in a complex model so as to avoid a combinatory explosion.

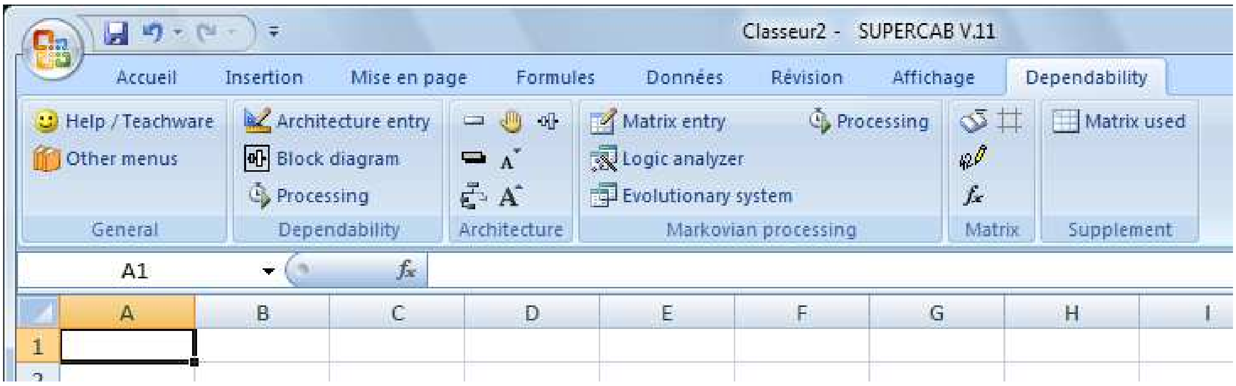
. Reliability/availability calculation functions for elements in redundancy M among N, assuming reconfiguration or repair times, and a program for creating personalized Markovian functions allowing to achieve its own library of various-redundancy calculation function.

1.2 Installing *SUPERCAB* on Hard Disk

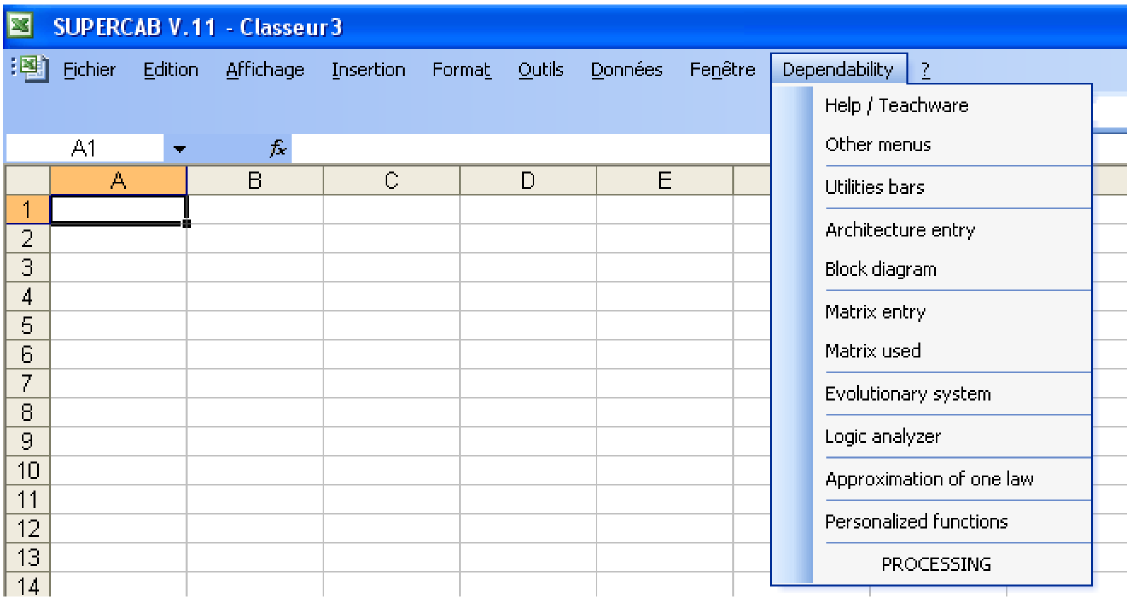
Please comply with instructions shown in CD-ROM.

1.3 To Start *SUPERCAB*

In EXCEL, open SUPERCAB.XLA file.
Software's functionalities are then accessible using menu "Dependability", spreadsheet functionalities remaining always available.



Banner on Excel versions after 2007



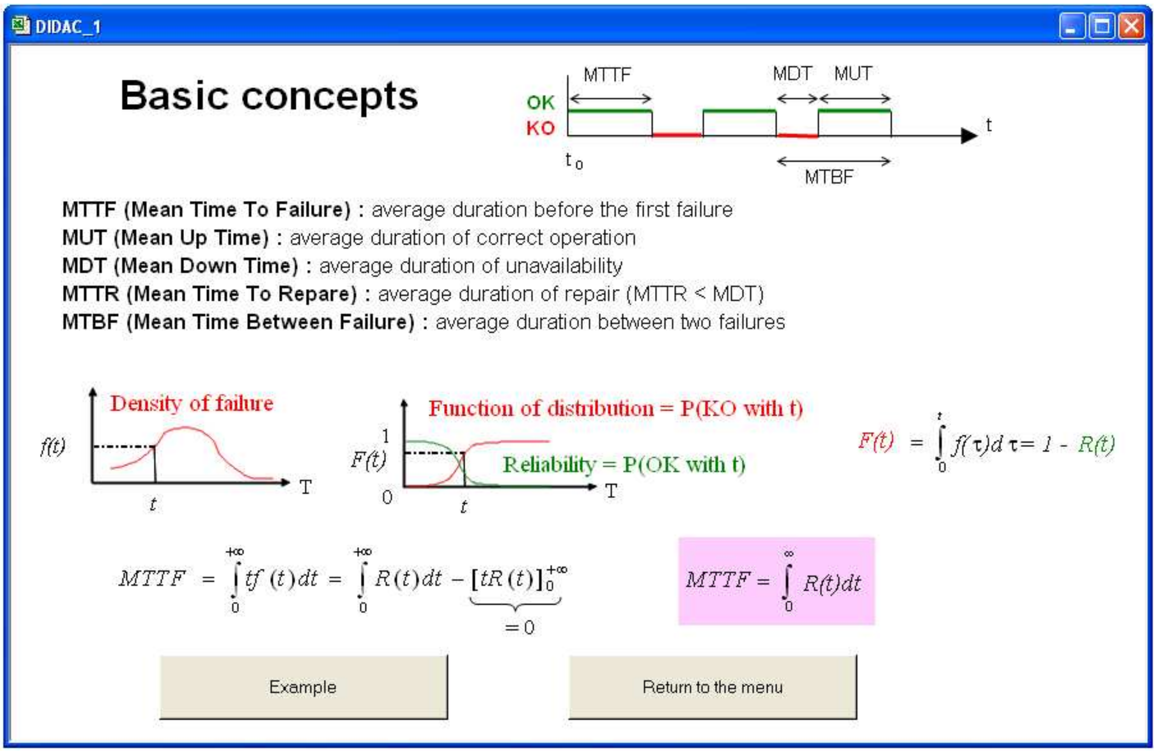
Menu on Excel versions prior to 2007

A help and a teachware are proposed in the menu.

2 Teachware

The teachware presents Markovian process by means of various boards and demonstrations.

2.1 Basic concepts



Basic concepts

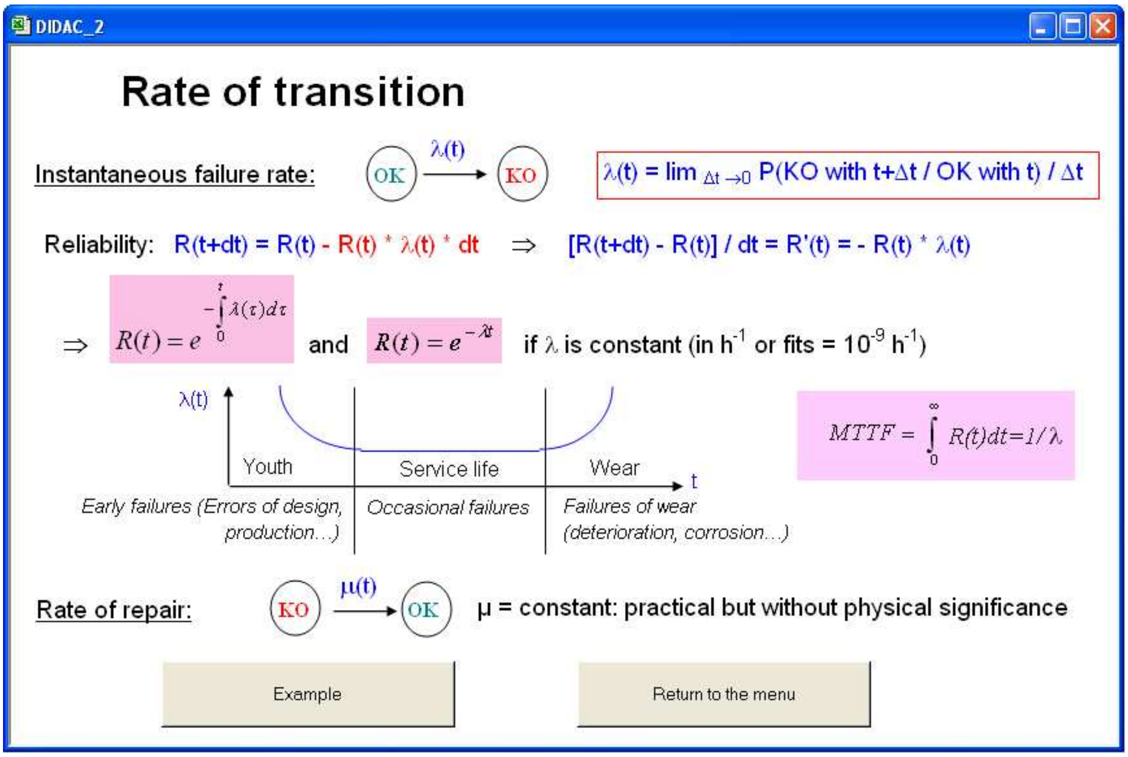
MTTF (Mean Time To Failure) : average duration before the first failure
MUT (Mean Up Time) : average duration of correct operation
MDT (Mean Down Time) : average duration of unavailability
MTTR (Mean Time To Repare) : average duration of repair (MTTR < MDT)
MTBF (Mean Time Between Failure) : average duration between two failures

$$F(t) = \int_0^t f(\tau)d\tau = 1 - R(t)$$

$$MTTF = \int_0^{+\infty} tf(t)dt = \int_0^{+\infty} R(t)dt - \underbrace{[tR(t)]_0^{+\infty}}_{=0}$$

$$MTTF = \int_0^{\infty} R(t)dt$$

2.2 Rate of transition



Rate of transition

Instantaneous failure rate: $\lambda(t) = \lim_{\Delta t \to 0} P(KO \text{ with } t+\Delta t \,/\, OK \text{ with } t) / \Delta t$

Reliability: $R(t+dt) = R(t) - R(t) * \lambda(t) * dt \Rightarrow [R(t+dt) - R(t)] / dt = R'(t) = - R(t) * \lambda(t)$

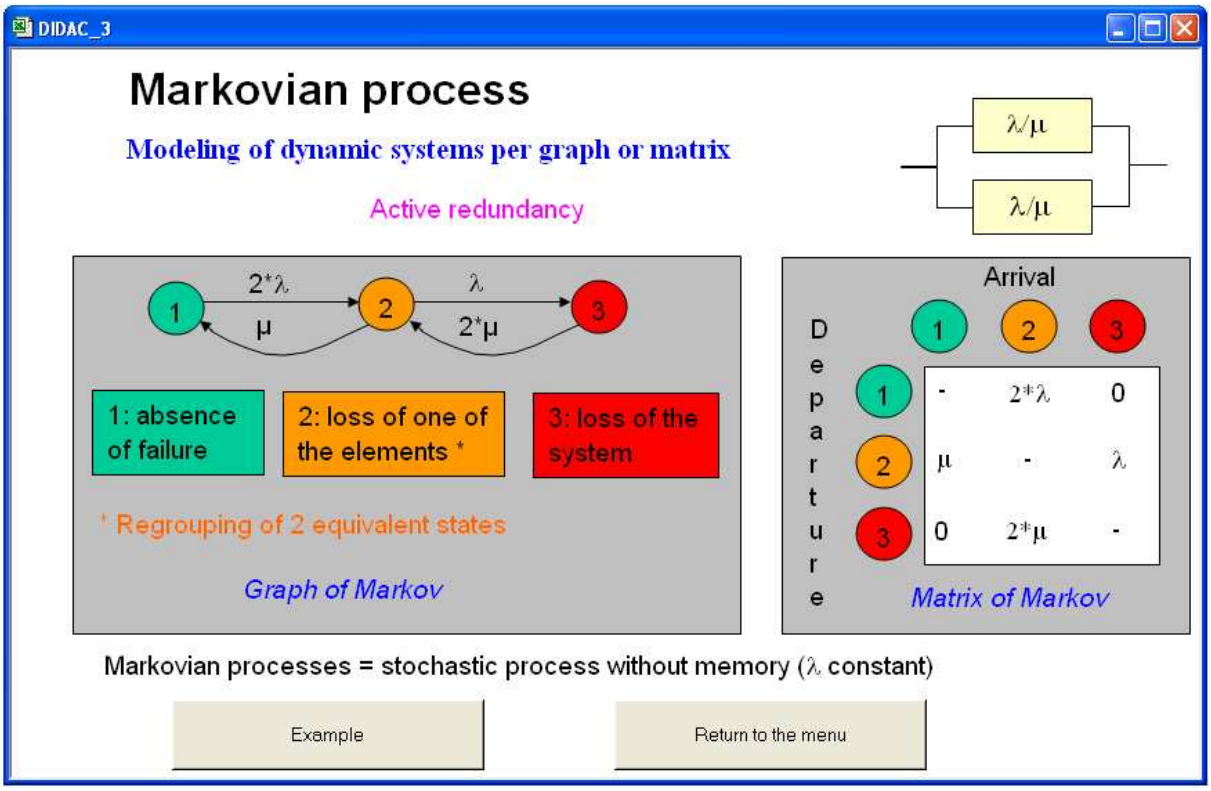
$\Rightarrow$ $R(t) = e^{-\int_0^t \lambda(\tau)d\tau}$ and $R(t) = e^{-\lambda t}$ if λ is constant (in h^{-1} or fits = 10^{-9} h^{-1})

Early failures (Errors of design, production…) | Occasional failures | Failures of wear (deterioration, corrosion…)

$$MTTF = \int_0^{\infty} R(t)dt = 1/\lambda$$

Rate of repair: μ = constant: practical but without physical significance

2.3 Markovian process



Markovian process

Modeling of dynamic systems per graph or matrix

Active redundancy

1: absence of failure

2: loss of one of the elements *

3: loss of the system

* Regrouping of 2 equivalent states

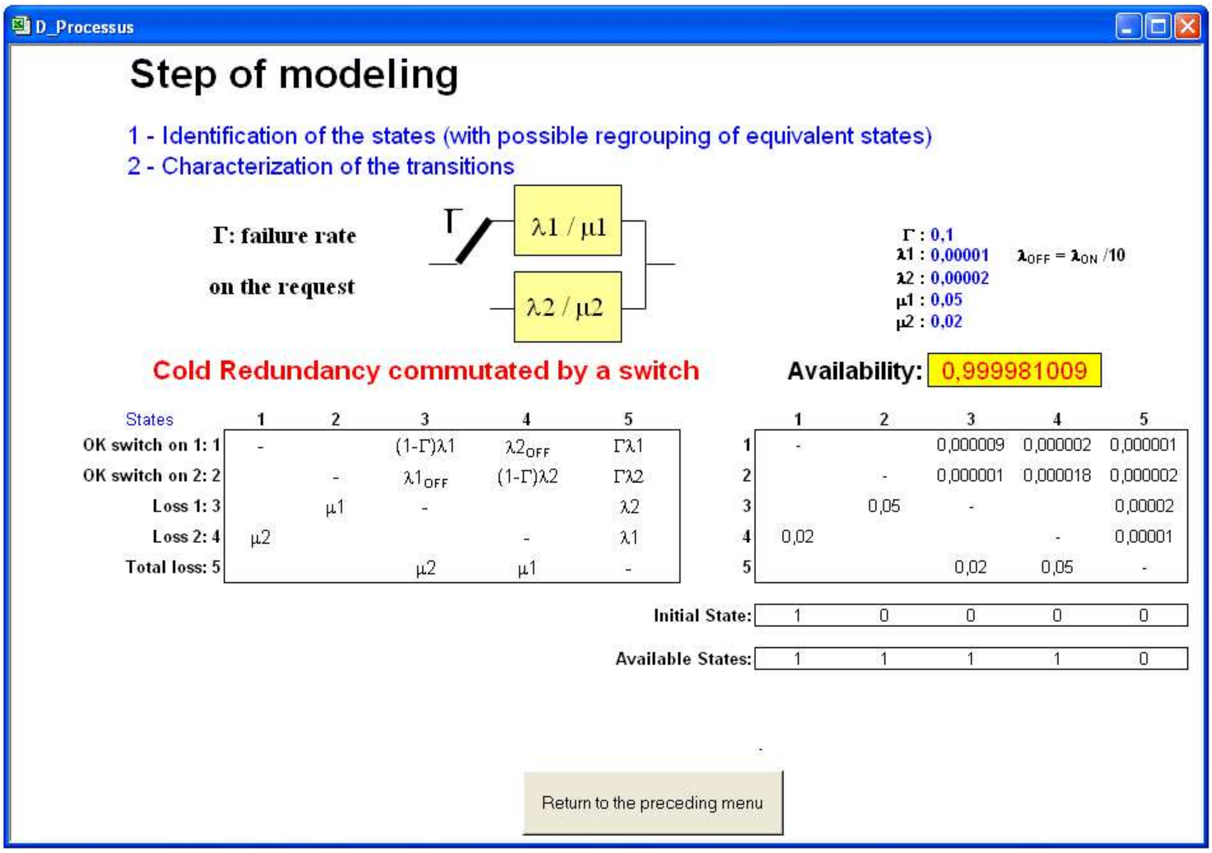
Graph of Markov

Arrival

Departure	1	2	3
1	-	2*λ	0
2	µ	-	λ
3	0	2*µ	-

Matrix of Markov

Markovian processes = stochastic process without memory (λ constant)



Step of modeling

1 - Identification of the states (with possible regrouping of equivalent states)

2 - Characterization of the transitions

Γ: failure rate

on the request

Γ : 0,1
λ1 : 0,00001 **$\lambda_{OFF} = \lambda_{ON}$ /10**
λ2 : 0,00002
µ1 : 0,05
µ2 : 0,02

Cold Redundancy commutated by a switch

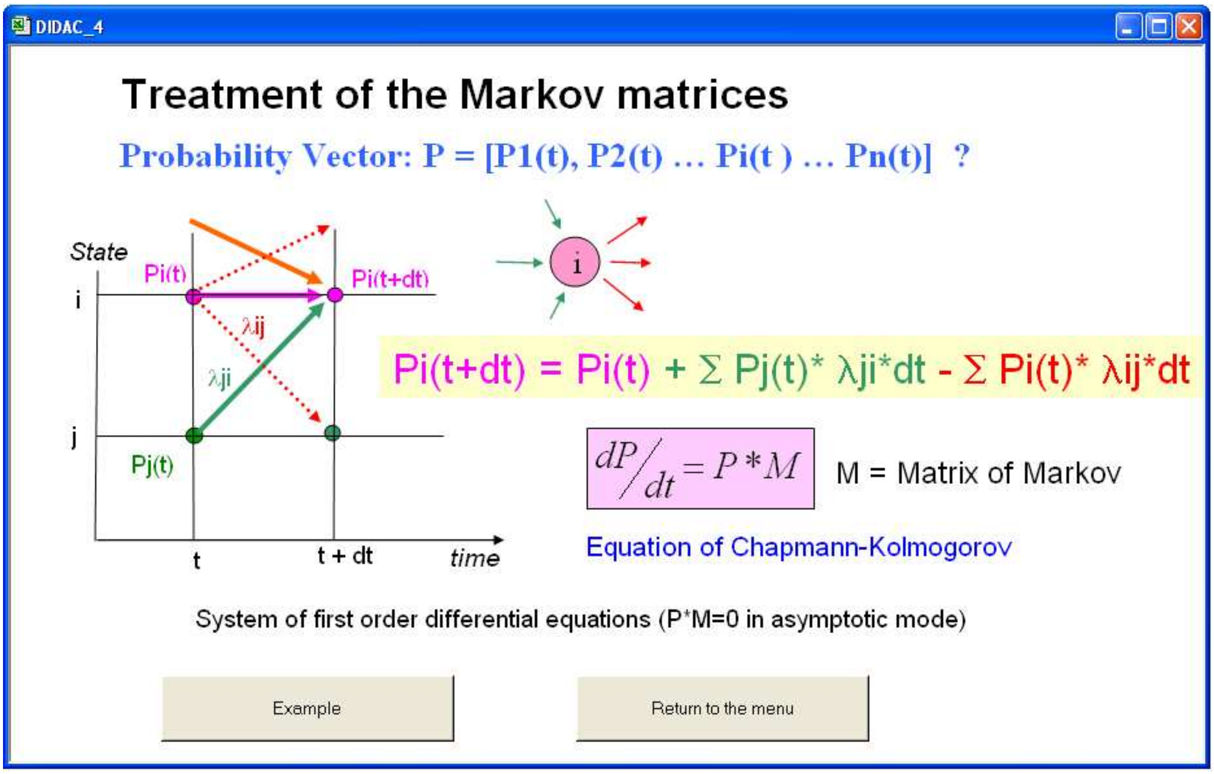
Availability: 0,999981009

States	1	2	3	4	5
OK switch on 1: 1	-		(1-Γ)λ1	$\lambda2_{OFF}$	Γλ1
OK switch on 2: 2		-	$\lambda1_{OFF}$	(1-Γ)λ2	Γλ2
Loss 1: 3		µ1	-		λ2
Loss 2: 4	µ2			-	λ1
Total loss: 5			µ2	µ1	-

	1	2	3	4	5
1	-		0,000009	0,000002	0,000001
2		-	0,000001	0,000018	0,000002
3		0,05	-		0,00002
4	0,02			-	0,00001
5			0,02	0,05	-

Initial State:	1	0	0	0	0
Available States:	1	1	1	1	0

2.4 Treatment of the Markov matrices



Treatment of the Markov matrices

Probability Vector: P = [P1(t), P2(t) … Pi(t) … Pn(t)] ?

