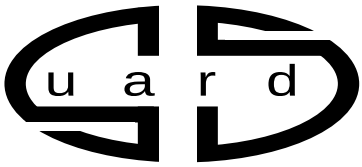
ESPRIT Project 20716



Generic Upgradable Architecture for Real-time Dependable Systems

ALPHA-COUNT MECHANISM AND INTER-CHANNEL DIAGNOSIS

DAVID POWELL

CHRISTOPHE RABÉJAC

ANDREA BONDAVALLI

GUARDS I1SA1 TN 5009 E

9 FEBRUARY 1998 17 pages (11 numbered)

Revision History

Rev.	Date	Comments
A	30.01.98	Initial draft by D. Powell
B	03.02.98	First revision by C. Rabéjac
C	05.02.98	Second revision by A. Bondavalli
D	09.02.98	Revision consolidation by D. Powell, all remarks removed or inserted as Word 6 annotations (can be printed if appropriate Word option is checked).
E	09.02.98	A few bugs corrcted. Revised Section 4.

Table of Contents

This note discusses inter-channel error diagnosis based on the alpha-count mechanism [Bondavalli *et al.* 1997]. Section 1 defines some notation and Section 2 summarises various options for the α-count mechanism. In Section 3, we consider two alternatives for consolidating multiple α-counts into an error syndrome that is consistent on non-faulty channels. Finally, Section 4 discusses use of the error syndrome to carry out fault diagnosis.

1. Notation and definitions[1]

α-cycle	one cycle of the α-count accumulation and consolidation process
α_{ij}	the α-count maintained by channel *i* regarding channel *j*, $i, j \in [1, C]$
α'_{ij}	a local copy of α_{ij}, $i, j \in [1, C]$ (non-negative integer or non-negative real)
$\dot{\alpha}'_i$	vector of α'_{ij}
$\alpha''_{ij}(k)$	channel *k*'s consolidated copy of α'_{ij}, $i, j, k \in [1, C]$ (non-negative integer or non-negative real, or undefined: *)
$\boldsymbol{\alpha}''(k)$	matrix of $\alpha''_{ij}(k)$
A_{ij}	local accusation by channel *i* regarding channel *j*, $i, j \in [1, C]$, $A_{ij} \in \{C, F\}$ (*C*=correct, *F*=Faulty)
$\dot{A}_i$	vector of A_{ij}
$A''_{ij}(k)$	channel *k*'s copy of the consolidated accusation by channel *i* regarding channel *j*, $i, j, k \in [1, C]$, $A''_{ij} \in \{C, F\}$ (*C*=correct, *F*=Faulty)
$\mathbf{A}''(k)$	matrix of $A''_{ij}(k)$
C	number of channels in instance, $C \in [1, 4]$
c	current number of active channels, $c \in [0, C]$
D_i	conclusion of diagnosis concerning channel *i*, $i \in [1, C]$, $D_i \in [C, F]$ (*C*=correct, *F*=Faulty)
$\dot{D}$	global diagnosis, vector of D_i
dec	an integer (decrement)
e_{ij}	number of errors attributed to channel *j* by channel *i* (in a given α-cycle)
f	number of channels diagnosed as faulty

[1] Some of the variables defined here are not used in this note, but are reserved for future use.

κ	a real constant, $\kappa \in [0,1]$
inc	an integer (increment)
m	α-cycle index
N_α	duration of α-cycle (integer number of ICN slots).
N_{frame}	duration of an ICN frame (integer number of ICN slots)
T	assumed maximum number of channels subject to hard faults (requiring passivation)
τ_s	self-accusation α-count threshold
τ_c	cross-accusation α-count threshold

2. Heuristics for α-counts

Each channel *i* maintains an α-count representing its opinion of its own health, α_{ii}, and its opinions of the health of the other channels, α_{ij}, $j \neq i$. We assume that the α-counts are updated and processed cyclically (each such cycle is called α-cycle). The duration of an α-cycle is a parameter of the mechanism that can be chosen such that $N_\alpha . n_1 = N_{frame}$ where n_1 is an integer.

The α-count α_{ij} accumulates an accusation from channel *i* against channel *j* in that it is increased whenever channel *i* detects an error that it attributes to channel *j*. Different accumulation weights can be attributed to different error detection events, for example:

- majority voting discrepancies (including omission due, e.g., to timing errors resulting from channel deadline violations),
- ICN bus transmission errors,
- clock synchronisation errors,
- inconsistent behaviour,
- incorrect channel status (e.g., wrong frame.....)
- …

Here, we assume for simplicity that all errors are given an equal weight.

Each α-count is also subjected to a decay process intended to allow errors from previous cycles to be “remembered” for a certain time before being ”forgotten”.

Many different heuristics can be defined for the accumulation and decay processes. Four possible heuristics inspired from [Bondavalli *et al.* 1997, Rabéjac 1997] are, for example, if $e_{ij}(m)$ is the number of errors attributed to channel *j* by channel *i* during α-cycle *m*, then we could consider the following:

- proportional accumulation, geometric decay on error-free α-cycles

$$\text{If } e_{ij}(m) = 0 \text{ then } \alpha_{ij}(m) = \alpha_{ij}(m-1) * \kappa$$
$$\text{else } \alpha_{ij}(m) = \alpha_{ij}(m-1) + e_{ij}(m) * inc$$

(`alpha1`)

- proportional accumulation, linear decay on error-free α-cycles

$$\text{If } e_{ij}(m) = 0 \text{ then } \alpha_{ij}(m) = \max\left(0, \alpha_{ij}(m-1) - dec\right)$$
$$\text{else } \alpha_{ij}(m) = \alpha_{ij}(m-1) + e_{ij}(m) * inc$$

(`alpha2`)

- proportional accumulation, geometric decay on all α-cycles

$$\alpha_{ij}(m) = \alpha_{ij}(m-1) * \kappa + e_{ij}(m) * inc$$

(`alpha3`)

- proportional accumulation, linear decay on all α-cycles

$$\alpha_{ij}(m) = \max\left(0, \alpha_{ij}(m-1) - dec + e_