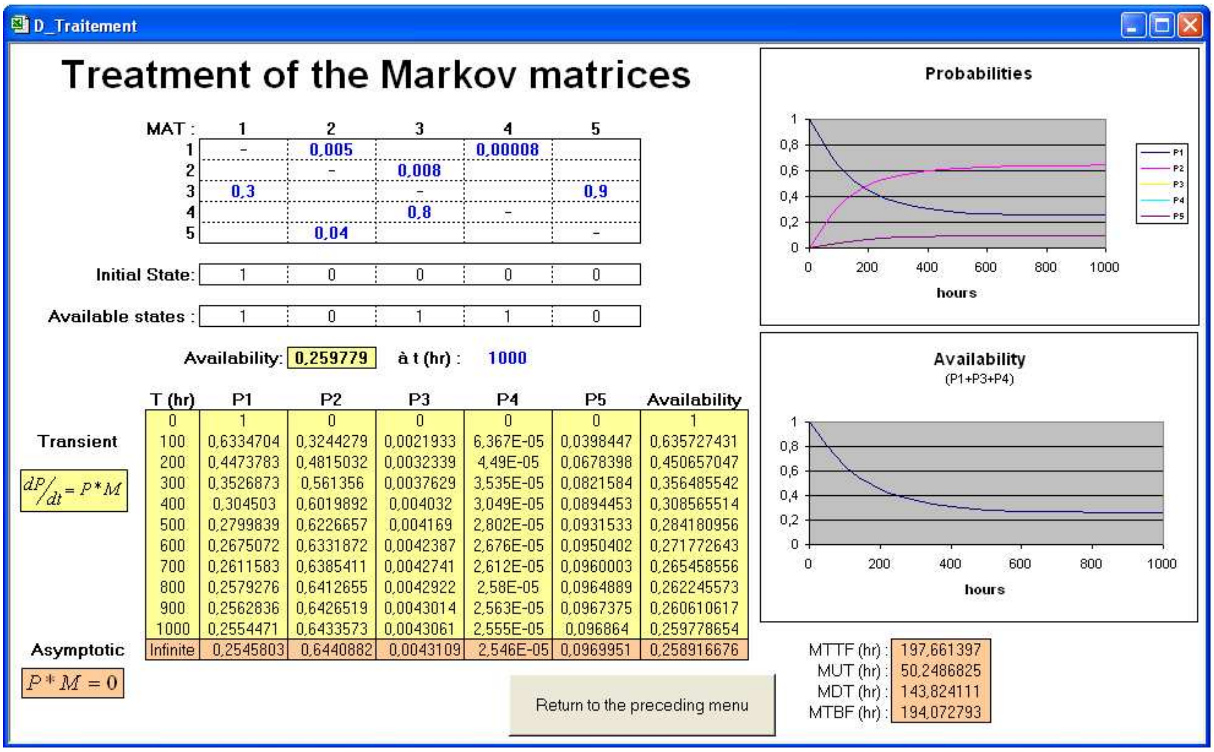
$$Pi(t+dt) = Pi(t) + \Sigma\ Pj(t)^{*}\ \lambda ji^{*}dt - \Sigma\ Pi(t)^{*}\ \lambda ij^{*}dt$$

$$\frac{dP}{dt} = P * M$$

M = Matrix of Markov

Equation of Chapmann-Kolmogorov

System of first order differential equations (P*M=0 in asymptotic mode)



Treatment of the Markov matrices

MAT :	1	2	3	4	5
1	-	0,005		0,00008	
2		-	0,008		
3	0,3		-		0,9
4			0,8	-	
5		0,04			-

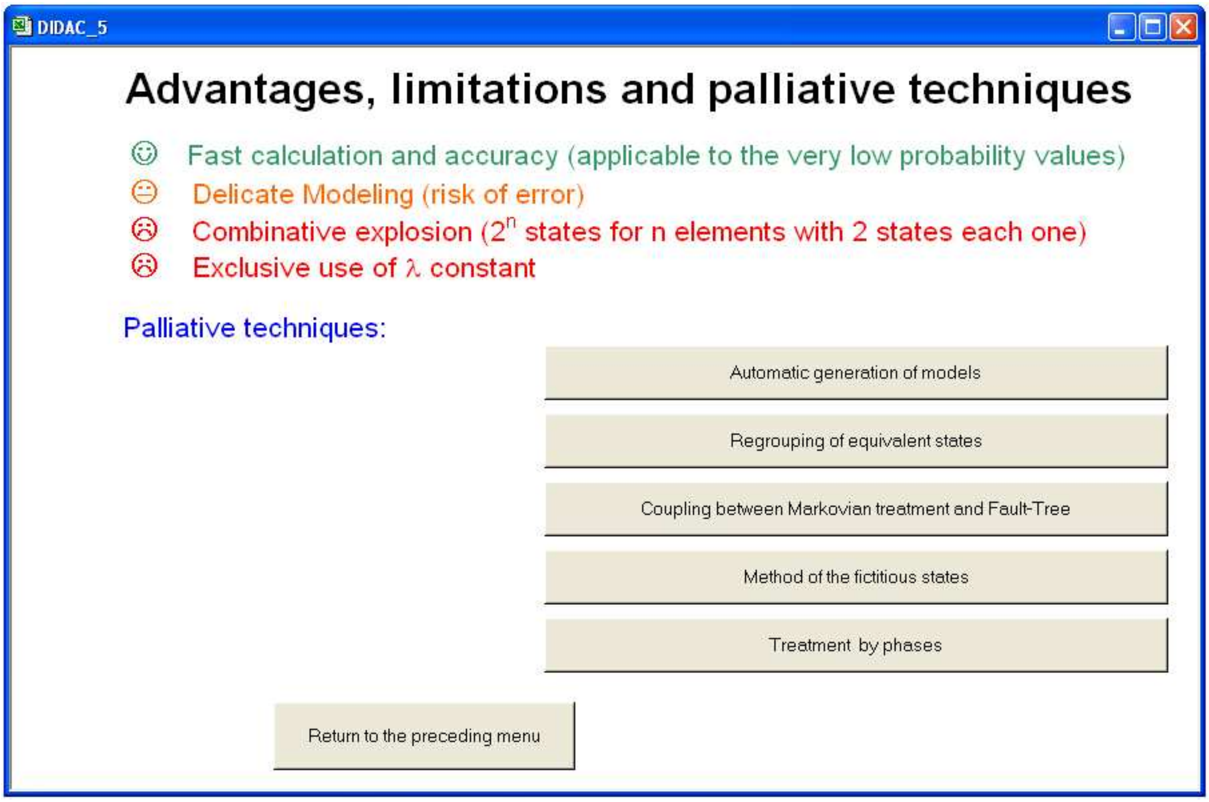
Initial State:	1	0	0	0	0
Available states :	1	0	1	1	0

Availability: 0,259779 à t (hr) : 1000

	T (hr)	P1	P2	P3	P4	P5	Availability
	0	1	0	0	0	0	1
Transient $\frac{dP}{dt} = P*M$	100	0,6334704	0,3244279	0,0021933	6,367E-05	0,0398447	0,635727431
	200	0,4473783	0,4815032	0,0032339	4,49E-05	0,0678398	0,450657047
	300	0,3526873	0,561356	0,0037629	3,535E-05	0,0821584	0,356485542
	400	0,304503	0,6019892	0,004032	3,049E-05	0,0894453	0,308565514
	500	0,2799839	0,6226657	0,004169	2,802E-05	0,0931533	0,284180956
	600	0,2675072	0,6331872	0,0042387	2,676E-05	0,0950402	0,271772643
	700	0,2611583	0,6385411	0,0042741	2,612E-05	0,0960003	0,265458556
	800	0,2579276	0,6412655	0,0042922	2,58E-05	0,0964889	0,262245573
	900	0,2562836	0,6426519	0,0043014	2,563E-05	0,0967375	0,260610617
	1000	0,2554471	0,6433573	0,0043061	2,555E-05	0,096864	0,259778654
Asymptotic $P*M=0$	Infinite	0,2545803	0,6440882	0,0043109	2,546E-05	0,0969951	0,258916676

MTTF (hr) :	197,661397
MUT (hr) :	50,2486825
MDT (hr) :	143,824111
MTBF (hr) :	194,072793

2.5 Advantages, limitations and palliative solutions



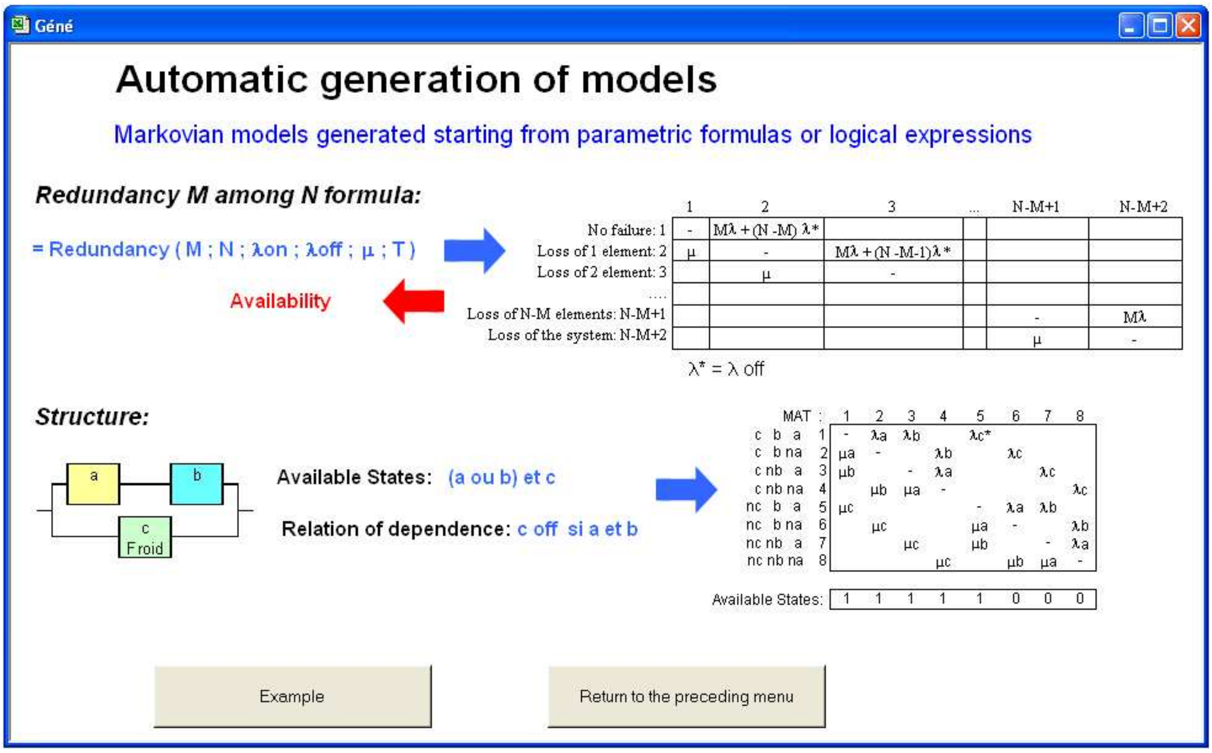
Advantages, limitations and palliative techniques

- ☺ Fast calculation and accuracy (applicable to the very low probability values)
- 😐 Delicate Modeling (risk of error)
- ☹ Combinative explosion (2^n states for n elements with 2 states each one)
- ☹ Exclusive use of λ constant

Palliative techniques:

Automatic generation of models
Regrouping of equivalent states
Coupling between Markovian treatment and Fault-Tree
Method of the fictitious states
Treatment by phases
Return to the preceding menu

2.5.1 Automatic generation of models



Automatic generation of models

Markovian models generated starting from parametric formulas or logical expressions

Redundancy M among N formula:

= Redundancy (M ; N ; λon ; λoff ; μ ; T)

Availability

	1	2	3	...	N-M+1	N-M+2
No failure: 1	-	Mλ + (N -M) λ*				
Loss of 1 element: 2	μ	-	Mλ + (N -M-1)λ*			
Loss of 2 element: 3		μ	-			
....						
Loss of N-M elements: N-M+1					-	Mλ
Loss of the system: N-M+2					μ	-

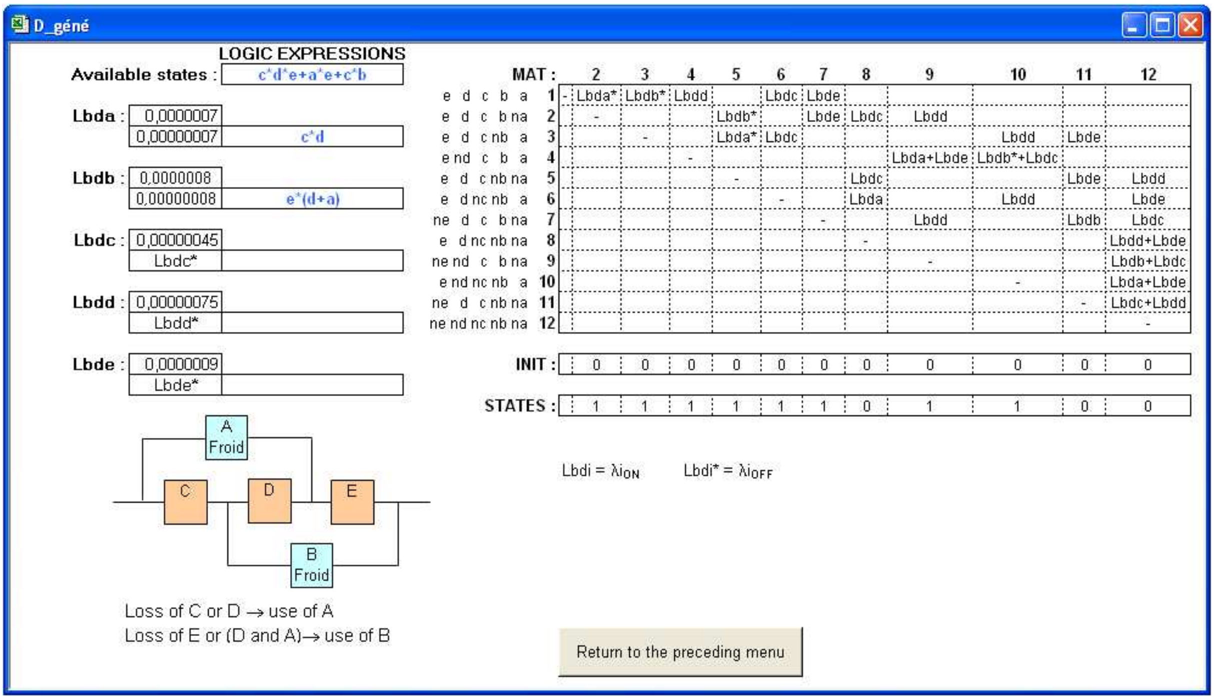
λ* = λ off

Structure:

Available States: (a ou b) et c

Relation of dependence: c off si a et b

MAT :		1	2	3	4	5	6	7	8
c b a	1	-	λa	λb		λc*			
c b na	2	μa	-		λb		λc		
c nb a	3	μb		-	λa			λc	
c nb na	4		μb	μa	-				λc
nc b a	5	μc				-	λa	λb	
nc b na	6		μc			μa	-		λb
nc nb a	7			μc		μb		-	λa
nc nb na	8				μc		μb	μa	-
Available States:		1	1	1	1	1	0	0	0



LOGIC EXPRESSIONS

Available states : c*d*e+a*e+c*b

Lbda : 0,0000007
0,00000007 c*d

Lbdb : 0,0000008
0,00000008 e*(d+a)

Lbdc : 0,00000045
Lbdc*

Lbdd : 0,00000075
Lbdd*

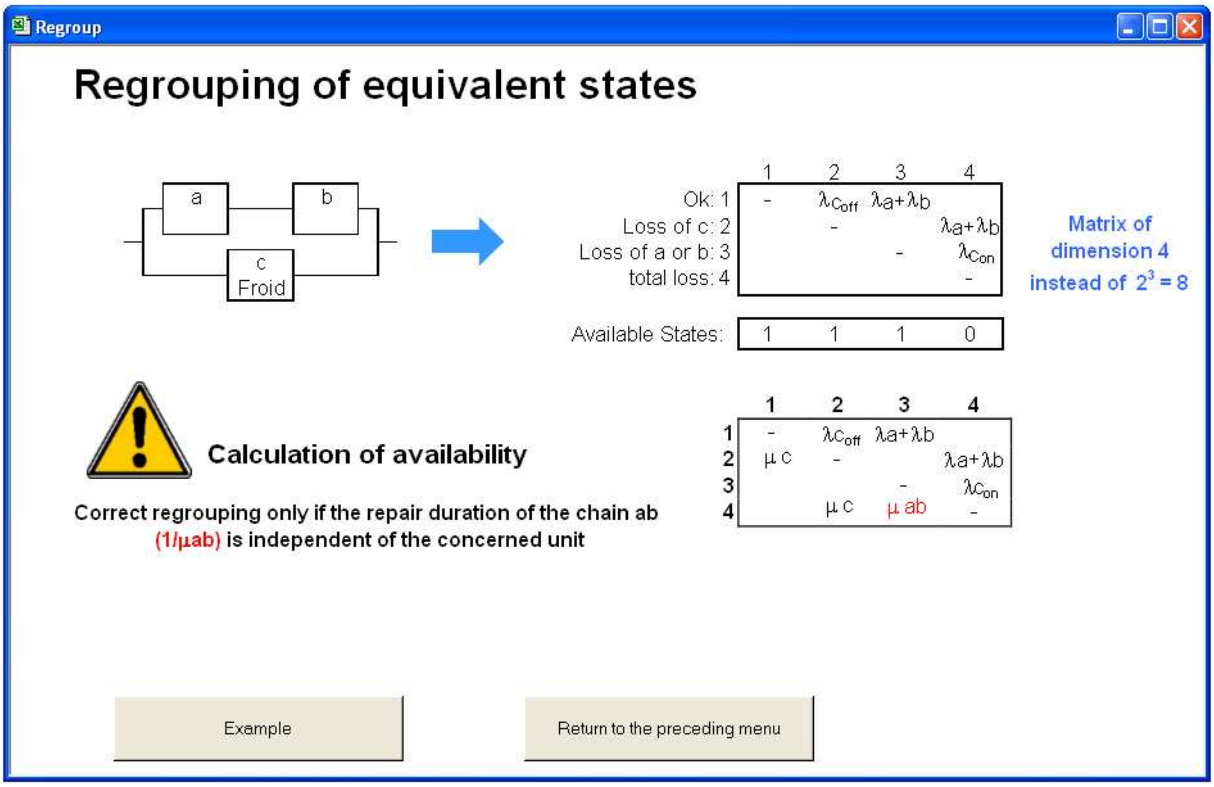
Lbde : 0,0000009
Lbde*

MAT :			2	3	4	5	6	7	8	9	10	11	12
e d c b a	1	-	Lbda*	Lbdb*	Lbdd		Lbdc	Lbde					
e d c b na	2		-			Lbdb*		Lbde	Lbdc	Lbdd			
e d c nb a	3			-		Lbda*	Lbdc				Lbdd	Lbde	
e nd c b a	4				-					Lbda+Lbde	Lbdb*+Lbdc		
e d c nb na	5					-			Lbdc			Lbde	Lbdd
e d nc nb a	6						-		Lbda		Lbdd		Lbde
ne d c b na	7							-		Lbdd		Lbdb	Lbdc
e d nc nb na	8								-				Lbdd+Lbde
ne nd c b na	9									-			Lbdb+Lbdc
e nd nc nb a	10										-		Lbda+Lbde
ne d c nb na	11											-	Lbdc+Lbdd
ne nd nc nb na	12												-
INIT :			0	0	0	0	0	0	0	0	0	0	0
STATES :			1	1	1	1	1	1	0	1	1	0	0

Lbdi = λi_{ON} Lbdi* = λi_{OFF}

Loss of C or D → use of A
Loss of E or (D and A)→ use of B

2.5.2 Regrouping of equivalent states



Regrouping of equivalent states

	1	2	3	4
Ok: 1	-	λC_{off}	$\lambda a+\lambda b$	
Loss of c: 2		-		$\lambda a+\lambda b$
Loss of a or b: 3			-	λC_{on}
total loss: 4				-

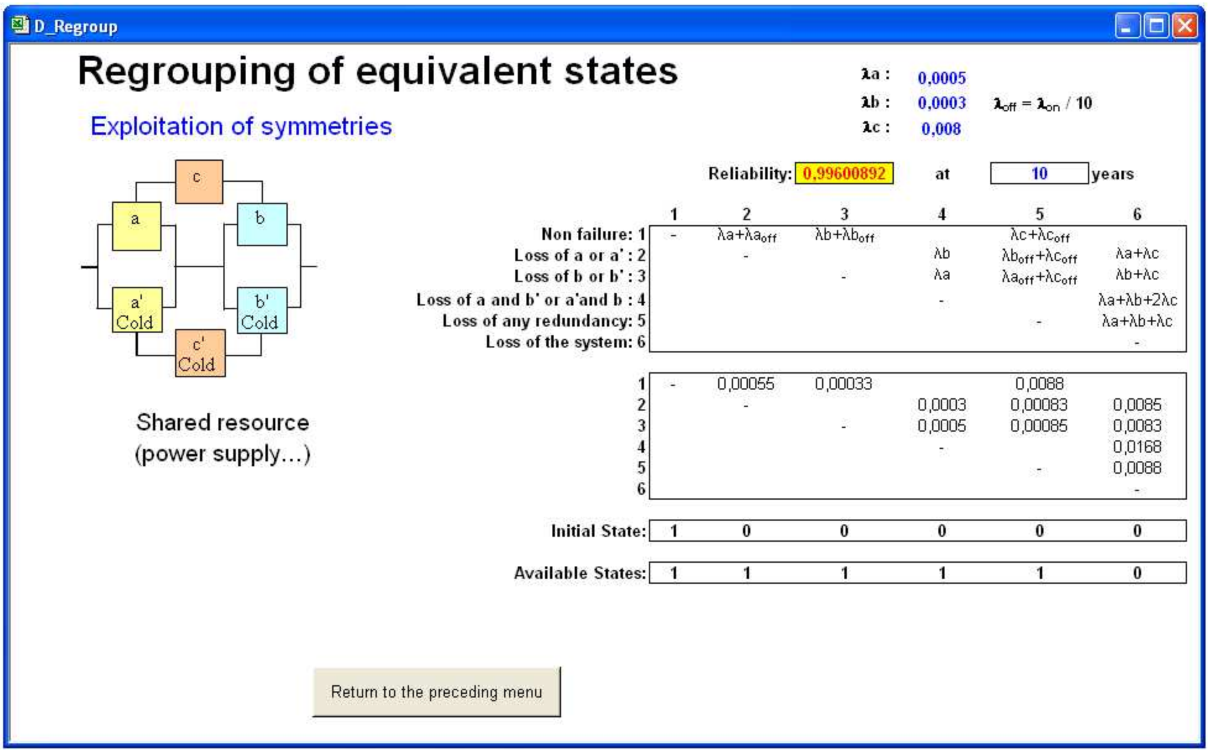
Matrix of dimension 4 instead of $2^3 = 8$

	1	2	3	4
Available States:	1	1	1	0

Calculation of availability

Correct regrouping only if the repair duration of the chain ab (1/μab) is independent of the concerned unit

	1	2	3	4
1	-	λC_{off}	$\lambda a+\lambda b$	
2	μ c	-		$\lambda a+\lambda b$
3			-	λC_{on}
4		μ c	μ ab	-



Regrouping of equivalent states

Exploitation of symmetries

λa : 0,0005

λb : 0,0003 $\lambda_{off} = \lambda_{on} / 10$

λc : 0,008

Shared resource (power supply…)

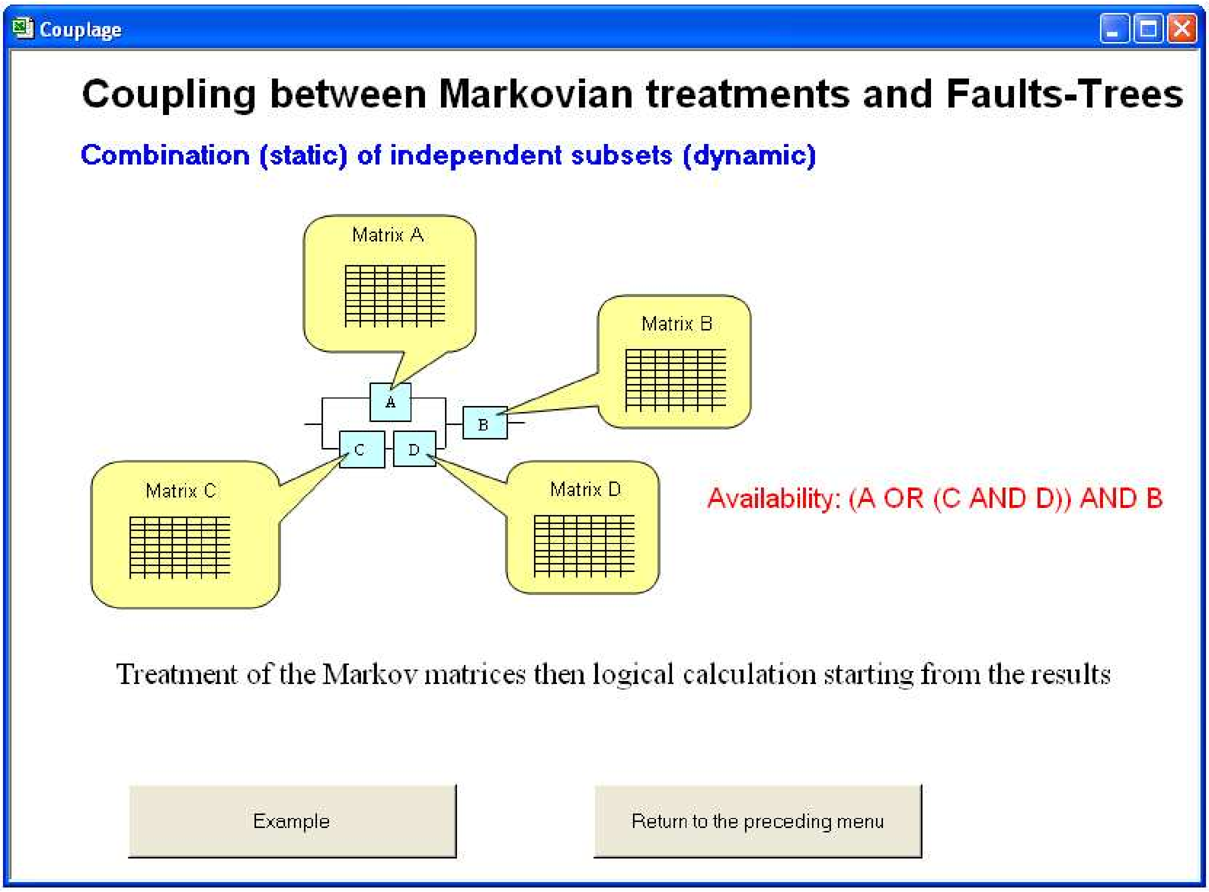
Reliability: 0,99600892 at 10 years

	1	2	3	4	5	6
Non failure: 1	-	$\lambda a+\lambda a_{off}$	$\lambda b+\lambda b_{off}$		$\lambda c+\lambda c_{off}$	
Loss of a or a' : 2		-		λb	$\lambda b_{off}+\lambda c_{off}$	$\lambda a+\lambda c$
Loss of b or b' : 3			-	λa	$\lambda a_{off}+\lambda c_{off}$	$\lambda b+\lambda c$
Loss of a and b' or a'and b : 4				-		$\lambda a+\lambda b+2\lambda c$
Loss of any redundancy: 5					-	$\lambda a+\lambda b+\lambda c$
Loss of the system: 6						-

1	-	0,00055	0,00033		0,0088	
2		-		0,0003	0,00083	0,0085
3			-	0,0005	0,00085	0,0083
4				-		0,0168
5					-	0,0088
6						-

Initial State:	1	0	0	0	0	0
Available States:	1	1	1	1	1	0

2.5.3 Coupling between Markovian treatments and Faults-Trees

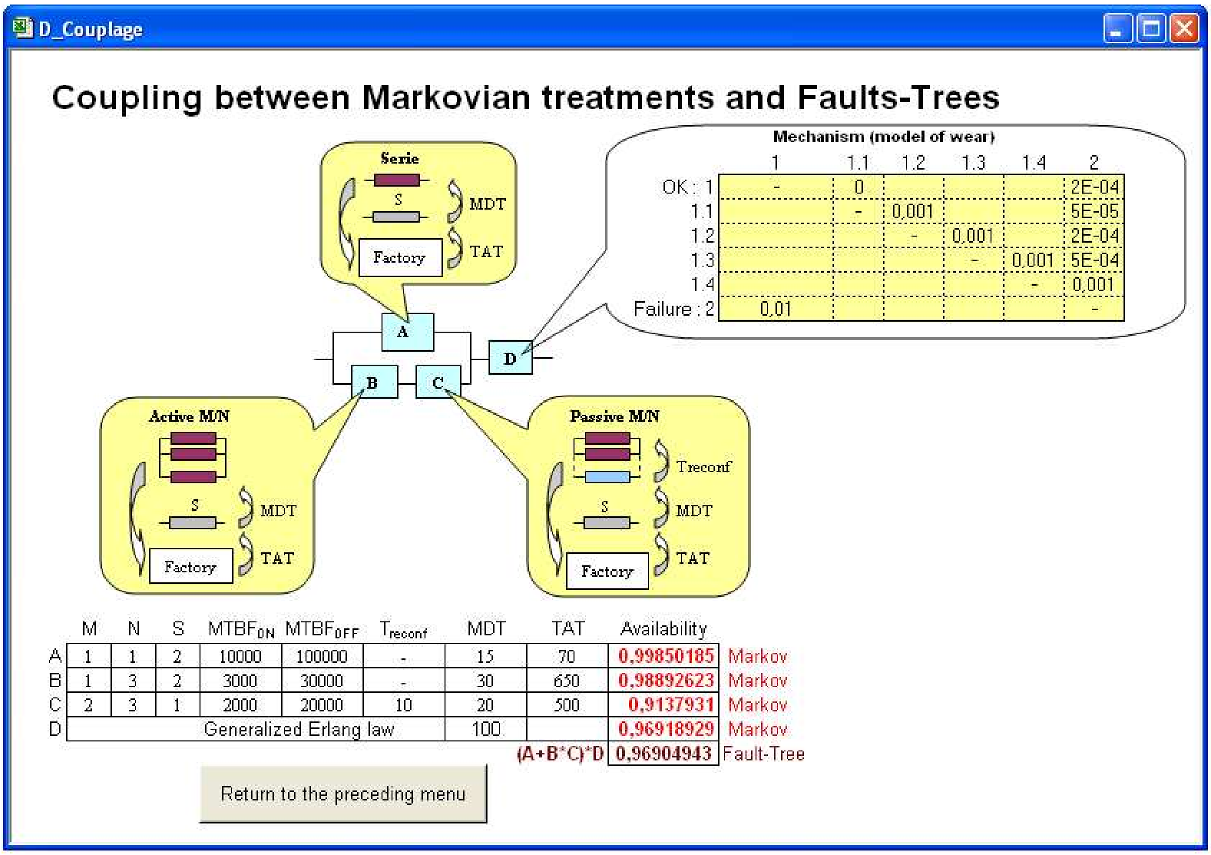


Coupling between Markovian treatments and Faults-Trees

Combination (static) of independent subsets (dynamic)

Availability: (A OR (C AND D)) AND B

Treatment of the Markov matrices then logical calculation starting from the results



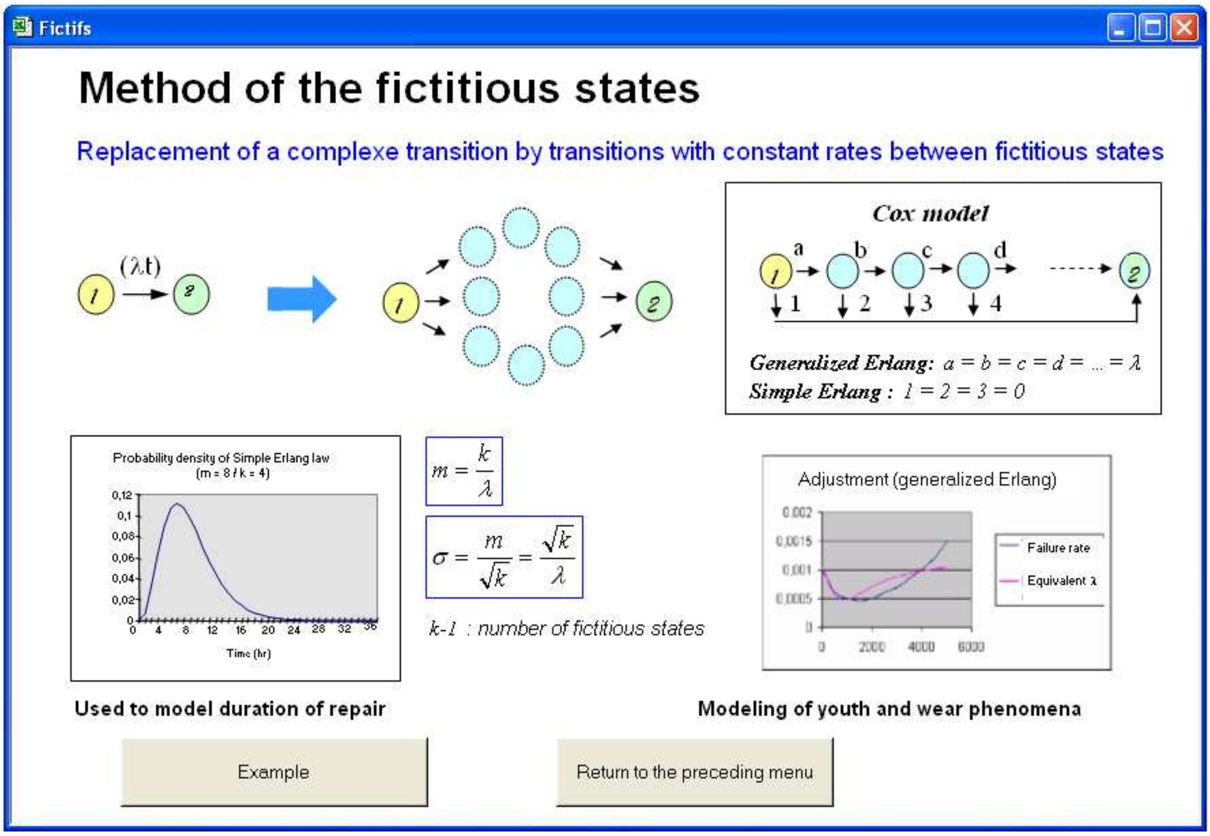
Coupling between Markovian treatments and Faults-Trees

Mechanism (model of wear)

	1	1.1	1.2	1.3	1.4	2
OK : 1	-	0				2E-04
1.1		-	0,001			5E-05
1.2			-	0,001		2E-04
1.3				-	0,001	5E-04
1.4					-	0,001
Failure : 2	0,01					-

	M	N	S	$MTBF_{ON}$	$MTBF_{OFF}$	T_{reconf}	MDT	TAT	Availability	
A	1	1	2	10000	100000	-	15	70	0,99850185	Markov
B	1	3	2	3000	30000	-	30	650	0,98892623	Markov
C	2	3	1	2000	20000	10	20	500	0,9137931	Markov
D	Generalized Erlang law						100		0,96918929	Markov
								(A+B*C)*D	0,96904943	Fault-Tree

2.5.4 Method of the fictitious states



Method of the fictitious states

Replacement of a complexe transition by transitions with constant rates between fictitious states

Generalized Erlang: $a = b = c = d = ... = \lambda$

Simple Erlang : $1 = 2 = 3 = 0$

$$m = \frac{k}{\lambda}$$

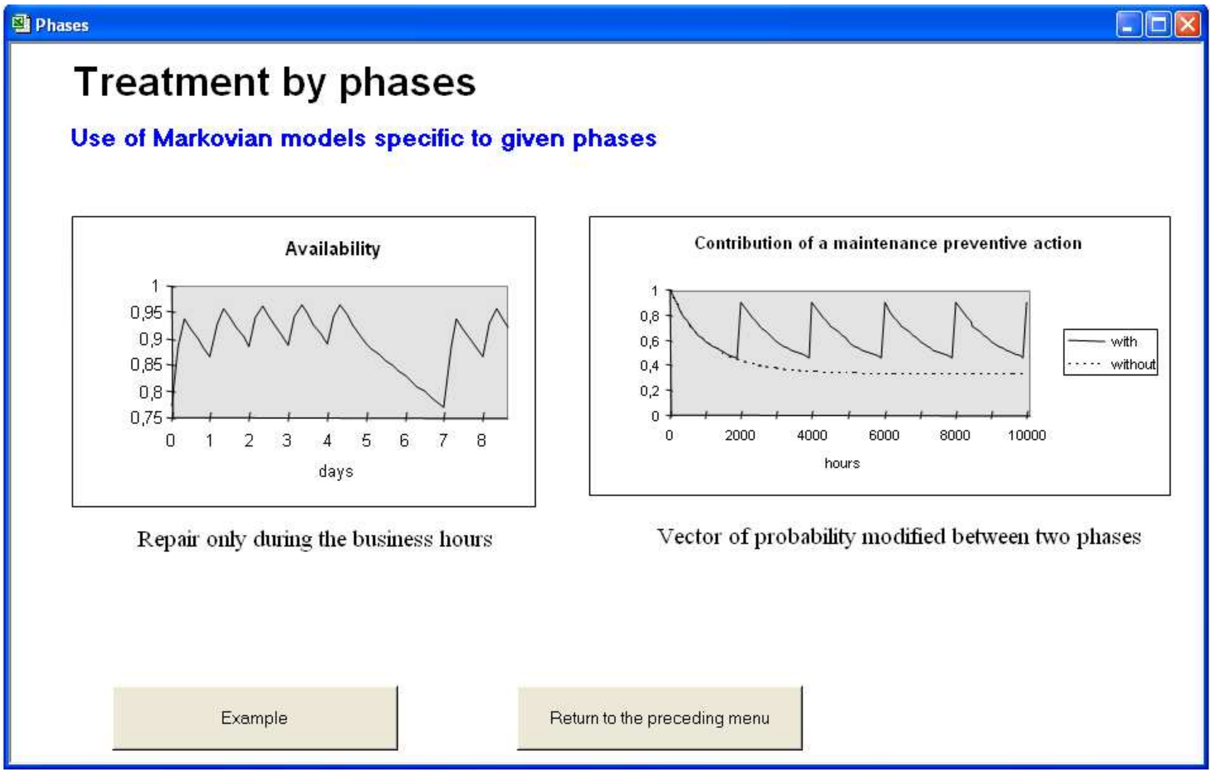
$$\sigma = \frac{m}{\sqrt{k}} = \frac{\sqrt{k}}{\lambda}$$

k-1 : number of fictitious states

Used to model duration of repair

Modeling of youth and wear phenomena

2.5.5 Treatment by phases



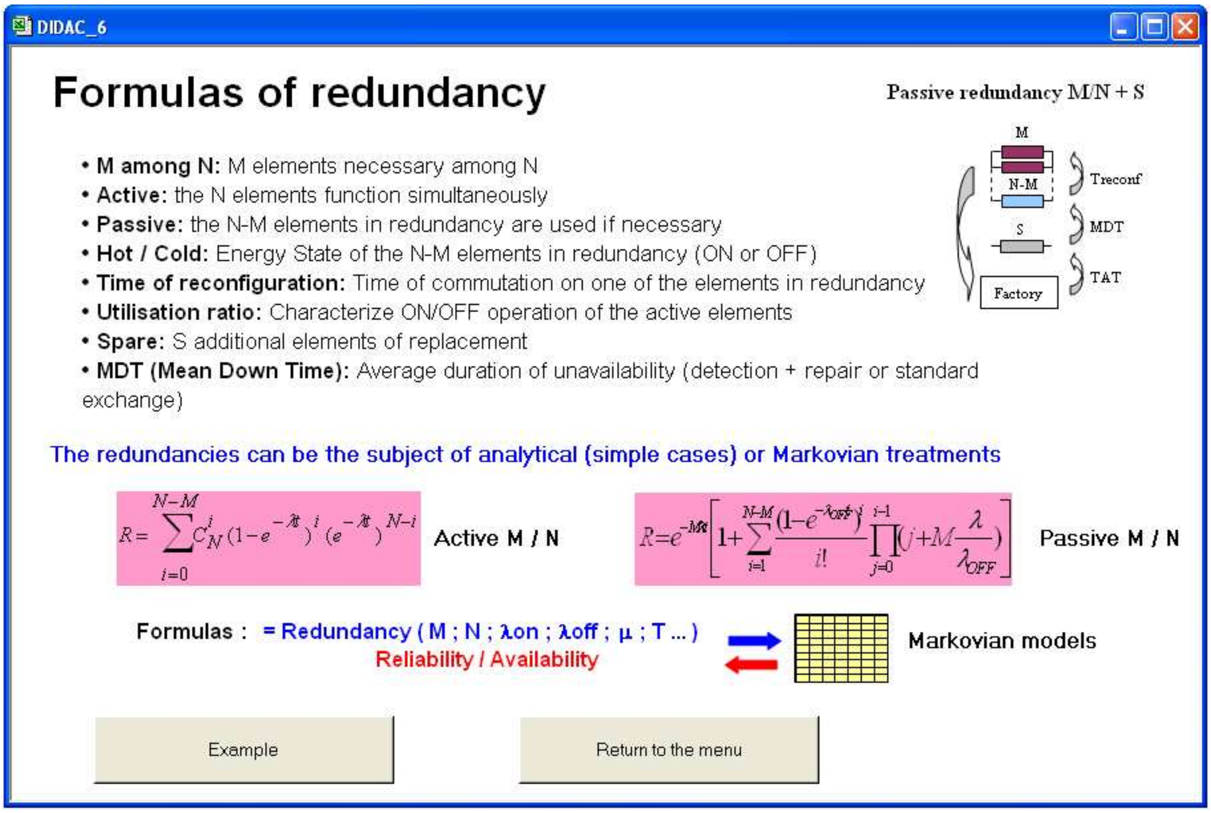
Treatment by phases

Use of Markovian models specific to given phases

Repair only during the business hours

Vector of probability modified between two phases

2.6 Formulas of redundancy



Formulas of redundancy

Passive redundancy M/N + S

- **M among N:** M elements necessary among N
- **Active:** the N elements function simultaneously
- **Passive:** the N-M elements in redundancy are used if necessary
- **Hot / Cold:** Energy State of the N-M elements in redundancy (ON or OFF)
- **Time of reconfiguration:** Time of commutation on one of the elements in redundancy
- **Utilisation ratio:** Characterize ON/OFF operation of the active elements
- **Spare:** S additional elements of replacement
- **MDT (Mean Down Time):** Average duration of unavailability (detection + repair or standard exchange)

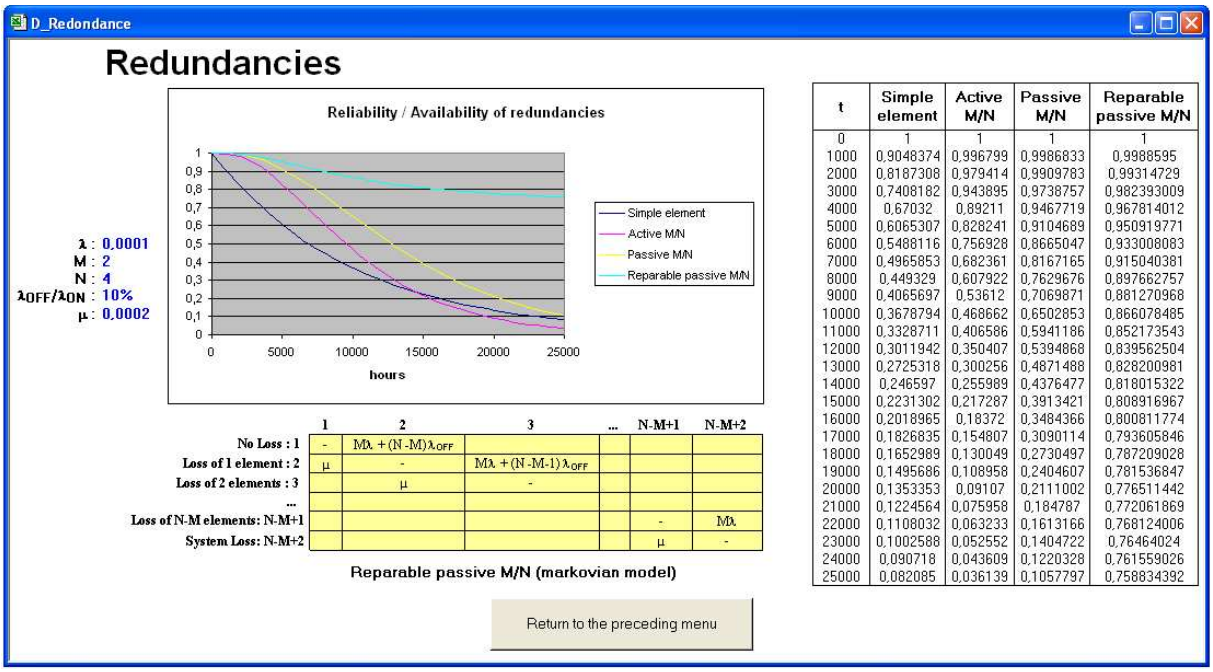
The redundancies can be the subject of analytical (simple cases) or Markovian treatments

$$R=\sum_{i=0}^{N-M} C_N^i (1-e^{-\lambda t})^i (e^{-\lambda t})^{N-i}$$ Active M / N

$$R=e^{-M\lambda t}\left[1+\sum_{i=1}^{N-M}\frac{(1-e^{-\lambda_{OFF} t})^i}{i!}\prod_{j=0}^{i-1}\left(j+M\frac{\lambda}{\lambda_{OFF}}\right)\right]$$ Passive M / N

Formulas : = Redundancy (M ; N ; λon ; λoff ; μ ; T ...)
Reliability / Availability

Markovian models



Redundancies

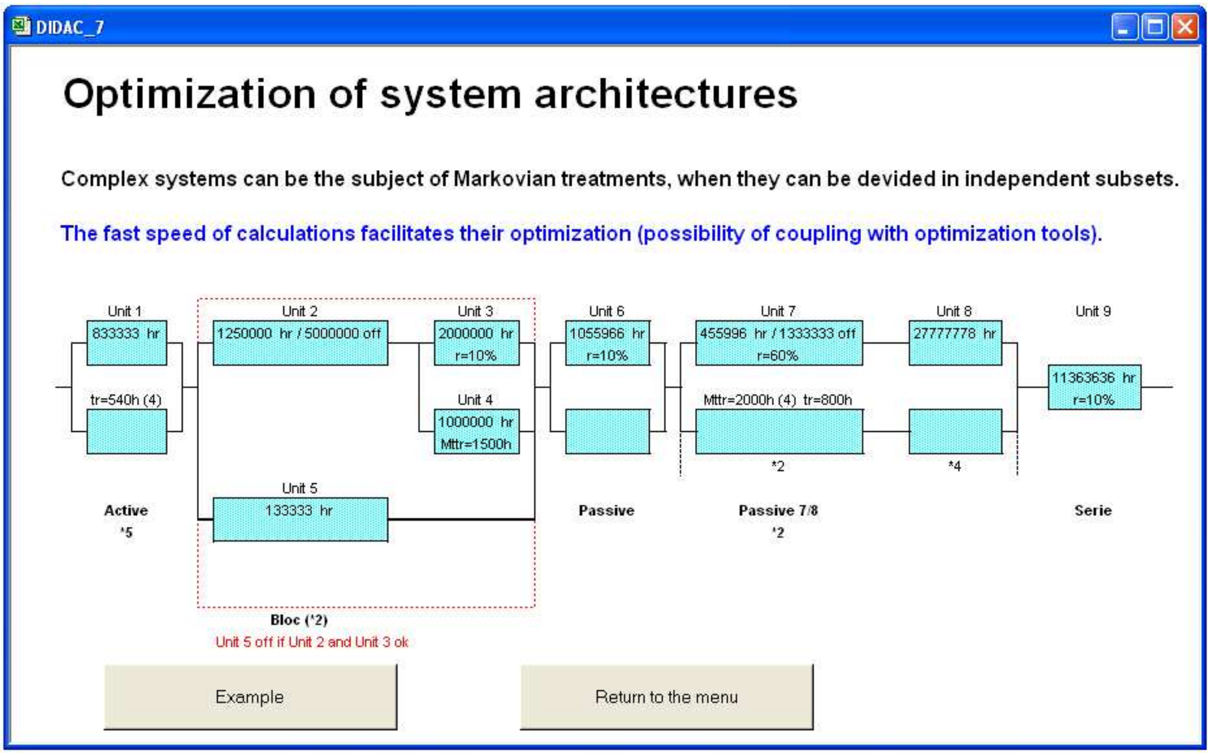
λ : 0,0001
M : 2
N : 4
λOFF/λON : 10%
μ : 0,0002

	1	2	3	...	N-M+1	N-M+2
No Loss : 1	-	$M\lambda + (N-M)\lambda_{OFF}$				
Loss of 1 element : 2	μ	-	$M\lambda + (N-M-1)\lambda_{OFF}$			
Loss of 2 elements : 3		μ	-			
...						
Loss of N-M elements: N-M+1					-	Mλ
System Loss: N-M+2					μ	-

Reparable passive M/N (markovian model)

t	Simple element	Active M/N	Passive M/N	Reparable passive M/N
0	1	1	1	1
1000	0,9048374	0,996799	0,9986833	0,9988595
2000	0,8187308	0,979414	0,9909783	0,99314729
3000	0,7408182	0,943895	0,9738757	0,982393009
4000	0,67032	0,89211	0,9467719	0,967814012
5000	0,6065307	0,828241	0,9104689	0,950919771
6000	0,5488116	0,756928	0,8665047	0,933008083
7000	0,4965853	0,682361	0,8167165	0,915040381
8000	0,449329	0,607922	0,7629676	0,897662757
9000	0,4065697	0,53612	0,7069871	0,881270968
10000	0,3678794	0,468662	0,6502853	0,866078485
11000	0,3328711	0,406586	0,5941186	0,852173543
12000	0,3011942	0,350407	0,5394868	0,839562504
13000	0,2725318	0,300256	0,4871488	0,828200981
14000	0,246597	0,255989	0,4376477	0,818015322
15000	0,2231302	0,217287	0,3913421	0,808916967
16000	0,2018965	0,18372	0,3484366	0,800811774
17000	0,1826835	0,154807	0,3090114	0,793605846
18000	0,1652989	0,130049	0,2730497	0,787209028
19000	0,1495686	0,108958	0,2404607	0,781536847
20000	0,1353353	0,09107	0,2111002	0,776511442
21000	0,1224564	0,075958	0,184787	0,772061869
22000	0,1108032	0,063233	0,1613166	0,768124006
23000	0,1002588	0,052552	0,1404722	0,76464024
24000	0,090718	0,043609	0,1220328	0,761559026
25000	0,082085	0,036139	0,1057797	0,758834392

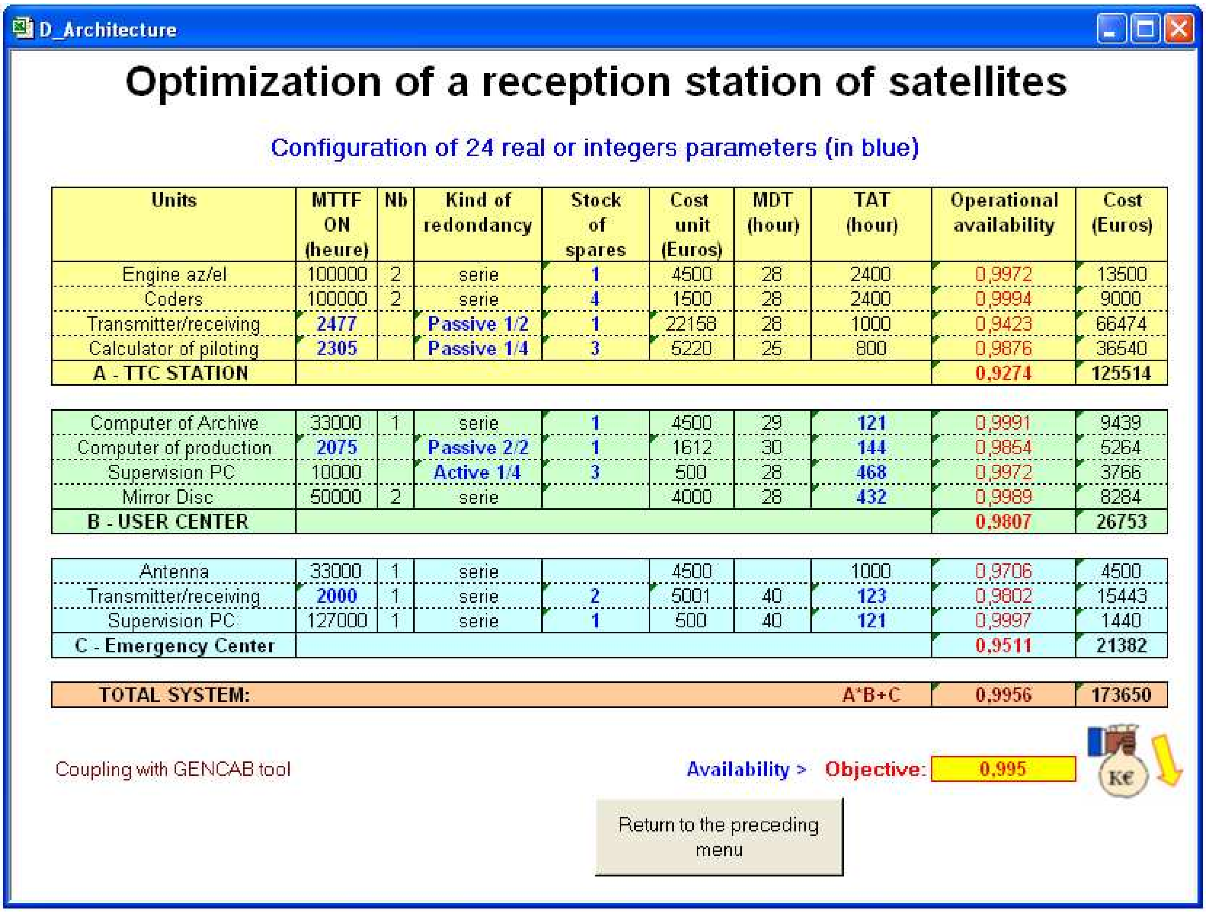
2.7 Optimization of system architectures



Optimization of system architectures

Complex systems can be the subject of Markovian treatments, when they can be devided in independent subsets.

The fast speed of calculations facilitates their optimization (possibility of coupling with optimization tools).



Optimization of a reception station of satellites

Configuration of 24 real or integers parameters (in blue)

Units	MTTF ON (heure)	Nb	Kind of redondancy	Stock of spares	Cost unit (Euros)	MDT (hour)	TAT (hour)	Operational availability	Cost (Euros)
Engine az/el	100000	2	serie	1	4500	28	2400	0,9972	13500
Coders	100000	2	serie	4	1500	28	2400	0,9994	9000
Transmitter/receiving	2477		Passive 1/2	1	22158	28	1000	0,9423	66474
Calculator of piloting	2305		Passive 1/4	3	5220	25	800	0,9876	36540
A - TTC STATION								**0,9274**	**125514**
Computer of Archive	33000	1	serie	1	4500	29	121	0,9991	9439
Computer of production	2075		Passive 2/2	1	1612	30	144	0,9854	5264
Supervision PC	10000		Active 1/4	3	500	28	468	0,9972	3766
Mirror Disc	50000	2	serie		4000	28	432	0,9989	8284
B - USER CENTER								**0,9807**	**26753**
Antenna	33000	1	serie		4500		1000	0,9706	4500
Transmitter/receiving	2000	1	serie	2	5001	40	123	0,9802	15443
Supervision PC	127000	1	serie	1	500	40	121	0,9997	1440
C - Emergency Center								**0,9511**	**21382**
TOTAL SYSTEM:							A*B+C	**0,9956**	**173650**

Coupling with GENCAB tool

Availability > **Objective:** **0,995**

3 Architecture Assessment

SUPERCAB calculates the reliability and availability of one architecture, from the features of its components preliminarily entered in one table, and draws the block-diagram automatically:

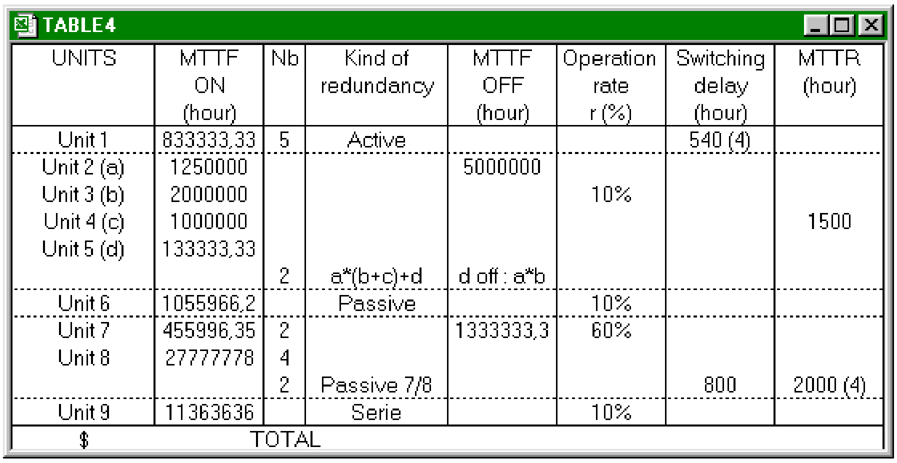
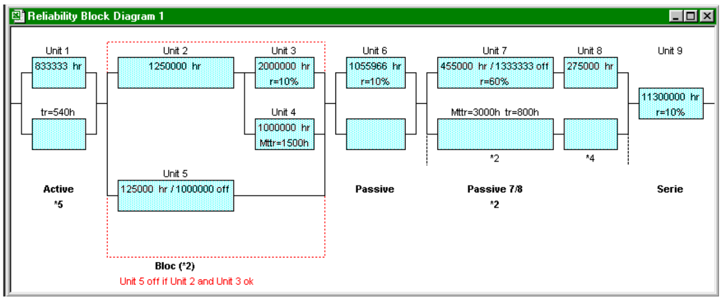
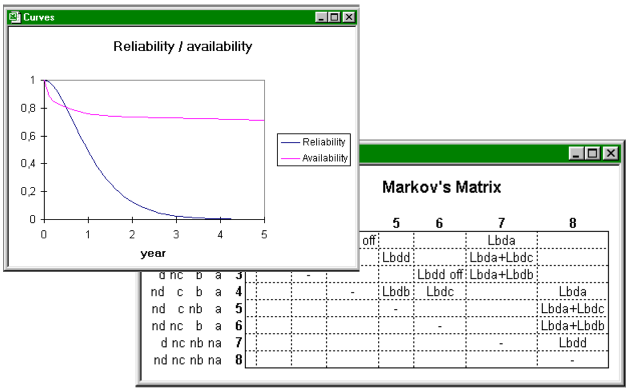


TABLE4

UNITS	MTTF ON (hour)	Nb	Kind of redundancy	MTTF OFF (hour)	Operation rate r (%)	Switching delay (hour)	MTTR (hour)
Unit 1	833333,33	5	Active			540 (4)	
Unit 2 (a)	1250000			5000000			
Unit 3 (b)	2000000				10%		
Unit 4 (c)	1000000						1500
Unit 5 (d)	133333,33						
		2	a*(b+c)+d	d off : a*b			
Unit 6	1055966,2		Passive		10%		
Unit 7	455996,35	2		1333333,3	60%		
Unit 8	27777778	4					
		2	Passive 7/8			800	2000 (4)
Unit 9	11363636		Serie		10%		
$		TOTAL					





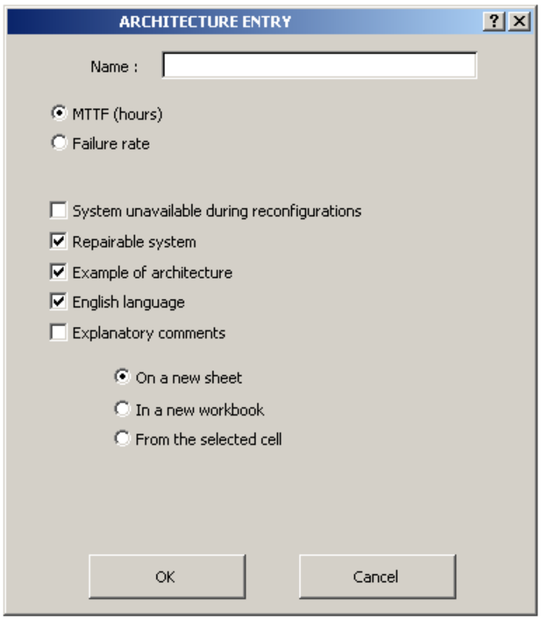
Architectures may consist of classical redundancies of type M among N active or passive, with possible reconfiguration or repair times, or more complex redundancies defined by user expressing their operating mode by logical expressions or using specific modelling.

Used with a posterior version of Excel to version 4, SUPERCAB allows to enter several tables in the same sheet, corresponding for example each one to a subsystem, and to obtain the block-diagram in the form of drawing or image. The various tables can be treated individually or collectively.

The operation of the system from that of subsystems, as well as the operation of the latter from that of the blocks which compose them, can be defined by logical expressions.

3.1 Entering an Architecture

Menu command "Entering architecture" allows getting the following dialog box which helps obtain a blank table, as that shown below, the last two columns of which are optional (used if system is unavailable during certain reconfigurations or if it contains repairable elements).



Name :

● MTTF (hours)
○ Failure rate

☐ System unavailable during reconfigurations
☑ Repairable system
☑ Example of architecture
☑ English language
☐ Explanatory comments

● On a new sheet
○ In a new workbook
○ From the selected cell

UNITS	MTTF	Nb	Kind of	MTTF	Operation	Switching	MTTR
	ON		redundancy	OFF	rate	delay	(hour)
	(hour)			(hour)	r (%)	(hour)	
$			Total :				

Elements may be featured by their MTTF (in hour) or their fault rate (in hour^{-1} or fit = hour^{-1}*10^{-9}). Table and documents made may be requested in the English or French language and one example of architecture preliminarily entered may be displayed for illustration purposes.
Then, the user enters features of system elements either directly in table, or by using the utilities bar "Blocks" the use of which is specified below.

The architecture entry table is filled in as follows :

1 - The name of the element is entered in the first column.

2 - The MTTF or the fault rate of element in active condition is entered in second column.

3 - The number of series elements or sets is entered in the third column.

4 - The type of redundancy to be entered in the fourth column may be "Active m/n", "Passive m/n", "Series", "Logic" (defined by an expression featuring the system availability) or "Complex". This last type does not lead to any processing but allows the user to complete manually the generated calculation tables and Block-Diagrams.
An active or passive redundancy 1/2 is assumed by default if m/n is omitted.
A probability value can be taken by default by entering it in this column.

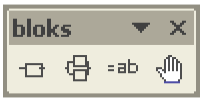
5 - A MTTF OFF equal to MTTF On * 10, or a fault rate OFF equal to the fault rate On / 10, is assumed by default if the latter is omitted in the fifth column.

6 - A 100 % operation rate is assumed by default if the latter is omitted in the sixth column.

7 - A reconfiguration time (in hour) may be entered in the seventh column. Parameter k of an Erlang law used for featuring this time may be related with it in parenthesis (see MARKOV program).

8 - MTTR may be entered in the same way in the eighth column.

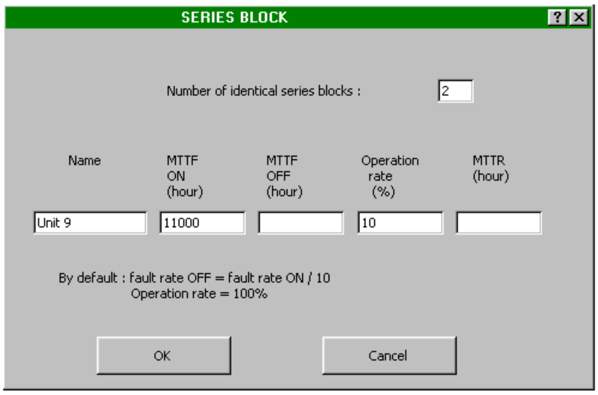
The following utilities bar "Blocks" facilitates such entry. It is displayed or disappears using menu command "Utilities bar".



Meaning of buttons is as follows :

Adding a series block

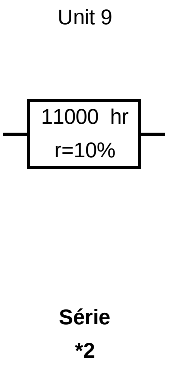
Initiating this button allows to display the following dialog box enabling to insert a series block in the table (or many identical ones) just above the line selected.


SERIES BLOCK

Number of identical series blocks : 2

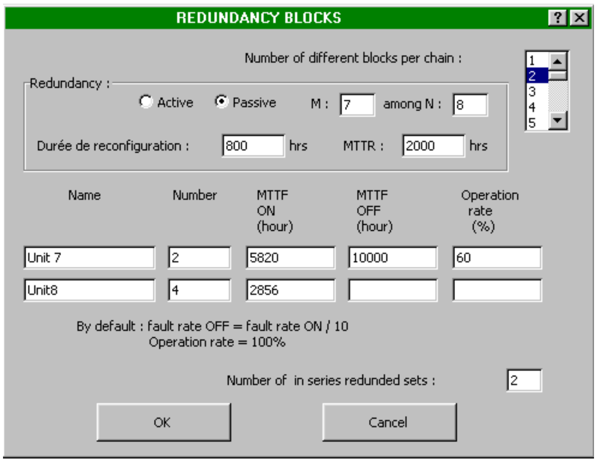
Name	MTTF ON (hour)	MTTF OFF (hour)	Operation rate (%)	MTTR (hour)
Unit 9	11000		10	

By default : fault rate OFF = fault rate ON / 10
Operation rate = 100%



Adding redundancy blocks

Initiating this button allows to display the following dialog box enabling to insert a redundancy block in the table (or many identical ones) just above the line selected.



Number of different blocks per chain : 1 [✓ 2] 3 4 5

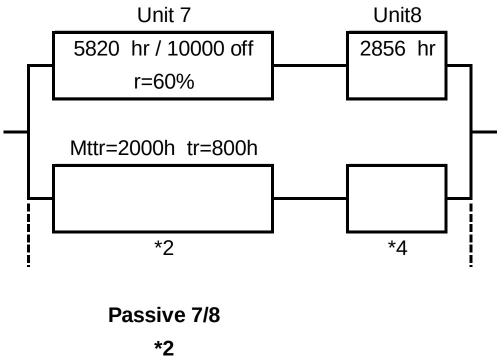
Redundancy : ○ Active ● Passive M : 7 among N : 8

Durée de reconfiguration : 800 hrs MTTR : 2000 hrs

Name	Number	MTTF ON (hour)	MTTF OFF (hour)	Operation rate (%)
Unit 7	2	5820	10000	60
Unit8	4	2856		

By default : fault rate OFF = fault rate ON / 10
Operation rate = 100%

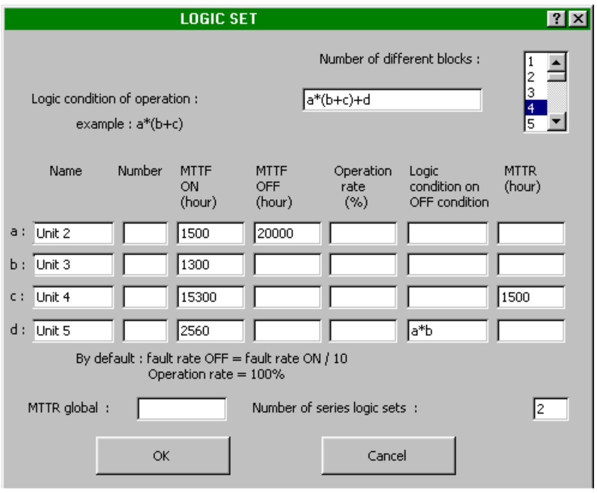
Number of in series redunded sets : 2



The considered redundancy may concern a unique block or a chain of various blocks as in example above (2 * Unit 7 + 4 * Unit 8, the whole in redundancy 7/8).
The type of redundancy of the complete chain is then informed in the entry table at next line according to those of relevant Units.

Adding a logic set

Initiating this button allows to display the following dialog box enabling to insert a logic set in the table (or many identical ones) just above the line selected.



Number of different blocks : 1 2 3 [✓ 4] 5

Logic condition of operation : a*(b+c)+d

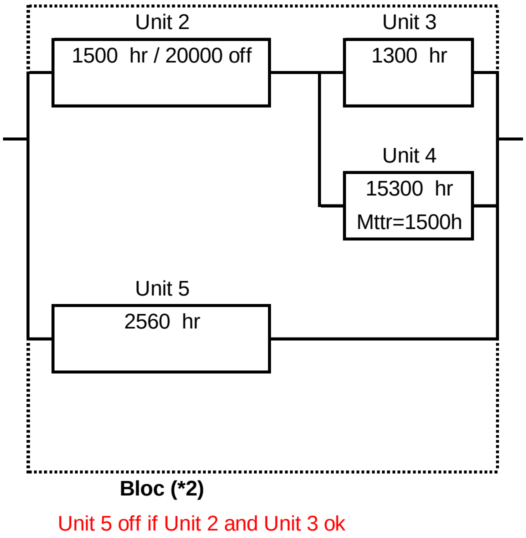
example : a*(b+c)

	Name	Number	MTTF ON (hour)	MTTF OFF (hour)	Operation rate (%)	Logic condition on OFF condition	MTTR (hour)
a :	Unit 2		1500	20000			
b :	Unit 3		1300				
c :	Unit 4		15300				1500
d :	Unit 5		2560			a*b	

By default : fault rate OFF = fault rate ON / 10
Operation rate = 100%

MTTR global :

Number of series logic sets : 2



The condition of proper operation of the set is defined by a logical expression between its various constituents. So, in this example, the expression "a*(b+c)+d" means :

"Unit 2 OK AND (Unit 3 OK OR Unit 4 OK) OR Unit d OK".

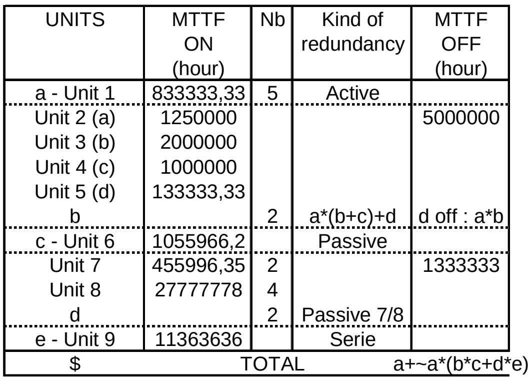
For each block of logic set, a similar expression may be used to define the possible conditions of setting to OFF condition (cold redundancy). So, in this example, Unit 5 (d) is in OFF condition as long as condition "a*b" is verified, that is "Unit 2 OK AND Unit 3 OK".

Adding a complex set

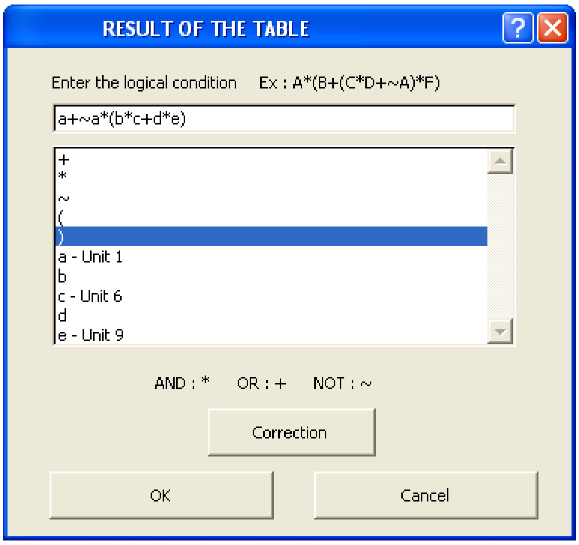
Initiating this button allows to display the following dialog box enabling to insert a complex set in the table (or many identical ones) just above the line selected.
This last type of set does not lead to any specific processing but allows the user to complete manually the generated Block-Diagrams and calculation tables, by using specific modelling. This dialog box is used as previous ones.

Result of the table

The activation of this button allows to display the following dialog box making it possible to insert in the table a logical expression defining the operation of the system from that of the blocks which make it up (by defect the blocks are in series). Each block of the table is then identified by a small letter (automatically inserted) and the expression is built by simple selection in a drop-down list of the logical operators (+: OR, *: AND, ~: NOT), the symbols of bracket and the various blocks. The logical expression is recopied in the table with the activation of button OK.



UNITS	MTTF ON (hour)	Nb	Kind of redundancy	MTTF OFF (hour)
a - Unit 1	833333,33	5	Active	
Unit 2 (a)	1250000			5000000
Unit 3 (b)	2000000			
Unit 4 (c)	1000000			
Unit 5 (d)	133333,33			
b		2	a*(b+c)+d	d off : a*b
c - Unit 6	1055966,2		Passive	
Unit 7	455996,35	2		1333333
Unit 8	27777778	4		
d		2	Passive 7/8	
e - Unit 9	11363636		Serie	
$		TOTAL		a+~a*(b*c+d*e)



RESULT OF THE TABLE

Enter the logical condition Ex : A*(B+(C*D+~A)*F)

a+~a*(b*c+d*e)

Structure on the N+1

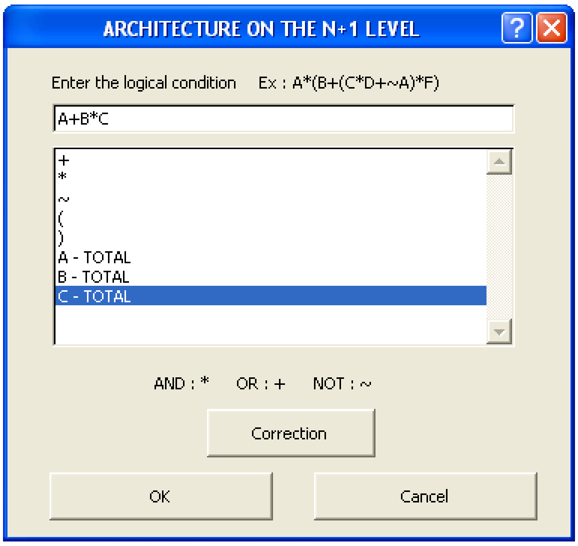
The activation of this button allows to display the following dialog box making it possible to insert on the sheet a logical expression defining the operation of the system, in nominal or degraded modes, from that the different ones subsystems which make it up (each one being defined by its own table).

UNITS	MTTF ON (hour)	Nb	Kind of redundancy
Unit 1	833333,33	5	Active
Unit 2	1055966,2		Passive
$	A - TOTAL		

UNITS	MTTF ON (hour)	Nb	Kind of redundancy
Unit 3	11363636		Serie
Unit 4	455996,35	2	
Unit 5	27777778	4	
		2	Passive 7/8
$	B - TOTAL		

UNITS	MTTF ON (hour)	Nb	Kind of redundancy
Unit 6	11363636		Serie
Unit 7	833333,33	3	Active
$	C - TOTAL		

ARCHITECTURE ON THE N+1 LEVEL	
Logical condition:	A+B*C
$$	


ARCHITECTURE ON THE N+1 LEVEL

Enter the logical condition   Ex : A*(B+(C*D+~A)*F)

A+B*C

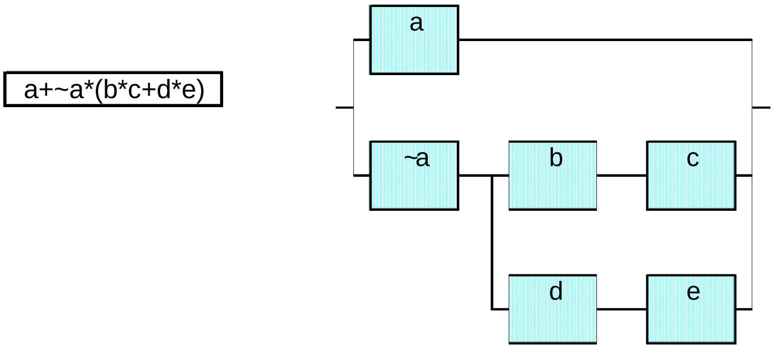
+
*
~
(
)
A - TOTAL
B - TOTAL
C - TOTAL

AND : *   OR : +   NOT : ~

Each subsystem is then identified by a capital letter (automatically inserted) and the expression is built again by simple selection in a drop-down list. The logical expression is recopied in an additional table corresponding to the system (N+1 level) with the activation of button OK.

Drawing of the logic structure

The activation of this button makes it possible to draw a logic structure, in the shape of a block diagram, starting from its expression defined in the selected cell (the symbol ~ corresponds to the negation of the element concerned).



BD *Block Diagram*

The activation of this button has the same effect as the command "Block Diagram" of the menu.

Notes :

- Lines may be inserted or deleted in the table provided the character "$" is maintained at the bottom of the latter (menu "Edit insert" or "Edit cancel").

- Blank lines may be especially used for separating elements or blocks in the table but they should not be inserted within a same block.

3.2 Calculating Reliability / Availability

Menu command "Processing" allows to calculate the reliability and the availability of the architecture preliminarily entered.

The user chooses in a dialog box the time unit to be considered (hour, day or year), and quotes the value of time t for which it desires a processing.

He may possibly request the calculation of the equivalent lambda or the equivalent MTTF for this time value (failure rate or MTTF of an element with same reliability at t than considered architecture).

If he wishes to get a reliability/availability graph, he should enter various time values in discrete form, or define a calculation increment, a starting value (tmin) and a final value (tmax).

The program also helps calculate the asymptotic availability of repairable systems (with infinite t) together with the MTTF, MTBF, MUT and MDT parameters of the different subassemblies.

Acting on the OK button commands the processing. Then, the program adds at table of calculation columns in which redundancy formulas (showed in paragraph 3.4) or probability formulas (below defined) are entered with their various arguments.

If architecture consists of logic sets, relevant Markovian models are automatically generated in specific sheets (see logic analysis program).

Notes :

The number of characters per cell being limited, an error will be indicated if the size of a block's redundancy formula exceeds spreadsheet capacity. So, the user will have to gather manually the features of some block elements to reduce the formula size.

Rates entered by the program in the line defining the type of block redundancy (second and fifth column) correspond to rates at active and passive condition of relevant chains.

For evaluation of architecture, the tool uses formulas of redundancy and Markovian treatments whose limitations are specified in the corresponding chapters.

For the treatment of the logical expressions of level higher than that of the block (table and N+1 level), it uses the following formula of probability (in the shape of macro function of the spreadsheet):

=PROBA("A+~A*B*C", Probability of A, Probability of B, Probability of C)

The first argument is the logical expression between quotation marks and the following corresponds to the value of the various probabilities in the alphabetical order.

By defect, the results displayed at bottom of columns correspond to products of intermediate results shown in relevant columns.

3.3 Generating Block-Diagram Automatically

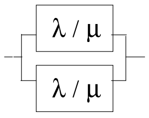
The program automatically generates the Block-Diagram as soon as the user initiates the menu command "Block-Diagram". Such function may be achieved prior or after the reliability / availability calculation.

The Block-Diagram may be in-line plotted or cut depending on printing format (Portrait or Landscape) and be subject to a reduction or enlargement rate.
Used with a posterior version of Excel to version 4, the tool allows obtaining the block-diagram in the form of image. Of variable size according to enlarging or reduction ratio's, the Block Diagram is then drawn in-line and appears on the right of the corresponding table.

4 Markov Matrix

4.1 Reminder on Markov Matrixes

Markovian processes are used in reliability/availability analyses for modelling the behaviour of various systems. The example below illustrates the way followed.



λ : fault rate

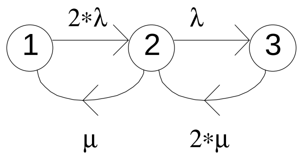
μ : repair rate

System conditions :

1 : No fault

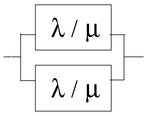
2 : Element loss

3 : System loss



Markov graph

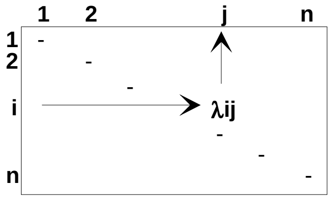
The analyst identifies different system conditions, then defines transition rates between conditions. Certain equivalent conditions may be regrouped as the loss of one or another element in this example.
System may also be represented directly in the form of matrix :



$$\begin{array}{r|ccc|} & 1 & 2 & 3 \\ \hline \text{No fault : 1} & - & 2*\lambda & 0 \\ \text{Loss of one element : 2} & \mu & - & \lambda \\ \text{System loss : 3} & 0 & 2*\mu & - \\ \hline \end{array}$$

Markov matrix

Coefficients λij of matrix correspond to transition rates of conditions i in line towards conditions j in column :



Such matrix representation is more synthetic than the graph and limits risks of errors (forgotten transition).

Transition rate λii located on the main diagonal are not generally informed ($\lambda_{ii} = -\sum_{j=1;\, j\neq i}^{n} t_{ij}$), together with the zero-value rates.

More complex examples of modelling are given in chapter 4.10 of this manual and are quoted for demonstration in online help (menu "Help").

4.2 Entering an Markov Matrix

Menu command "matrix entry" helps obtaining on a calculation sheet, from the selected cell (top left corner), the following format of a blank matrix with dimension n.

MAT :	1	2	3	4	5
1	-				
2		-			
3			-		
4				-	
5					-

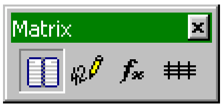
INIT :	1	0	0	0	0

STATE :	1	1	1	1	0

Then, the user enters matrix's transition rates (in hour $^{-1}$) likely to be expressed either in the discrete form, or in the form of formulas incorporating variable names (terms of main diagonal are automatically computed during the processing and empty cells are replaced by 0).

He enters vector INIT, corresponding to probabilities to be in different states at T = 0, and defines a combination of states to be assessed by entering 1 or 0 in corresponding cells of vector STATE.

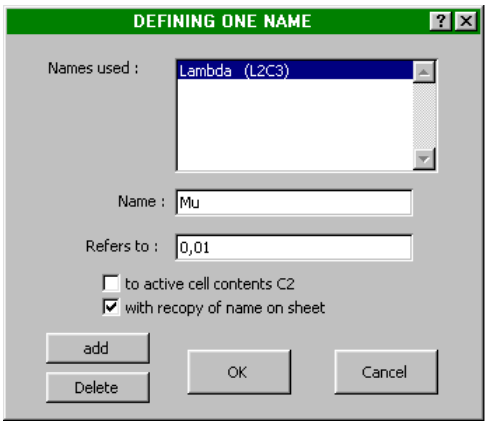
The following utilities bar "Matrixes" allows to facilitate the entry. It is displayed or disappears using menu command "Utilities bar".



Meaning of buttons is as follows :

Defining names

Initiating this button allows to display the following dialog box which helps define variable names in selected sheet.



Names used : Lambda (L2C3)

Name : Mu

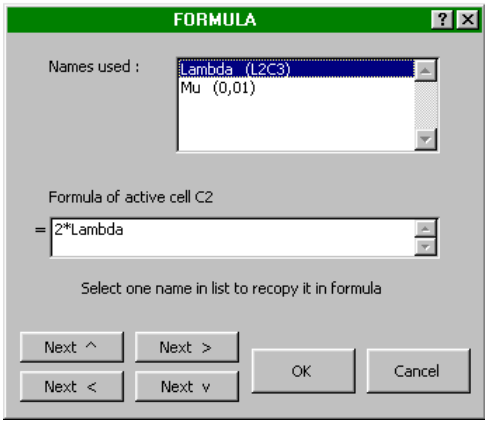
Refers to : 0,01

☐ to active cell contents C2
☑ with recopy of name on sheet

Such names may refer to discrete values or to contents of sheet cells.
The box especially enables to give a name to contents of cell selected and possibly recopy such name on sheet, at the left hand side of such cell.

Using names in a formula

Initiating this button allows to display the following dialog box which helps use names of predefined variables in a formula to enter in selected sheet.



Names used : Lambda (L2C3)
Mu (0,01)

Formula of active cell C2
= 2*Lambda

Select one name in list to recopy it in formula

Selecting one name in the list generates its immediate copy in formula.
Selected cell may be incremented in any direction so as to successively enter the different rates of matrix.

Displaying formulas

Initiating this button allows to display immediately all formulas entered in sheet instead of the value of their result (replacement of "=" by " =" or inversely).

Displaying / deleting square pattern

Initiating this button allows to display or delete sheet square pattern.

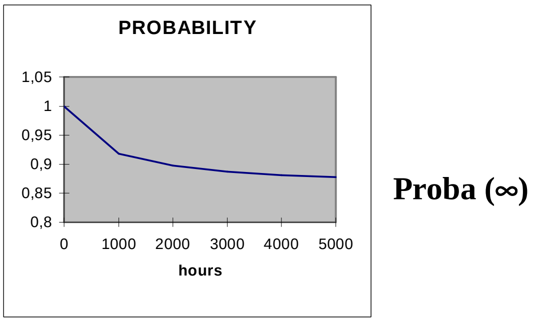
4.3 Processing Markov Matrixes

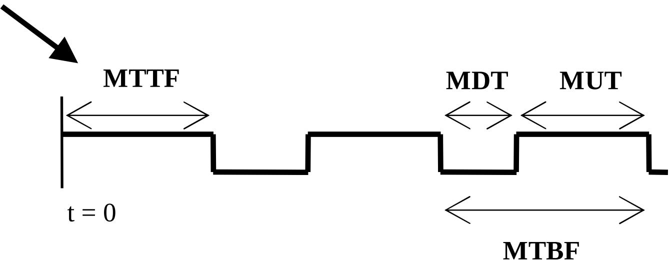
SUPERCAB processes Markov matrixes in transient and steady state and calculates the MTTF, MUT, MDT and MTBF parameters :

MAT :	1	2	3	4
1	-	8,0E-6		6,0E-4
2		-	3,2E-4	
3		5,0E-4	-	
4	1,2E-2			-

INIT :	0,8	0,2	0	0

STATE :	1	1	0	0





MTTF : Mean Time To Failure

MUT : Mean Up Time

MDT : Mean Down Time

MTBF : Mean Time Between Failure

(MTBF = MUT + MDT)

Calculating the probability to be in the various states or a combination of states, at time T, is performed by using menu command "Processing".

Initiating the latter generates the display of a dialog box in which the user enters the value of T, together with a number of intermediate values if he wishes to get the evolution curve between 0 and T.

In addition to Markovian calculation (resolution of the equation of Chapman-Kolmogorov), the treatment can be also carried out in transient state by Monte-Carlo simulation. The user then defined a number of steps corresponding to the number of simulations to carry out between 0 and T.

Pressing button OK commands the processing (see example 1).

If vector STATE is not defined, the program calculates the probability to be in state 1 and displays all probabilities from P1 to Pn if a calculation for intermediate values was requested.

If various matrixes were entered on the same sheet, the user may choose one of them by selecting the relevant cell "MAT :" prior to command the processing.

The probability calculation may also be directly carried out in any cell of calculation sheet by entering the MARKOV function as follows :

"=MARKOV(MAT;INIT;STATE;T)"

This function meets the formalism of spreadsheet macro-functions, a " ; " separating each arguments, and matrix arguments being defined by references of top left hand side and lower right hand side cells of relevant matrixes separated by " : ".
If argument T is not defined, MARKOV function calculates in steady state (at T infinite).
It calculates MTTF, MUT, MDT or MTBF if user enters "MTTF", "MUT", "MDT" or "MTBF" respectively as an argument T, instead of a numerical value.

MARKOV function may be used in matrix form. To do so, select a range of cells and press "block capital" and "CTRL" keys by entering the function.

Used in matrix form, MARKOV function helps displaying time T, probability Pr to be in the combination of states defined by vector STATE, or probabilities P1 to Pn if vector STATE is not defined. It gives such results for intermediate time values between 0 and T if range of selected cells includes various lines.

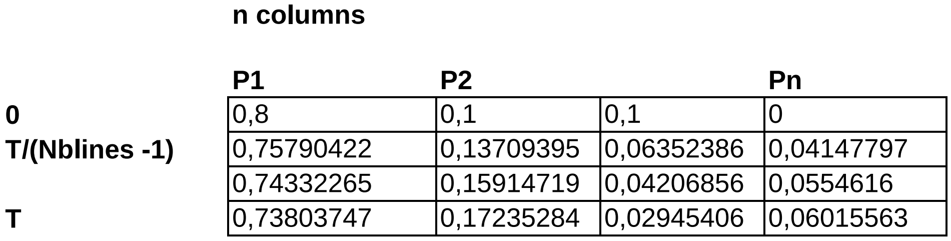
Following examples specify the formalism adopted for such function when used in matrix form.

a) Vector STATE is defined :

	2 columns or more			**1 column**
	Time	**Pr**		**Pr**
0	0	0,9		0,9
T/(Nblines -1)	100	0,82142808		0,82142808
	200	0,78539121		0,78539121
T	300	0,76749154		0,76749154

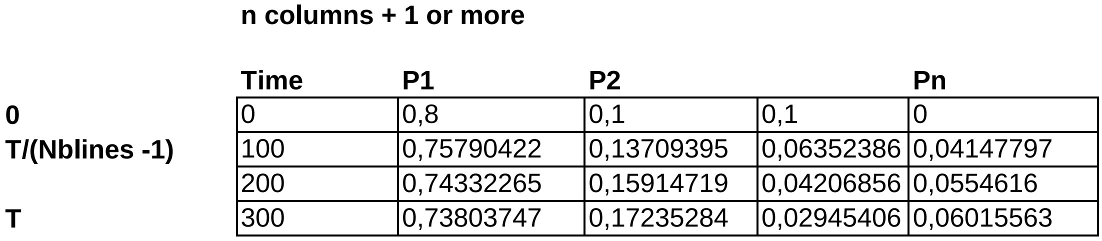
b) Vector STATE is not defined :

	1 column		**2 columns**	
	P1		**P1**	**P2**
0	0,8		0,8	0,1
T/(Nblines -1)	0,75790422		0,75790422	0,13709395
	0,74332265		0,74332265	0,15914719
T	0,73803747		0,73803747	0,17235284



n columns

	P1	P2		Pn
0	0,8	0,1	0,1	0
T/(Nblines -1)	0,75790422	0,13709395	0,06352386	0,04147797
	0,74332265	0,15914719	0,04206856	0,0554616
T	0,73803747	0,17235284	0,02945406	0,06015563



n columns + 1 or more

	Time	P1	P2		Pn
0	0	0,8	0,1	0,1	0
T/(Nblines -1)	100	0,75790422	0,13709395	0,06352386	0,04147797
	200	0,74332265	0,15914719	0,04206856	0,0554616
T	300	0,73803747	0,17235284	0,02945406	0,06015563

The treatment by Monte-Carlo simulation is carried out in a similar way by means of function MARK_SIM defined below:

"=MARK_SIM(MAT;INIT;STATES;T,N)"

With the exception of additional argument N, corresponding to the number of steps of simulation, the formalism of this macro-function is identical to that of the Markov function. Its employment with SIMCAB tool makes it possible to introduce various laws of probability into a Markovian model.

Notes :

- Algorithm of MARKOV function is based on calculation of matrix exponential. It allows to process Markov matrixes of dimension lower than 254 (80 for thr Excel versions before 2007) with a 10^{-15} resolution.

Accuracy of results is guaranteed at 10^{-13} for any value of T lower than the lowest mean transition duration of system (opposite to maximum transition rate).

Beyond such value, the guaranteed accuracy deteriorates proportionally (10^{-9} for a value of T equal to 10000 times the minimum transition duration), and processing duration increases linearly.

- Error value #N/A is returned if sum of probabilities of vector INIT is not equal to 1 or in case of typing error (confusion between dot (.) and decimal point (,) especially in numbers).

- In steady state, the error value #DIV/0! is returned if Markov matrix consists of various absorbing conditions (zero-rate line) or absorbing loops between conditions.

- During a MTTF calculation, the error value #DIV/0! is returned if Markov matrix consists of an absorbing condition (or loop) among the combination of conditions retained.

Example 1 :

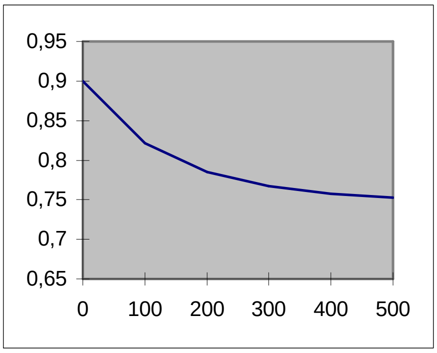
MAT :	1	2	3	4
1	-	8,00E-06		8,50E-04
2		-	3,00E-04	
3		5,00E-03	-	3,00E-07
4	1,00E-02			-

INIT :	0,8	0,1	0,1	0

STATE :	1	0	1	0

Probability : **0,75250553**
à t (hr) : **500**

	T (hr)	PR
	0	0,9
	100	0,82142808
	200	0,78539121
PR = P1 + P3	300	0,76749154
	400	0,75793647
	500	0,75250553
	infinite	0,05858807



MTTF (hr) :	952,399732
MUT (hr) :	206,160984
MDT (hr) :	3312,6611
MTBF (hr) :	3518,82208

MAT :	1	2	3	4
1	-	8,00E-06		8,50E-04
2		-	3,00E-04	
3		5,00E-03	-	3,00E-07
4	1,00E-02			-

INIT :	0,8	0,1	0,1	0

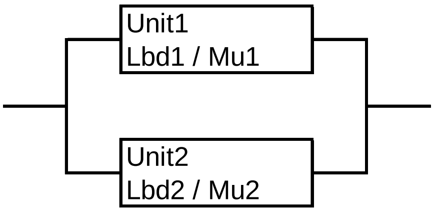
STATE :				

Probability : **0,73481079** **(P1)**
à t (hr) : **500**

T (hr)	P1	P2	P3	P4
0	0,8	0,1	0,1	0
100	0,75790422	0,13709395	0,06352386	0,04147797
200	0,74332265	0,15914719	0,04206856	0,0554616
300	0,73803747	0,17235284	0,02945406	0,06015563
400	0,73589341	0,18035253	0,02204306	0,061711
500	0,73481079	0,18528851	0,01769473	0,06220596
infinite	0,00211764	0,94123024	0,05647043	0,00018169

Example 2 :

Redondancy 1 among 2



Lbd1 =	1E-05	hr -1	
Lbd2 =	0,0005	hr -1	
Mu1 =	0,0208	hr -1	(MTTR : 48 hours)
Mu2 =	0,0139	hr -1	(MTTR : 72 hours)

		1	2	3	4
No failure :	1	-	Lbd1	Lbd2	0
Loss of unit 1 :	2	Mu1	-	0	Lbd2
Loss of unit 2 :	3	Mu2	0	-	Lbd1
System loss :	4	0	Mu2	Mu1	-

MAT :	1	2	3	4
1	-	0,00001	0,0005	
2	0,02083	-		0,0005
3	0,01389		-	0,00001
4		0,01389	0,02083	-

INIT :	1	0	0	0

Availabity

STATE :	1	1	1	0

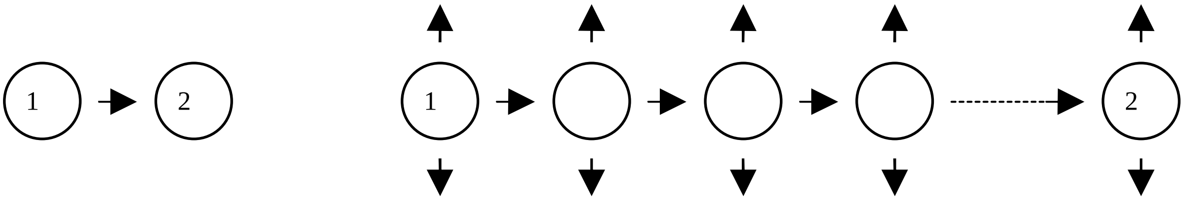
at T = 2000 hours: 0,9999833

P (2000hr) :	0,96479	0,00046	0,03473	1,70E-05

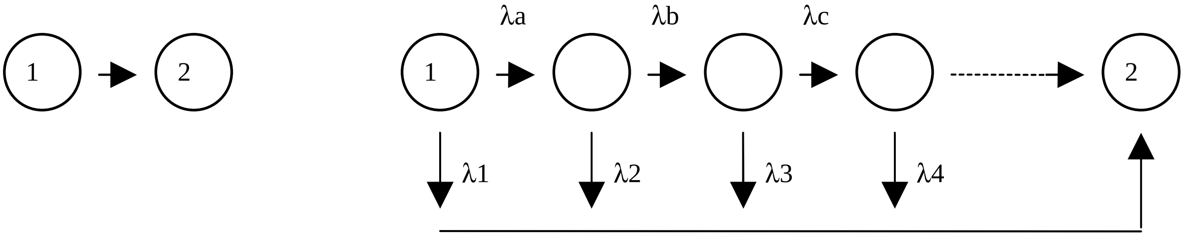
4.4 Fictitious State Method

Only the homogeneous Markovian processes, in which transition rates are constant, may be processed with a matrix calculation (resolution of Chapman-Kolmogorov equation dP/dt = P*M).
Now, if a constant fault rate may have a physical meaning, especially for certain electronic components, it differs for a repair rate.

However, the fictitious state method helps compensate for such limitation. It consists in replacing in a Markovian model any transition between two states by a set of constant-rate transitions between fictitious states.



Any type of transition may be modelled in this way, and the sequence of transitions between fictitious states may be limited to Cox model shown below :

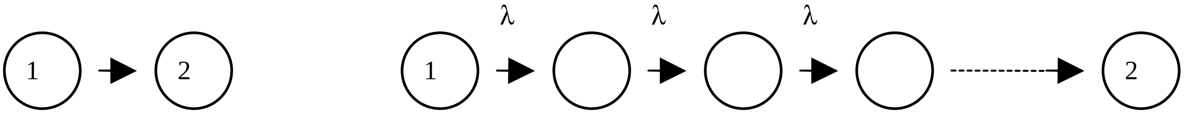


Relevant
Markov matrix :

State 1	-	λa				λ1
		-	λb			λ2
			-	λc		λ3
				-		λ4
State 2						-

If transition rates of direct chain are identical (λa, λb, λc... = λ), model is said generalized Erlang.

If, moreover, the sequence of transitions is limited to direct chain, the resulting law is a simple Erlang law with parameter k, corresponding to the convolution of exponential laws k.



Adding up k-1 fictitious states

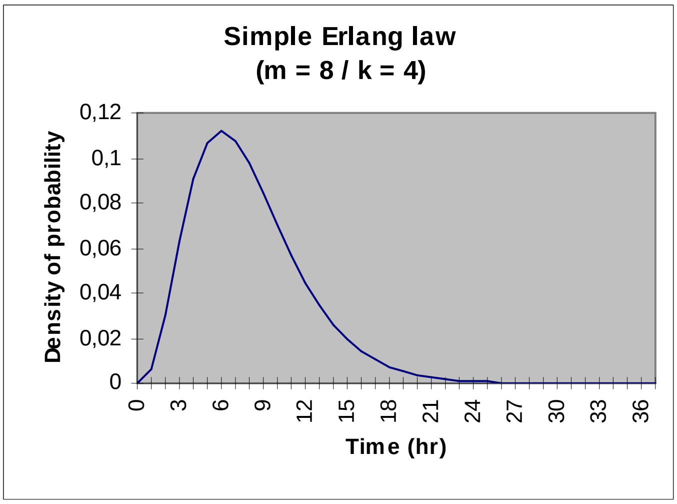
Close to a log-normal law, the latter is often used for modelling repair rates.

The features of a simple Erlang law are as follows :

Mean value : $\frac{k}{\lambda}$

Standard deviation : $\sigma = \frac{m}{\sqrt{k}} = \frac{\sqrt{k}}{\lambda}$

Density : $f(x) = \frac{\lambda^{k} t^{k-1} e^{-\lambda t}}{(k-1)!}$



The fictitious state method is implemented on SUPERCAB using different programs we are about to describe below.

Simple Erlang Law

To enable that a transition in a Markovian model follows a simple Erlang law, the user enters directly in matrix mean value m (in hours) of distribution and parameter value k as follows :

-	a	b
E(m;k)	-	c
d	e	-

→

-	a					b
0	-	λ				c
		-	λ			c
			-			c
					λ	c
λ					-	c
d	e					-

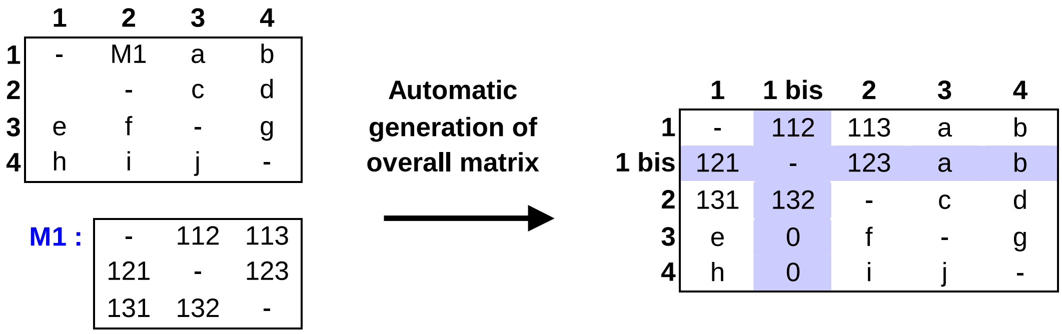
The program will automatically generate the k-1 relevant fictitious states before processing the matrix (see example 3).

Generating Markovian Models

SUPERCAB helps generating Markovian models typical of any transition laws.
So, it is possible to process correctly fault rates relating to mechanical elements (bath-shaped curve), or overcome the combinatory explosion encountered in the study of certain systems by replacing subassembly models by simplified models limited to a few conditions. Such functionality is described in chapter 4.9.

Modelling Entirely A System

In order to facilitate the implementation of the fictitious state method, SUPERCAB helps user achieve the Markov matrix of a complete system from various matrixes, which model its transitions. It generates automatically this matrix from a user-defined simplified matrix in which the complex transition rates are replaced by references to defined Markov matrixes :



	1	2	3	4
1	-	M1	a	b
2		-	c	d
3	e	f	-	g
4	h	i	j	-

M1 :

-	112	113
121	-	123
131	132	-

Automatic generation of overall matrix

	1	1 bis	2	3	4
1	-	112	113	a	b
1 bis	121	-	123	a	b
2	131	132	-	c	d
3	e	0	f	-	g
4	h	0	i	j	-

The simplified matrix may contain up to 10 different references (M1 to M10), and relevant matrixes, preliminarily entered on calculation sheet, are entered as an additional argument of "Markov" function as follows :

"=MARKOV(MAT;INIT;STATE;T;M1;M2;...;M10)"

This functionality enables to achieve automatically the Markov matrix of a complex system from the matrixes of its constituents.
Such matrices may possibly be linked each other and represent an arborescence. (see examples 4 and 5).

Notes Relating To Using Fictitious States :

- If a transition rate tij is replaced by a Markov matrix, transition from state j to state i is also defined by the latter and relevant cell may remain empty.

- Transition rates placed on the same line as a matrix with transition Mi or as a "simple Erlang" type transition, remain operational for the whole fictitious states generated.

- Using various transition matrixes or "simple Erlang" type transitions on a same line should be proscribed as it results in conflicts between transitions.

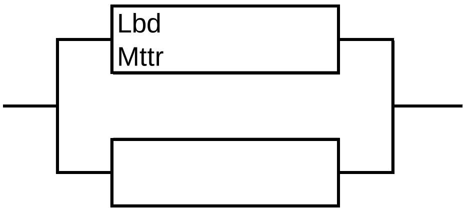
- The Markov program is currently limited to the processing of a 80-states matrix including the possibly generated fictitious states.

- The error value #DIV/0 is returned if mean transition duration m is zero in the expression E(m;k).

- Menu command "Used Matrix " helps displaying, following processing, the matrix generated by program including all fictitious states.

Example 3 :

Simple repairable redundancy



	1	2	3
No fault	-	2*Lbd	0
Loss of one element : 2	Mttr	-	Lbd
Sysem loss : 3	0	0	-

Lbd =	0,00001	hr -1	
Mttr =	48	hours	
k =	4	-->	typical deviation : 24 hours

MAT :	1	2	3
1	-	0,00002	
2	E(48;4)	-	0,00001
3			-

INIT :	0,8	0,2	0

Availability at T= 2000 hours 0,999885

STATE :	1	1	0

Program Generated Matrix :

Mat :

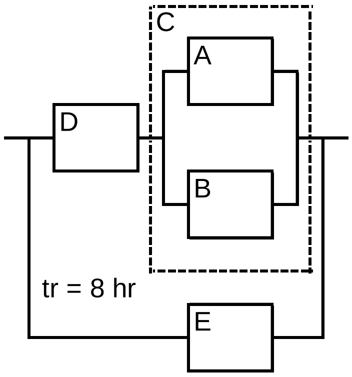
-0,00002	0,00002	0	0	0	0
0	-0,08334	0,083333	0	0	0,00001
0	0	-0,08334	0,083333	0	0,00001
0	0	0	-0,08334	0,083333	0,00001
0,083333	0	0	0	-0,08334	0,00001
0	0	0	0	0	0

Init :	0,8	0,2	0	0	0	0

State :	1	1	1	1	1	0

Example 4 :

Arborescence



Availability at T= 4000 hours :

0,999984971

MAT :	1	2	3	4	5
No fault : 1	-	9E-07	M3	0	0
Loss of D : 2		-	0	E(8;1)	0
Loss of : 3			-	E(8;1)	0
Reconfigured on E : 4				-	M4
Sytem loss : 5					-

INIT :	1	0	0	0	0

STATE :	1	0	0	1	0

M1 (A) :	1	2	3
No loss : 1	-	9E-06	4E-06
Repairable deterioration : 2	E(48;1)	-	8E-06
Definitive loss of A : 3			-

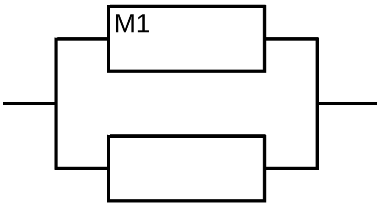
M2 (B) :	1	2	3
No fault : 1	-	8E-06	3E-06
Repairable deterioration : 2	E(24;1)	-	6E-06
Definitive loss of B : 3			-

M3 (C) :	1	2	3
No fault : 1	-	M1	0
Loss of A : 2		-	M2
Loss of C : 4			-

M4 (E) :	1	2	3	4
No fault : 1	-	9E-06	4E-06	1E-06
Repairable deterioration : 2	E(36;1)	-	8E-06	5E-06
Non-repairble deterioration : 3			-	6E-06
Definitive loss of E : 4				-

Example 5 :

Elements undergoing wear
passively redundant



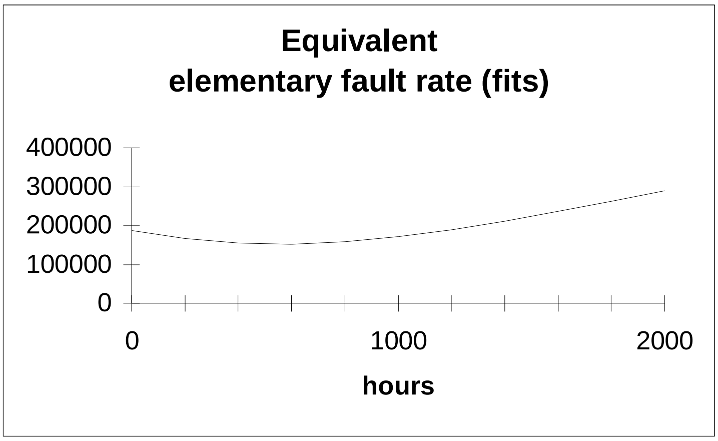
MAT :		1	2	3
No fault :	1	-	M1	0
Loss of active element :	2		-	M1
System loss :	3			-

INIT :	1	0	0

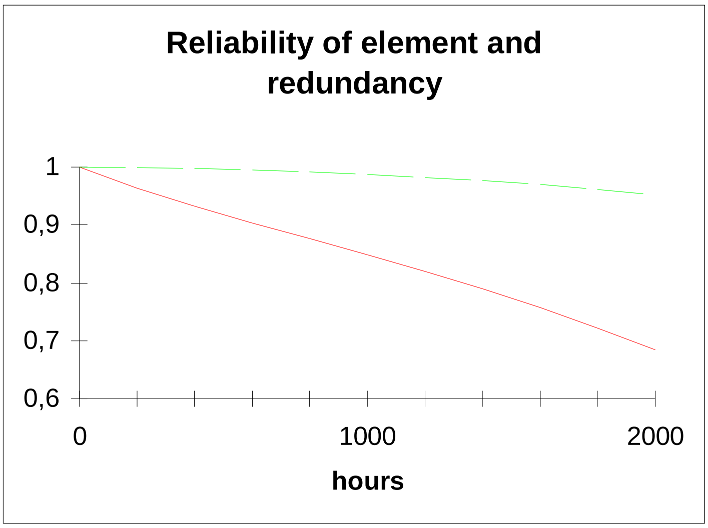
STATE :	1	1	0

M1:		1*	2*	3*	4*	5*	6*
Initial status :	1*	-	0,001				0,0002
	2*		-	0,001			0,00005
	3*			-	0,001		0,00015
	4*				-	0,001	0,00045
	5*					-	0,00135
Final status :	6*						-

Reliability at T= 2000 hours: 0,952194


Equivalent elementary fault rate (fits)

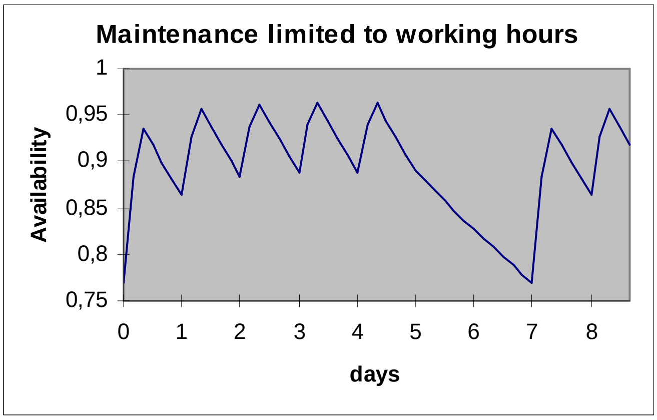
T	Elem,	equiv,	red,
(hr)	rel,	rate	rel,
0	1	186539	1
200	0,96338	165912	0,99929
400	0,93194	154781	0,99745
600	0,90353	152713	0,99476
800	0,87635	158719	0,99134
1000	0,84897	171472	0,98719
1200	0,82035	189533	0,98226
1400	0,78983	211530	0,97642
1600	0,75712	236250	0,96955
1800	0,72217	262686	0,96151
2000	0,68521	290037	0,95219
2200	0,6466		0,9415


Reliability of element and redundancy

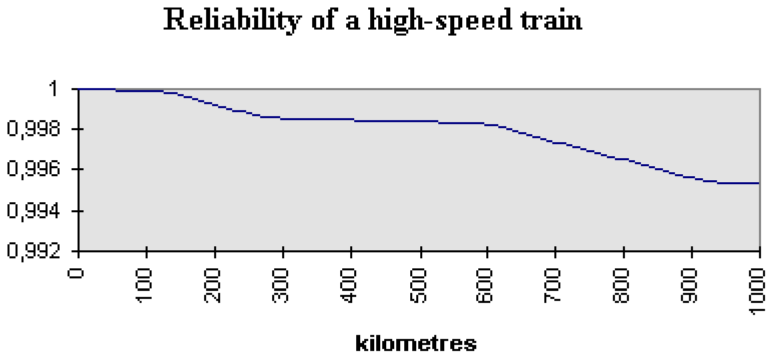
4.5 Evolutionary System

SUPERCAB allows to process Markovian systems the state or the behaviour of which is modified at determined times.

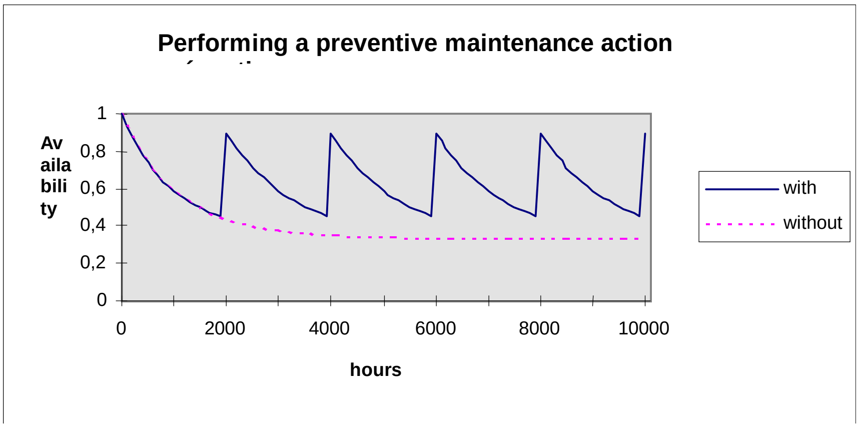
Such developments may correspond to changes in transition rate values or to forcing of states, as shown by examples below :



Repair rates are zero out of working hours.



Failure rates are modified during the journey (high or low speed sections).



System condition is periodically modified by a preventive maintenance action.

Menu command "Evolutionary System " helps processing systems submitted to such determinist developments.

After stating the different numbers of Markov matrixes and forcings required for describing system development, the user defines the latter numbers on a program-formatted calculation sheet (see example 6).

The user specifies the beginning and duration of validity phase of each matrix, which can possibly be periodic.
If validity phases of different matrixes are superimposed, the program warns about it and only considers the phase of matrix with superscript.

Similarly, user specifies the occurrence time of each forcing as well as any possible periodicity.
If forcing sequences take place simultaneously, the latter will be successively performed in their numbering order.

Probabilities (P1 to Pn) following a forcing of states, may be defined either by numerical values, or by expressions comprising such probabilities just prior to forcing (sum is to be equal to 1) :

P1 : P1+alpha*P2 +P3   P2 : (1-alpha)*P2   P3 : 0   P4 : (unchanged if empty box)

(sum of probabilities after forcing is equal to $\sum_{i=1}^{n} Pi$ prior to forcing).

Finally the user defines vectors INIT and STATE, duration of mission considered (Tmax) and the calculation sequence. The latter should be lower than half the duration of matrix validity phases and lower than forcing periods.

Processing will be performed by initiating menu command "Processing".

Notes :

- In addition to the graph of results and relevant numeric values entered in two columns of sheet, detail of calculations is given in the form of one table (in depth configuration for each calculation sequence and in width configuration if events occur between two successive sequences).

- Sum of probabilities of vector INIT should be equal to 1.

- In the event where the processing is requested for a time value corresponding to one forcing, obtaining the calculation result before or after the latter is possible.

Example 6 :

EVOLUTIONARY SYSTEM

Tmax :	800	(hours)
Step :	4	

INIT :	1	0	0
STATE :	1	1	0

Tinit	Period	Duration
0	24	8

MAT1 :	1	2	3
1	-	0,005	
2		-	0,01
3	0,1		-

Tinit	Period	Duration
8	24	16

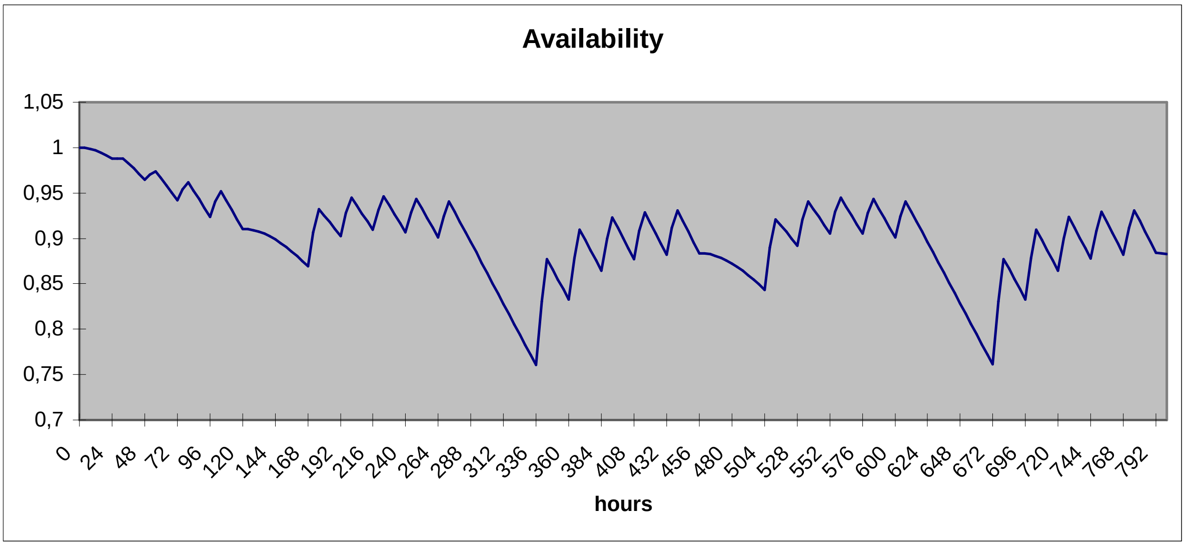
MAT2 :	1	2	3
1	-	0,005	
2		-	0,01
3			-

Tinit	Period	Duration
120	168	48

MAT3 :	1	2	3
1	-	0,005	
2		-	0,01
3			-

Tinit	Period
120	336

Forcing1	P1	P2	P3
	p1+p2	0	

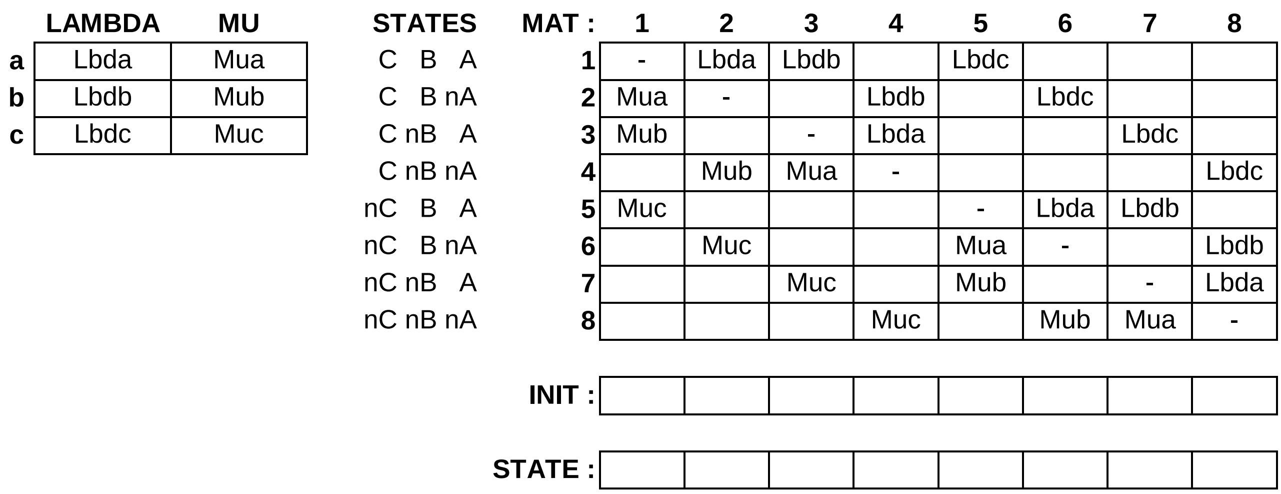


System only maintained on working hours being subject to a specific periodic-maintenance action (every 2 weeks on the weekend)

4.6 Principle of Insertion

Menu command "Matrix entry" helps obtain the matrix of a system of n elements (n ≤ 6), each with two possible states (proper operation or failure), in applying the principle called "Insertion" which is described below.

System elements are designated by characters (from a to f).
Table is updated automatically as soon as user enters elements' fault or repair rates.
Such rates may generate fictitious states if necessary.



	LAMBDA	MU
a	Lbda	Mua
b	Lbdb	Mub
c	Lbdc	Muc

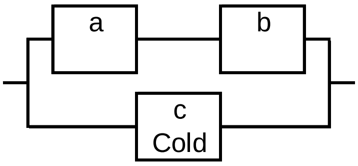
STATES	MAT :	1	2	3	4	5	6	7	8
C B A	**1**	-	Lbda	Lbdb		Lbdc			
C B nA	**2**	Mua	-		Lbdb		Lbdc		
C nB A	**3**	Mub		-	Lbda			Lbdc	
C nB nA	**4**		Mub	Mua	-				Lbdc
nC B A	**5**	Muc				-	Lbda	Lbdb	
nC B nA	**6**		Muc			Mua	-		Lbdb
nC nB A	**7**			Muc		Mub		-	Lbda
nC nB nA	**8**				Muc		Mub	Mua	-

INIT :								
STATE :								

4.7 Logic Analyzer

SUPERCAB generates automatically Markov matrix of a system, from logic expressions featuring its behaviour, and possibly optimizes the dimension of the latter :

a) With No Optimization :



Available states : **(a × b) + c**

Dependence relation : **a × b ⇒ λc* (λoff)**

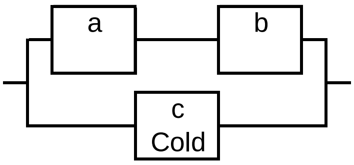
	MAT :	1	2	3	4	5	6	7	8
c b a	1	-	λa	λb		λc*			
c b na	2	μa	-		λb		λc		
c nb a	3	μb		-	λa			λc	
c nb na	4		μb	μa	-				λc
nc b a	5	μc				-	λa	λb	
nc b na	6		μc			μa	-		λb
nc nb a	7			μc		μb		-	λa
nc nb na	8				μc		μb	μa	-

STATE :	1	1	1	1	1	0	0	0

States in which system is available are defined by a logic expression using operators OR (+), AND (×), NOT (n).
Any possible dependence relations, linking certain transition rate values to system conditions (conditional maintenance, cold redundancy...), may be similarly expressed.
Dimension of matrix generated being 2^n, the number of elements is limited to 6 ($2^6 < 80$).

b) With Optimization :

If system elements are not all considered as individually repairable, equivalent states may be regrouped by program for optimizing matrix dimension :



	MAT :	1	2	3	4
c b a	1	-	λc*	λa+λb	
nc b a	2		-		λa+λb
c nb na	3			-	λc
nc nb na	4				-

STATUSES:	1	1	1	0

So as to identify the regrouped states, to each of them is given the name of group state comprising most of failing elements :

c nb na = c b na + c nb a + c nb na   nc nb na = nc b na + nc nb a + nc nb na

Repair rates relating to blocks may be subsequently introduced in reduced matrix (example : repair rates of set ab).

The number of elements is limited to 26 (a to z) and that of regrouped states to 80.
The operator NOT (n) cannot be used in logic expressions when optimization is requested.

Menu command "Logic analyzer" helps obtaining entry formats (see example 7) in which system elements are designated by characters (a to f).

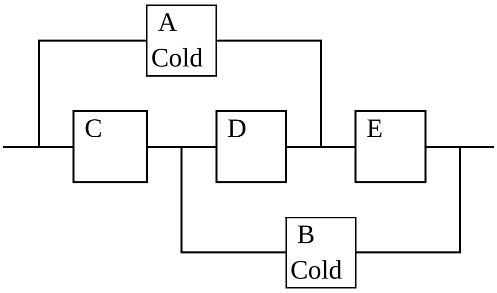
User defines the combination of states being sought, i.e. the one in which system is available, by entering a logic expression between the states of different elements as follows :

n(a*b+n(c+e*nf))

Then, he enters the time, transition rate values (in hr^{-1}), and any possible dependence relations from logic expressions identical to previous one.

Many dependence relations may affect a same rate and user may insert if necessary additional lines in the table.
In case of superimposing states between different logic expressions, the lowest suffix rate is considered.
Markov matrix is generated then processed by the program as soon as user initiates menu command "Processing".

Notes : Example 7 processed with optimization leads to a matrix of dimension 12 instead of 64. Such dimension may still be reduced to 8, as shown below, if logic expressions defining dependence relations are modified for considering the latter only when concerned elements are operational.

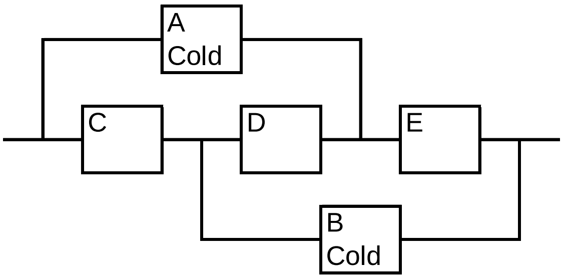


Logic expressions :

Avail :		c*d*e+a*e+c*b
a :	λa	
	λa*	c*d**(a*e)**
b :	λb	
	λb*	e*(d+a)**(c*b)**
c :	λc	
d :	λd	
e :	λe	

MAT :	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
e d c b a **: 1**	-	λa*	λb*	λd		λc	λe	
e d c b na **: 2**		-			λb*		λd+λe	λc
e d c nb a **: 3**			-		λa*	λc+λd		λe
e nd c b a **: 4**				-		λb*+λc	λa+λe	
e d c nb na **: 5**					-			λc+λd+λe
e nd nc nb a **: 6**						-		λa+λe
ne nd c b na **: 7**							-	λb+λc
ne nd nc nb na **: 8**								-
STATE :	1	1	1	1	1	1	1	0

Example 7 :



Loss of C or D -> use of A

Loss of E or (D and A) -> use of B

Probability = 0,9999768
à t (hr) = 5000

		Logic expressions
Available states :		c*d*e+a*e+c*b
Lbda :	7,00E-07	
	7E-08	c*d
	Lbda*2	
Mua :	9,00E-06	
	Mua*1	
	Mua*2	
Lbdb :	8,00E-07	
	8E-08	e*(d+a)
	Lbdb*2	
Mub :	7,00E-06	
	Mub*1	
	Mub*2	
Lbdc :	4,50E-07	
	Lbdc*1	
	Lbdc*2	
Muc :	8,00E-06	
	Muc*1	
	Muc*2	
Lbdd :	7,50E-07	
	Lbdd*1	
	Lbdd*2	
Mud :	6,00E-06	
	Mud*1	
	Mud*2	
Lbde :	9,00E-07	
	Lbde*1	
	Lbde*2	
Mue :	9,50E-06	
	Mue*1	
	Mue*2	

4.8 Personalized Functions

Menu command "Personalized Functions" helps creating and using personalized Markovian functions with the same formalism as redundancy functions m among n shown in previous chapters.

Such functions are stored in files of directory "PERSO" the opening and closing command of which is directly accessible by pressing button "Loading".

Button "Creation" allows to create such function, either in an already existing file, or in a new file.

The user defines in a dialog box the names of function and each of its arguments (maximum 10), the dimension of Markov matrix, then the value and position in the latter of the different transition rates (see example 9).
To do so, he enters relations made from different arguments and may use the $ as variable of 1 to n (n being the matrix dimension).

If different relations define rates in a same position of matrix, only the last relation is considered.

The user may command additional dialog boxes, if necessary, to enter all transition rates.

When matrix is completely defined, initiating the button OK displays a new dialog box used for defining the combination of states being sought (states at 1).

Then, initiating the button OK commands the creation of the personalized function then its storage in the predefined file.

Notes :

- To validate a personalized function, the user may use it on a numerical example and display the relevant Markov matrix using menu command "Matrix used".

- In EXCEL5, personalized function files are displayed in reduced-size windows.

Example 9 :

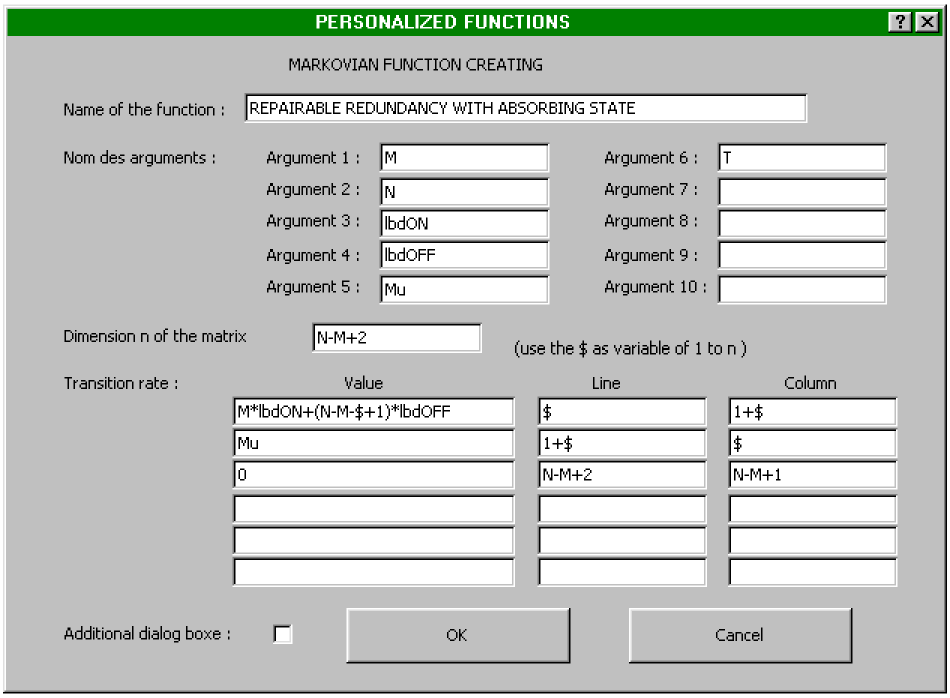
Crating a personalized function

Redundancy m among n repairable with final absorbing state

Markov Matrix:	1	2	3	...	N-M	N-M+1	N-M+2
No fault : 1	-	$M\lambda + (N-M)\ \lambda^*$					
1 element loss : 2	μ	-	$M\lambda + (N-M-1)\lambda^*$				
2 element loss : 3		μ	-				
...							
loss N-M-1 elements : N-M					-	$M\lambda + \lambda^*$	
Loss N-M elements : N-M+1					μ	-	$M\lambda$
System loss : N-M+2						0	-
STATE :	1	1	1		1	1	0

Formula : "= REPAIRABLE _REDUNDANCY_WITH_ABSORBING_STATE (M;N;lbdON;lbdOFF;Mu;T)"

Procedure :



MARKOVIAN FUNCTION CREATING

Name of the function : REPAIRABLE REDUNDANCY WITH ABSORBING STATE

Nom des arguments :

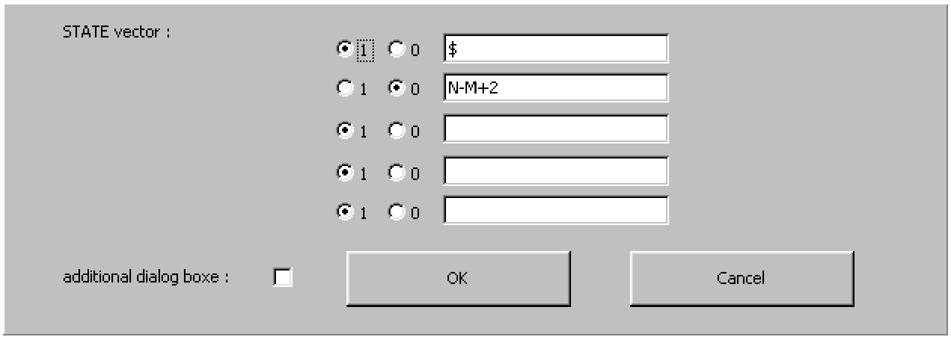
Argument 1 :	M	Argument 6 :	T
Argument 2 :	N	Argument 7 :	
Argument 3 :	lbdON	Argument 8 :	
Argument 4 :	lbdOFF	Argument 9 :	
Argument 5 :	Mu	Argument 10 :	

Dimension n of the matrix N-M+2 (use the $ as variable of 1 to n)

Transition rate :

Value	Line	Column
M*lbdON+(N-M-$+1)*lbdOFF	$	1+$
Mu	1+$	$
0	N-M+2	N-M+1

Additional dialog boxe : □



STATE vector :

● 1	○ 0	$
○ 1	● 0	N-M+2
● 1	○ 0	
● 1	○ 0	
● 1	○ 0	

additional dialog boxe : □

4.9 Approximation of One Law

Menu command "Approximation of one law " helps making a model from a reliability curve, probability density, fault rates or a time sample.

The user chooses one type of model among the exponential, normal, lognormal, Weibull, generalized Erlang or any Markovian laws. He specifies in the last two cases, the dimension of relevant Markov matrix (≤10) and, in the last case, the number of parameters to be considered (≤10).

He also specifies the nature of data considered, the number of curve points or the size of relevant sample, then presses button OK.

Then the program offers a format in which the user enters his data, defines, if necessary, a Markov matrix by using 10 maximum parameters (a to j), and possibly initializes model's parameters.

Then, he initializes menu command "Processing" and defines the number of calculation sequences to be performed.

Then, parameter optimization of model is performed using the method of least error squares in the case of a reliability curve, probability density or fault rate or is based on the method of maximum likelihood in the case of a time sample.

The user may reinitiate the calculation using command "Processing", as many times as desired, after being informed of the evolution of the residual error or likelihood (see example 8).

Notes :

- Normal and lognormal laws can only be used for modelling probability density curves or time samples (due to the absence, for such laws, of an analytical expression of reliability and fault rate).

- In the case of the approximation of a curve by a Markovian model, the processing is slightly quicker if time values start from 0 and follow an arithmetic progression.

- In the case of any Markovian model, seeking an optimum may prove to be very hard as error shows generally many local minima.

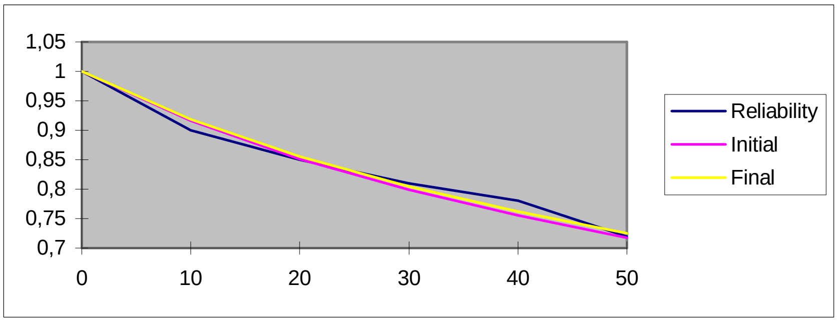
Example 8 :

Time	Reliability
0	1
10	0,9
20	0,85
30	0,81
40	0,78
50	0,72

Markovian model (generalized Erlang)

	1	2	3	4	5
1	-	d			a
2		-	d		b
3			-	d	c
4				-	d
5					-

a0 :	0,009447231	a :	0,009187634
b0 :	0,001432604	b :	0,001253136
c0 :	0,000354237	c :	0,000309561
d0 :	0,019258912	d :	0,019306467
Error :	0,031726406	Error :	0,027472586



	Reliability	Initial	Final
0	1	1	1
10	0,9	0,916639127	0,918959937
20	0,85	0,851477823	0,855651882
30	0,81	0,799183676	0,80483682
40	0,78	0,755466131	0,76228833
50	0,72	0,717064578	0,724791482

Matrix :

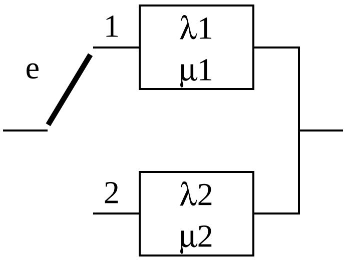
1	2	3	4	5
-	0,000309561			
	-	0,000309561		0,009187634
		-	0,000309561	0,001253136
			-	0,000309561
				-

Approximation of a reliability curve using a generalized Erlang law

4.10 A Few Examples of Markovian Modelling

Example 10 :

Cold redundancy initiated by a switch



	1	2	3	4	5
OK on 1 : 1	-		$e\times\lambda 1$	$\lambda 2^*$	$(1-e)\times\lambda 1$
OK on 2 : 2		-	$\lambda 1^*$	$e\times\lambda 2$	$(1-e)\times\lambda 2$
Loss 1 : 3		$\mu 1$	-		$\lambda 2$
Loss 2 : 4	$\mu 2$			-	$\lambda 1$
Unavailable : 5			$\mu 2$	$\mu 1$	-

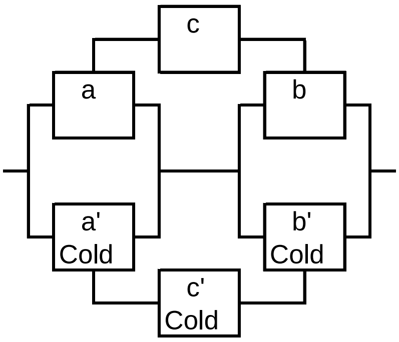
e : efficient when initiated

$\lambda 1^*$ / $\lambda 2^*$: fault rate in off condition of elements 1 / 2

Example 11 :

Cross-strapping with common resources

Example 8 :



a = a'   b = b'   c = c' (ressources)

Lambda(a in hot condition) = a
Lambda(a in cold condition) = a* = a/10
Lambda(b in hot condition) = b
Lambda(b in cold condition) = b* = b/10
Lambda(c in hot condition) = c
Lambda(c in cold condition) = c* = c/10

Simultaneous activation of the minimum elements

	1	2	3	4	5	6
No fault : 1	-	a+a*	b+b*	0	c+c*	0
Loss of a or a' : 2	0	-	0	b	b*+c*	a+c
loss of b or b' : 3	0	0	-	a	a*+c*	b+c
Loss of a and b' or a' and b : 4	0	0	0	-	0	a+b+2c
Loss of all redundancies : 5	0	0	0	0	-	a+b+c
System loss : 6	0	0	0	0	0	0

a =	9,8E-08	hr -1
b =	7,8E-08	hr -1
c =	6,2E-08	hr -1

Reliab(t=10000) = 0,9999977

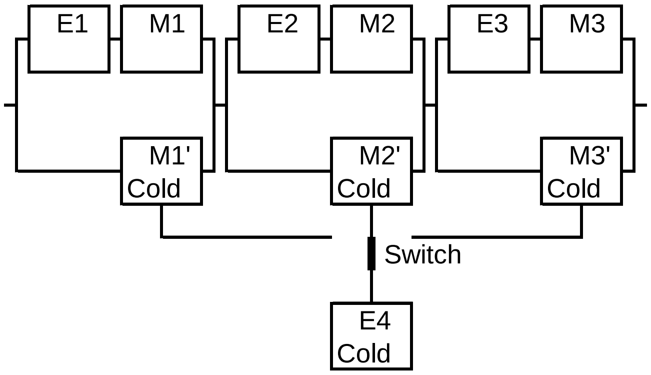
	1	2	3	4	5	
1	-	1,1E-07	8,6E-08	0	6,8E-08	0
2	0	-	0	7,8E-08	1,4E-08	1,6E-07
3	0	0	-	9,8E-08	1,6E-08	1,4E-07
4	0	0	0	-	0	3E-07
5	0	0	0	0	-	2,4E-07
6	0	0	0	0	0	0

INIT :	1	0	0	0	0	0

STATE :	1	1	1	1	1	0

Example 12 :

Commands of redundancy mechanisms 3 among 4



M1 = M2 = M3 = M'1 = M'2 = M'3 = M
(mechanisms)

E1 = E2 = E3 = E4 = E
(control electronics)

Lambda(E in hot condition) = E
Lambda(E in cold condition) = E* = E/10
Lambda(M in hot condition) = M
Lambda(M in cold condition) = M* = M/10

	1	2	3	4	5
No fault : 1	-	3*M*	0	3*(E+M)+E*	0
Loss of 1 passive elements M' : 2	0	-	2*M*	2*(E+M)+E*	E+M
Loss of 2 passive elements M' : 3	0	0	-	E+M+M*+E*	2*(E+M)
Loss of all redundancies : 4	0	0	0	-	3*(E+M)
System loss : 5	0	0	0	0	-

		1	2	3	4	5
E = 1,325E-06 hr -1	1	-	2E-07	0	5,6465E-06	0
M = 5,13E-07 hr -1	2	0	-	1E-07	3,8085E-06	1,838E-06
	3	0	0	-	2,0218E-06	3,676E-06
	4	0	0	0	-	5,514E-06
	5	0	0	0	0	-

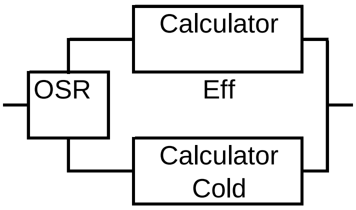
Reliab (t=4000hr) = 0,999752

INIT :	1	0	0	0	0
STATE :	1	1	1	1	0

Example 13 :

Redundancy Calculators

Example 10 :



OSR : Supervision and Reconfiguration Component
(watchdog...)
Eff : Supervision Efficiency (rate)

Lambda(Calculator in hot condition) = Lbd
Lambda(Calculator in cold condition) = Lbdf
Lambda(OSR) = r

	1	3	3
No fault : 1	-	eff*Lbd+r+Lbdf	(1-eff)*Lbd
Loss of a channel : 3	0	-	Lbd
System loss : 3	0	0	-

Lbd =	2,5E-06	hr -1
Lbdf =	2,5E-07	hr -1
r =	4E-07	hr -1
eff =	90%	

	1	3	3
1	-	0,0000029	0,00000025
2	0	-	0,0000025
3	0	0	-

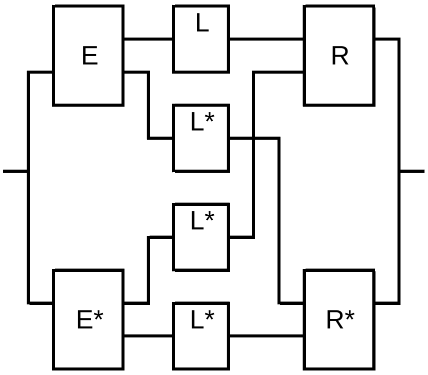
Reliab (t=8000hr) = 0,997796

INIT :	1	0	0
STATE :	1	1	0

Example 14 :

Redunded Communication System

Example 11:



X* : Cold redundancy equipment
Lambda E (Transmitter) = E
Lambda R (Receiver) = R
Lambda L (Linking) = L
Lambda OFF = Lambda* = Lambda/10
Reconfiguration on E* L* R* in case of loss of L

MAT :	**1**	**2**	**3**	**4**	**5**	**6**	**7**
No fault : 1	-	L+3L*	0	E+E*	R+R*	0	0
Loss 1 linking : 2		-	L+2L*	E+E*+L*	R+R*+L*	0	0
Loss 2 linkings : 3			-	0	0	E+E*+R+R*+L+L*	0
Loss 1 transmitter (or linkings) : 4				-	0	R+R*+L+L*	E
Loss 1 receiver (or linkings) : 5					-	E+E*+L+L*	R
All redundancy loss : 6						-	E+L+R
System loss : 7							-

E = 2,40E-7
R = 1,10E-7
C = 8,00E-8

Reliability (t=10 years) :
0,9997083

MAT :	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	-	1,04E-7	0,00E+0	2,64E-7	1,21E-7	0,00E+0	0,00E+0
2		-	9,60E-8	2,72E-7	1,29E-7	0,00E+0	0,00E+0
3			-	0,00E+0	0,00E+0	4,73E-7	0,00E+0
4				-	0,00E+0	2,09E-7	2,40E-7
5					-	3,52E-7	1,10E-7
6						-	4,30E-7
7							-

INIT :	1	0	0	0	0	0	0

STATE :	1	1	1	1	1	1	0

5 Redundancy Formulas

SUPERCAB offers a number of macro-functions to calculate the reliability or the availability of a set of elements in redundancy m among n.
Such functions may be directly selected in the list of functions proposed by spreadsheet by following menu "Selection" then "Insert a function ".

5.1 Active Redundancy

n elements are simultaneously active but only m elements are necessary to ensure mission.
Reliability is obtained by the following formula by replacing arguments in the indicated order :

"= ACTIVE_REDONDANCY(M;N;lambda;T)"

lambda : fault rate of element in hour^{-1}
T : mission duration in hour

"= ACTIVE_REDONDANCY (5;7;0,00001;5000)" = 0,9965

Note : Reliability of such redundancy may be expressed as follows :

$$R = \sum_{i=0}^{N-M} C_N^i (1-e^{-\lambda t})^i (e^{-\lambda t})^{N-i}$$

5.2 Passive Redundancy

Only m elements required for ensuring the mission are simultaneously active.
The reliability is obtained by following formula :

"= PASSIVE_REDUNDANCY(M;N;lambda_on;lambda_off;T)"

lambda_on : fault rate of element in active condition
lambda_off : fault rate of element in passive condition

Note : Reliability of such redundancy may be expressed as follows :

$$R = e^{-M\lambda t}\left[1 + \sum_{i=1}^{N-M} \frac{(1-e^{-\lambda^* t})^i}{i!} \prod_{j=0}^{i-1} (j + M\frac{\lambda}{\lambda^*})\right] \quad \text{with } \lambda^* = \text{lambda\_off}$$

5.3 Redundancy with Reconfiguration Duration

In case of failure of an active element, the mission is stopped for all reconfiguration duration. Availability is obtained by the following formula :

"= REDONDANCY_WITH_DURATION(M;N;lambda_on;lambda_off;T;Treconf;k)"

Treconf : average reconfiguration duration in hour
k : parameter of an Erlang law
(random variable of mean value Treconf and with typical deviation Treconf/k)
k = 1 by default (exponential law)

Notes : Such function is processed by performing the following Markov matrix with dimension 2*(N-M)+2.
Therefore it is limited at values of M and N defined by relation N-M ≤ 39 (if k = 1).

	1	2	3	4	5	...		2(N-M)+1	2(N-M)+2
No fault : 1	-	$M\lambda$	$(N-M)\ \lambda^*$						
Reconfiguration : 2		-	tr	$(M-1)\ \lambda +$ $(N-M)\ \lambda^*$					
1 element loss : 3			-	$M\lambda$	$(N-M-1)\ \lambda^*$				
Reconfiguration : 4				-	tr				
2-element loss : 5					-				
...									
Reconfiguration : 2(N-M)							-	tr	$(M-1)\ \lambda + \lambda^*$
N-M element loss : 2(N-M)+1								-	$M\lambda$
System loss : 2(N-M)+2									-

λ: lambda ON λ^* : lambda Off tr : 1/Treconf if k=1

5.4 Redundancy of Repairable Elements

Availability is obtained by following formula :

"= REPAIRABLE_REDUNDANCY(M;N;lambda_on;lambda_off;T;MDT;k)"

MDT : mean repair time in hour

Notes : Such function considers only one single repair officer and is processed by performing following Markov matrix with dimension N-M+2 .
It is therefore limited to values of M and N defined by relation N-M ≤ 78 (if k = 1).

	1	2	3	...	N-M+1	N-M+2
No fault : 1	-	$M\lambda + (N-M) \lambda^*$				
1 element loss : 2	μ	-	$M\lambda + (N-M-1)\lambda^*$			
2-element loss : 3		μ	-			
...						
N-M element loss : N-M+1					-	$M\lambda$
System loss : N-M+2					μ	-

μ : 1/MDT if k=1

5.5 Repairable Redundancy with Reconfiguration Duration

In case of failure of an active element, the mission is stopped for all reconfiguration duration. Elements are repairable.

Availability is obtained by the following formula :

"=M_AMONG_N_REDUNDANCY(M;N;Lambda_on;Lambda_off;T;Treconf;kr;MDT;km)"

Treconf : reconfiguration meantime in hour
kr : parameter of the relevant Erlang law

MDT : mean repair time in hour
km : parameter of relevant Erlang law

Notes : Such function is processed by performing the following Markov matrix.
It is limited to values of M and N defined by relation $N-M \leq 39$ (if kr and km = 1).

	1	2	3	4	5	...		2(N-M)+1	2(N-M)+2
No fault : 1	-	$M\lambda$	$(N-M) \lambda^*$						
Reconfiguration : 2	μ	-	Tr	$(M-1) \lambda +$ $(N-M) \lambda^*$					
1 element loss : 3	μ		-	$M\lambda$	$(N-M-1) \lambda^*$				
Reconfiguration : 4			μ	-	tr				
2-element loss : 5			μ		-				
...									
Reconfiguration : 2(N-M)							-	tr	$(M-1) \lambda + \lambda^*$
N-M element loss : 2(N-M)+1								-	$M\lambda$
System loss : 2(N-M)+2								μ	-

5.6 Global Redundancy

This more generic model can take into account ON and OFF failure rates (λ and λ_{OFF}), probability of failure before using up (γ), average reconfiguration duration in which the mission is stopped (MDT_s) and average non-operational duration due to failure (MDT_l).

Both durations can be modelled by exponential (rates μ_i) or Erlang law (with k fictitious states). Several politics of maintenance can be considered with 1 or n repair officers or with duration of reparation independent of the number of units in failure.

GLOBAL_REDONDANCY (M; N; λ;λ_{OFF}; γ; T; MDT_l; kl; Type_l; MDT_s; ks; Type_s; Re; Na; Nb)

This parametrical function assesses the availability (at t or $\propto$) or the reliability, considering the loss of system as an absorbing state.

In case of the mission is not stopped during reconfiguration duration, the markovian matrix with dimension N-M+2 as shown in the following figure is used.

	1	2	3	4	...	N-M+1	N-M+2
No failure : 1	-	$M\lambda(1-\gamma) + (N-M)\ \lambda^*$	$M\lambda\gamma\ (1-\gamma)$	$M\lambda\gamma^2\ (1-\gamma)$		$M\lambda\gamma^{N-M-1}\ (1-\gamma)$	$M\lambda\gamma^{N-M}$
Loss of 1 unit : 2	μl_1	-	$M\lambda(1-\gamma) + (N-M-1)\lambda^*$	$M\lambda\gamma\ (1-\gamma)$		$M\lambda\gamma^{N-M-2}\ (1-\gamma)$	$M\lambda\gamma^{N-M-1}$
Loss of 2 unit : 3	$\mu l'$	μl_2	-	$M\lambda(1-\gamma) + (N-M-2)\lambda^*$		$M\lambda\gamma^{N-M-3}\ (1-\gamma)$	$M\lambda\gamma^{N-M-2}$
Loss of 3 unit : 4	$\mu l'$		μl_3	-		$M\lambda\gamma^{N-M-4}\ (1-\gamma)$	$M\lambda\gamma^{N-M-3}$
...							
Loss of N-M unit : N-M+1	$\mu l'$					-	$M\lambda$
System loss: N-M+2	$\mu l''$					μl_{N-M+1}	-

M among N redundancy (system available during reconfigurations)

In case of the mission is stopped during reconfiguration duration, the markovian matrix shown in the following figure, with dimension 2(N-M)+2, is used.

	1	2	3	4	5	6	...	2(N-M)+2
No failure : 1	-	$M\lambda$	$(N-M)\ \lambda^*$					
Reconfiguration : 2	$\mu 2$	-	$tr(1-\gamma)$	$tr\gamma +(M-1)\ \lambda +(N-M)\ \lambda^*$				
Loss of 1 unit : 3	$\mu 1$		-	$M\lambda$	$(N-M-1)\ \lambda^*$			
Reconfiguration : 4			$\mu 2$	-	$tr(1-\gamma)$	$tr\gamma +(M-1)\ \lambda +(N-M-1)\ \lambda^*$		
Loss of 2 unit : 5			$\mu 1$		-	$M\lambda$		
Reconfiguration : 6						-		
...								
Reconfiguration : 2(N-M)								$tr\gamma +(M-1)\ \lambda + \lambda^*$
Loss of N-M unit : 2(N-M)+1								$M\lambda$
System loss : 2(N-M)+2								-

M among N redundancy (system unavailable during reconfigurations)

Type_l = 1 : 1 repair officer ($\mu l_i = \mu l$, $\mu l' = \mu l'' = 0$)

Type_l = 2 : n repair officers ($\mu l_i = i * \mu l$, $\mu l' = \mu l'' = 0$)

Type_l = 3: duration of reparation independent of the number of units in failure

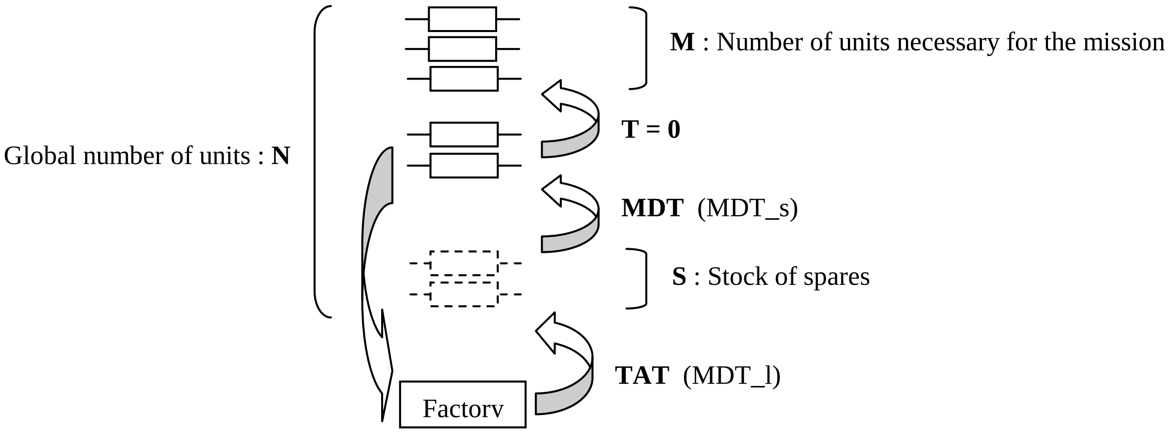
($\mu l_i = 0$, $\mu l' = \mu$ l, $\mu l'' = \mu l$ if R = False)

Type_s = true: repairing starts at the beginning of reconfiguration duration

Re = true (Reliability) $\Rightarrow \mu l_{N-M+1}$ and $\mu l'' = 0$.

The parameter Na allows to assess the probability of the state of row Na (in case of the mission is stopped during reconfiguration or not)

The Nb parameter allows per example to calculate the reliability or the availability of a set of units in redundancy M among N with a stock S of spares as shown in the following figure.


M : Number of units necessary for the mission
T = 0
Global number of units : N
MDT (MDT_s)
S : Stock of spares
TAT (MDT_l)
Factorv

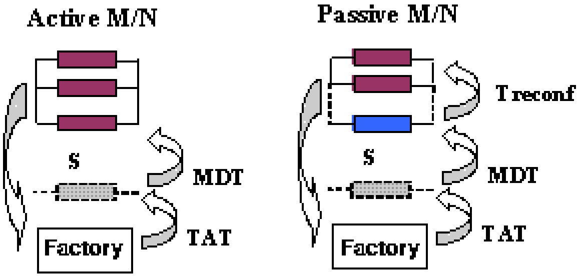
M among N with a stock S of spares

The system is available during the Nb = N-M-S first reconfiguration duration but unavailable during the following ones.

Example 15 of the following page uses this formula of redundancy within the framework of an optimization of batches of spare. The coupling of software SUPERCAB and GENCAB indeed makes it possible to automate this type of treatment of minimizing, for example, the total cost of a system having to respect a certain objective of availability.

5.6 Active and passive redundancy with stock of spares

One second generic formula of redundancy is proposed to treat the active or passive redundancies of type M among N with spares stock of S dimension.
This one allows evaluating the reliability or the availability of such a redundancy by considering the duration of reconfiguration (Treconf), the duration of repair per standard exchange (MDT) and the Turn around Time (TAT), corresponding to the duration of spare replacement.



Redondancy(M;N;λON;λOFF;T;Treconf;MDT;Nb_operators;S;TAT;Nb_repairers;Active/passive; Reliability/Availability)

T, Treconf, MDT, TAT, 1/λ, 1/λOFF

- Same unit of time

λOFF

- 0 per defect

Nb_operators

- 1 : 1 operator carries out the standard exchanges (in series)
- 2 : N operators carry out the standard exchanges (in parallel)

Nb_ repairers

- 1 : 1 repairer carries out repairs (in series)
- 2 : N repairers carry out repairs (in parallel)

Active/passive

- 1 : Active
- 2 : Passive

Reliability/Availability

- 1 : Final loss of the system so less M active elements
- 2 ou 3 : Temporary unavailability so less M active elements
- 3 : Final loss of the system so less M active elements and absence of spare

The Markovian models used by this formula are provided in the following pages.

Example 15 :

UNITS	MTTF ON (hour)	Nb	Kind of redundancy	Stock	Unit Cost (Euros)	MDT (hour)	TAT (hour)	Operational availability	Cost (Euros)
Az/el Engine	100000	2	Serie	1	4500	28	2400	0,9972	13500
Coders	100000	2	Serie	2	1500	28	2400	0,9993	6000
Butted	1000000	4	Serie	1	500	28	2400	0,9998	2500
ACU	40000	1	Serie	1	6500	26	1368	0,9981	13000
Piloting Calculator	31150	1	Passive 1/2	0	4000	25	1368	0,9980	8000
LNA chanel PCG or PCD	100000	2	Serie	1	3500	27	3120	0,9957	10500
Down converterchanel PCG or PCD	60000	2	Serie	1	2500	26	2256	0,9937	7500
Up converter	100000	1	Serie	1	2000	26	2376	0,9991	4000
HPA	20000	1	Serie	1	5000	27	1368	0,9940	10000
PM -TC Modulator	8000	1	Serie	2	3500	26	1200	0,9932	10500
PM/PSK-TC Modulator	50000	1	Serie	1	3500	26	1200	0,9989	7000
SEBB Calculator	16013	1	Passive 1/2	1	7000	27	1368	0,9991	21000
DRR70 Receiver	33300	1	Serie	1	8000	26	1200	0,9979	16000
Time link 1	35000	1	Serie	1	4500	26	1248	0,9979	9000
Router	50000	1	Serie	1	1500	26	1080	0,9990	3000
Electric box	50000	1	Serie	1	1000	26	432	0,9994	2000
$							GLOBAL STATION :	**0,9609**	**143500**

Availability > Objective: **0,96**

Optimization of a stock of spares by using GENCAB tool

Criterion of cost with respect of a constraint of availability

Active redundancy

			1	2	3	-	N-M	N-M+1	N-M+2	N-M+3	N-M+4	N-M+5	-		(N-M+2)*2-1	(N-M+2)*2	(N-M+2)*3	(N-M+2)*4	-		(N-M+2)*(S+1)-1	(N-M+2)*(S+1)
OK		1	-	$N*\lambda_{ON}$						$S*\lambda_{OFF}$												
Loss 1 unit		2		-	$(N-1)*\lambda_{ON}$					1/MDT	$S*\lambda_{OFF}$											
Loss 2 units		3			-						1/MDT (4)	$S*\lambda_{OFF}$										
-	-	-				-																
Loss N-M-1 units		N-M					-	$(M+1)*\lambda_{ON}$						$S*\lambda_{OFF}$								
Loss N-M units		N-M+1						-	$M*\lambda_{ON}$ (2)					1/MDT (4)	$S*\lambda_{OFF}$							$M*\lambda_{ON}$ (1)
Indisponible		N-M+2							-						1/MDT (4)	$S*\lambda_{OFF}$						
	-1 spare	N-M+3	1/TAT							-	$N*\lambda_{ON}$						$(S-1)*\lambda_{OFF}$					
Loss 1 unit	-1 spare	N-M+4		1/TAT							-	$(N-1)*\lambda_{ON}$					1/MDT	$(S-1)*\lambda_{OFF}$				
Loss 2 units	-1 spare	N-M+5			1/TAT							-						1/MDT (4)				
-	-	-											-									
Loss N-M-1 units	-1 spare	(N-M+2)*2-2					1/TAT							-	$(M+1)*\lambda_{ON}$							
Loss N-M units	-1 spare	(N-M+2)*2-1						1/TAT							-	$M*\lambda_{ON}$ (2)						$M*\lambda_{ON}$ (1)
Indisponible	-1 spare	(N-M+2)*2							1/TAT							-						
	-2 spares	(N-M+2)*3								1/TAT (5)							-	$N*\lambda_{ON}$				
Loss 1 unit	-2 spares	(N-M+2)*4									1/TAT (5)							-				
-	-	-																	-			
Loss N-M-1 units	-S spares	(N-M+2)*(S+1)-2																		-	$(M+1)*\lambda_{ON}$	
Loss N-M units	-S spares	(N-M+2)*(S+1)-1																			-	$M*\lambda_{ON}$
Loss Système		(N-M+2)*(S+1)																			1/(TAT+MDT) (3)	-

M :	2
N :	4
S :	2
(N-M+2)*(S+1) :	12

(1) Reliability/Availability = 1 : Final Loss of the system so less M units active

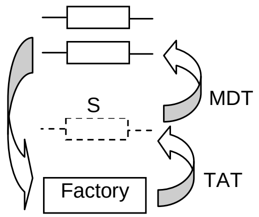
(2) Reliability/Availability = 2 or 3 : Temporary unavailability so less M units active

(3) Reliability/Availability = 2 : Unavailability of MDT+TAT/i duration if i repairers
Reliability/Availability = 1 ou 3 : Absorbing state

(4) i/MDT I operators, with I lower or equal to the number of spares available

(5) j/TAT if j repairers

Active M/N

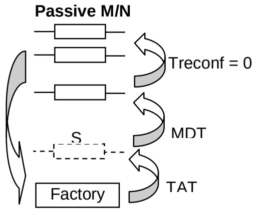


Redondancy(M;N;λ_{ON};λ_{OFF};T;Treconf;MDT;Nb_operators;S;TAT;Nb_repairers;Active/passive;Reliability/Availability)

Passive Redundancy - Treconf = 0			1	2	3	-	N-M	N-M+1	N-M+2	N-M+3	N-M+4	N-M+5	-	(N-M+2)*2-2	(N-M+2)*2-1	(N-M+2)*2	(N-M+2)*3	(N-M+2)*4	-		(N-M+2)*(S+1)-1	(N-M+2)*(S+1)
OK		1	-	$M*\lambda_{ON}+(N-M)*\lambda_{OFF}$						$S*\lambda_{OFF}$												
Loss 1 unit		2		-	$M*\lambda_{ON}+(N-M-1)*\lambda_{OFF}$					1/MDT	$S*\lambda_{OFF}$											
Loss 2 units		3			-						1/MDT (4)	$S*\lambda_{OFF}$										
-	-	-				-																
Loss N-M-1 units		N-M					-	$M*\lambda_{ON}+\lambda_{OFF}$						$S*\lambda_{OFF}$								
Loss N-M units		N-M+1						-	$M*\lambda_{ON}$ (2)					1/MDT (4)	$S*\lambda_{OFF}$							$M*\lambda_{ON}$ (1)
Indisponible		N-M+2							-						1/MDT (4)	$S*\lambda_{OFF}$						
	-1 spare	N-M+3	1/TAT							-	$M*\lambda_{ON}+(N-M)*\lambda_{OFF}$						$(S-1)*\lambda_{OFF}$					
Loss 1 unit	-1 spare	N-M+4		1/TAT							-	$M*\lambda_{ON}+(N-M-1)*\lambda_{OFF}$					1/MDT	$(S-1)*\lambda_{OFF}$				
Loss 2 units	-1 spare	N-M+5			1/TAT							-						1/MDT (4)				
-	-	-											-									
Loss N-M-1 units	-1 spare	(N-M+2)*2-2					1/TAT							-	$M*\lambda_{ON}+\lambda_{OFF}$							
Loss N-M units	-1 spare	(N-M+2)*2-1						1/TAT							-	$M*\lambda_{ON}$ (2)						$M*\lambda_{ON}$ (1)
Indisponible	-1 spare	(N-M+2)*2							1/TAT							-						
	-2 spares	(N-M+2)*3								1/TAT (5)							-	$M*\lambda_{ON}+(N-M)*\lambda_{OFF}$				
Loss 1 unit	-2 spares	(N-M+2)*4									1/TAT (5)							-				
-	-	-																	-			
Loss N-M-1 units	-S spares	(N-M+2)*(S+1)-2																		-	$M*\lambda_{ON}+\lambda_{OFF}$	
Loss N-M units	-S spares	(N-M+2)*(S+1)-1																			-	$M*\lambda_{ON}$
Loss Système		(N-M+2)*(S+1)																			1/TAT (3)	-

M : 2
N : 4
S : 2
(N-M+2)*(S+1) : 12

(1) Reliability/Availability = 1 : Final Loss of the system so less M units active
(2) Reliability/Availability = 2 or 3 : Temporary unavailability so less M units active
(3) Reliability/Availability = 2 : Unavailability of TAT/(S+1) duration if S repairers
Reliability/Availability = 1 ou 3 : Absorbing state
(4) i/MDT I operators, with I lower or equal to the number of spares available
(5) j/TAT if j repairers



Redondancy(M;N;λ_{ON};λ_{OFF};T;Treconf;MDT;Nb_operators;S;TAT;Nb_repairers;Active/passive;Reliability/Availability)

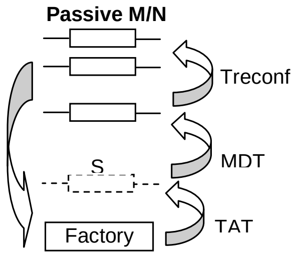
Passive redundancy - Treconf <> 0

			1	2	3	4	5	-		2*(N-M)-1	2*(N-M)	2*(N-M)+1	2*(N-M)+2	2*(N-M)+3	2*(N-M)+4	2*(N-M)+5	2*(N-M)+6
OK		1	-	$M*\lambda_{ON}$	$(N-M)*\lambda_{OFF}$									$S*\lambda_{OFF}$			
Reconfiguration		2		-	1/Treconf	$M*\lambda_{ON}+(N-M-1)*\lambda_{OFF}$									$S*\lambda_{OFF}$		
Loss 1 unit		3			-	$M*\lambda_{ON}$	$(N-M-1)*\lambda_{OFF}$							1/MDT		$S*\lambda_{OFF}$	
Reconfiguration		4				-	1/Treconf										$S*\lambda_{OFF}$
Loss 2 units		5					-									1/MDT (4)	
-	-	-						-									
Reconfiguration		2*(N-M)-2							-	1/Treconf	$M*\lambda_{ON}+\lambda_{OFF}$						
Loss N-M-1 units		2*(N-M)-1								-	$M*\lambda_{ON}$	λ_{OFF}					
Reconfiguration		2*(N-M)									-	1/Treconf	$M*\lambda_{ON}$ (2)				
Loss N-M units		2*(N-M)+1										-	$M*\lambda_{ON}$ (2)				
Indisponible		2*(N-M)+2											-				
	-1 spare	2*(N-M)+3	1/TAT											-	$M*\lambda_{ON}$	$(N-M)*\lambda_{OFF}$	
Reconfiguration		2*(N-M)+4		1/TAT											-	1/Treconf	$M*\lambda_{ON}+(N-M-1)*\lambda_{OFF}$
Loss 1 unit	-1 spare	2*(N-M)+5			1/TAT											-	$M*\lambda_{ON}$
Reconfiguration		2*(N-M)+6				1/TAT											-
Loss 2 units	-1 spare	2*(N-M)+7					1/TAT										
-	-	-															
Reconfiguration		(2*(N-M)+2)*2-4															
Loss N-M-1 units	-1 spare	(2*(N-M)+2)*2-3								1/TAT							
Reconfiguration		(2*(N-M)+2)*2-2									1/TAT						
Loss N-M units	-1 spare	(2*(N-M)+2)*2-1										1/TAT					
Indisponible	-1 spare	(2*(N-M)+2)*2											1/TAT				
	-2 spares	(2*(N-M)+2)*2+1												1/TAT (5)			
Reconfiguration		(2*(N-M)+2)*2+2													1/TAT (5)		
Loss 1 unit	-2 spares	(2*(N-M)+2)*2+3														1/TAT (5)	
-	-	-															
Reconfiguration		(2*(N-M)+2)*(S+1)-4															
Loss N-M-1 units	-S spares	(2*(N-M)+2)*(S+1)-3															
Reconfiguration		(2*(N-M)+2)*(S+1)-2															
Loss N-M units	-S spares	(2*(N-M)+2)*(S+1)-1															
Loss Système		(2*(N-M)+2)*(S+1)															

M :	2
N :	4
S :	2
(2*(N-M)+2)*(S+1) :	18

(1) Reliability/Availability = 1 : Final Loss of the system so less M units active
(2) Reliability/Availability = 2 or 3 : Temporary unavailability so less M units active
(3) Reliability/Availability = 2 : Unavailability of TAT/(S+1) duration if S repairers
Reliability/Availability = 1 ou 3 : Absorbing state
(4) i/MDT I operators, with I lower or equal to the number of spares available
(5) j/TAT if j repairers

Redondancy(M;N;λ_{ON};λ_{OFF};T;Treconf;MDT;Nb_operators;S;TAT;Nb_repairers;Active/passive;Reliability/Availability)



2*(N-M)+7	-		(2*(N-M)+2)*2-3	(2*(N-M)+2)*2-2	(2*(N-M)+2)*2-1	(2*(N-M)+2)*2	(2*(N-M)+2)*2+1	(2*(N-M)+2)*2+2	(2*(N-M)+2)*2+3	-		(2*(N-M)+2)*(S+1)-3	(2*(N-M)+2)*(S+1)-2	(2*(N-M)+2)*(S+1)-1	(2*(N-M)+2)*(S+1)
$S*\lambda_{OFF}$															
			$S*\lambda_{OFF}$												
				$S*\lambda_{OFF}$											$M*\lambda_{ON}$ (1)
			1/MDT (4)		$S*\lambda_{OFF}$										$M*\lambda_{ON}$ (1)
					1/MDT (4)										
							$(S-1)*\lambda_{OFF}$								
								$(S-1)*\lambda_{OFF}$							
$(N–M–1)*\lambda_{OFF}$							1/MDT		$(S-1)*\lambda_{OFF}$						
1/Treconf															
-									1/MDT (4)						
	-														
		-	1/Treconf	$M*\lambda_{ON}+\lambda_{OFF}$											
			-	$M*\lambda_{ON}$	λ_{OFF}										
				-	1/Treconf	$M*\lambda_{ON}$ (2)									$M*\lambda_{ON}$ (1)
					-	$M*\lambda_{ON}$ (2)									$M*\lambda_{ON}$ (1)
						-									
							-	$M*\lambda_{ON}$	$(N–M)*\lambda_{OFF}$						
								-	1/Treconf						
									-						
										-					
											-	1/Treconf	$M*\lambda_{ON}+\lambda_{OFF}$		
												-	$M*\lambda_{ON}$	λ_{OFF}	
													-	1/Treconf	$M*\lambda_{ON}$
														-	$M*\lambda_{ON}$
														1/TAT(3)	-

OPERATING LICENCE AGREEMENT

OF SUPERCAB SOFTWARE PACKAGE

ARTICLE 1 : SUBJECT

The purpose of this Agreement is to define the conditions in which the CAB INNOVATION Company grants the customer with a non-transferable, non-exclusive and personal right to use the software package referred to as "SUPERCAB" and whose features are specified in user's manual.

ARTICLE 2 : SCOPE OF THE OPERATING RIGHT

The customer may use the software package on one single computer and on a second one provided that the second computer does not operate at the same time as the first one. The customer can only have one software package copy maintained in a safe place as a backup copy.

If this license is regarding a performance on site, the customer may install the package software on a server, while scrupulously complying with purchase conditions stated on specific conditions especially defining the maximum number of users authorized to use the software package from their terminal and the maximum number of users authorized to use it simultaneously. The customer is therefore authorized to perform a number of software package documentation copies equal to the maximum number of users allowed to use it..

CAB INNOVATION will be in a position to perform inspections, either itself or through a specialized entity purposefully authorized by CAB INNOVATION, at customer premises to verify if customer has met its requirements : number of software package copies used, location of such copies, etc... Parties will agree as regards the practical modalities of performance of such inspections so as to disturb minimally customer's activity.

ARTICLE 3 : DELIVERY, INSTALLATION AND RECEPTION

The software package and attached supplies will be delivered to the customer on mail reception date. The customer installs, at its own costs, the software package using relevant manual delivered by CAB INNOVATION.
The customer performs the inventory and shall inform CAB INNOVATION, within three working days of the delivery, of any apparent nonconformity with respect to the order. The customer is liable for any loss or any damage caused to supplies as from the delivery.

ARTICLE 4 : TESTING AND GUARANTEE

Guarantee is effective as from the mail delivery date set forth in Article 3 and has a three-month validity.
During the guarantee validity, if the customer experiences a software package operation trouble, he should inform CAB INNOVATION about it, so as to receive any helpful explanations with the purpose of remedying such trouble. If the trouble is continuing, the customer will return the C.D. ROM to CAB INNOVATION, at CAB INNOVATION's Head Office, at his own expense and with registered mail with acknowledgement of receipt, by specifying exactly the troubles encountered.

Within the three months of reception of consignment set forth in preceding paragraph, CAB INNOVATION will deliver, at its own expense, a new product version to the customer. This new version will be benefiting of the same guarantee as benefited the first version.

The customer looses the benefit of the guarantee if he does not comply with the instructions manual recommendations, if he performs modifications of configuration set forth in Article 2 above without obtaining a prior written consent from CAB INNOVATION, or if he performs modifications, additions, corrections, etc... on software package, even with the support from a specialized service company, without obtaining a prior written consent from CAB INNOVATION.

ARTICLE 5 : PROPERTY RIGHT

CAB INNOVATION declares to be holding all the rights provided for by the intellectual property code for SUPERCAB package software and its documentation.

As this operating-right granting generates no property-right transfer, the customer abstains from :
- any SUPERCAB software package reproduction, whether it is wholly or partly carried out, whatever the form assumed, excepting the number of copies authorized in Article 2 ;
- any SUPERCAB software package transcription in any other language than that provided for in this Agreement (see Appendix), any adaptation to use it in other equipment or with other basic software packages de base than those provided for in this Agreement.

To ensure this property protection, the customer undertakes especially to

- maintain clearly visible any property and copyright specifications that CAB INNOVATION would have affixed on programs, supporting material and documentation ;
- assume with respect to his staff and any external person any helpful information and prevention step.

ARTICLE 6 : USING SOURCES

Any SUPERCAB software package modification, transcription and, as a general rule, any operation requiring the use of sources and their documentation are exclusively reserved for CAB INNOVATION.
The customer holds the right to get the information required for the software package interoperability with other softwares he is using, under the conditions provided for in the intellectual property code.
In each case, an amendment of these provisions will set out the price, time limits and general terms of performance thereof.

ARTICLE 7 : LIABILITY

The customer is liable for :
- choosing SUPERCAB software package, its adequacy with his requirements, precautions to be assumed and back-up files to be made for his operation, his staff qualification, as he received from CAB INNOVATION recommendations and information required upon its operating conditions and limits of its performances set forth in user's manual;
- the use made for results he obtains.

CAB INNOVATION is liable for the software package conformity with his documentation. The customer shall prove any possible non-conformity.
CAB INNOVATION does not assume any whatsoever guarantee, whether explicit or implicit, relating to the software package, manuals, attached documentation or any supporting item or material provided and, especially, any guarantee for marketing of any products relating to software package or for using software package for a determined use, any guarantee for absence of forgery, etc...
Under no circumstances CAB INNOVATION could be held responsible for any whatsoever damage, especially loss in performance, data loss or any other financial loss resulting from the use or impossibility to use the SUPERCAB software package, even if CAB INNOVATION was told about the possibility of such damage.
In the event where CAB INNOVATION liability is retained, it is expressly agreed upon that the total amount of compensation to be paid by CAB INNOVATION, all cases taken together, could not in any way exceed the initial-royalty price reduced by 25 % per period of twelve months elapsed as from the mailing delivery date.

ARTICLE 8 : DURATION

This Agreement is entered into for an undetermined period of time as of the date set forth in Article 3.

ARTICLE 9 : TERMINATION

Each party may terminate this Agreement, by registered mail with acknowledgement of receipt forwarded to the other party, for any breach by such party of its obligations, despite a notice remaining unresponsive for 15 days, and this occurring with no prejudice to damages it could claim and provided that the last paragraph of Article 7 above, be enforced.

At end of this Agreement or in case of termination for whatsoever reason, the customer will have to stop using SUPERCAB software package, pay all sums remaining due on date of termination and return all elements composing the software package (computer programs, documentation, etc ...) without maintaining any copy of it.

ARTICLE 10 : ROYALTY

As a payment for the operating-right concession, the customer pays CAB INNOVATION an initial royalty the amount of which is determined in specific conditions.

ARTICLE 11 : PROHIBITED TRANFER

The customer refrains from transferring the software package operating right granted personally to him by these provisions. The customer also abstains from making documentation and supporting material (CD ROM), even free of charge, available to a person not expressly set forth in second paragraph of Article 2.

ARTICLE 12 : ADDITIONAL SERVICES

Any additional services will be subject to an amendment of these provisions, possibly through an exchange of letters, so as to specify the contents, modalities of achievement and the price.

ARTICLE 13 : CORRECTIVE AND PREVENTIVE MAINTENANCE

The corrective and preventive maintenance may be subject, upon customer's request, to a separate Agreement attached to these provisions.

ARTICLE 14 : ENTIRETY OF THE AGREEMENT

The user's manual defining the SUPERCAB software package features is appended to these provisions.
The provisions of this Agreement and his Appendix express the entirety of the Agreement entered into between the parties. They are prevailing among any proposition, exchange of letters preceding its signing up, together with any other provision stated in documents exchanged between the parties and relating to the Agreement's subject matter.
If any whatsoever clause of this Agreement is null and void with respect to a rule of Law or a Law in force, it will considered as not being written though not involving the Agreement's nullity.

ARTICLE 15 : ADVERTISING

CAB INNOVATION could mention the customer in its business references as a SUPERCAB software package user.

ARTICLE 16 : CONFIDENTIALITY

Each party undertakes not to disclose any kind of documents or information about the other party that it would have been informed of on the Agreement's performance and undertakes to have such obligation fulfilled by the persons it is liable for

ARTICLE 17 : AGREEMENT'S LANGUAGE

This Agreement is entered into and drawn up in the French language.
In the event where it is translated into one or more foreign languages, only the French text will be deemed authentic in case of any dispute between the parties.

ARTICLE 18 : APPLICABLE LAW - DISPUTES

The French Law governs this Agreement.
In the event of any disagreement over the interpretation and performance of any whatsoever provision of this Agreement, and if parties fail to reach an agreement under an arbitration procedure, only Toulouse’s Courts will be competent to settle the dispute, despite the plurality of defendants or the appeal for guarantee.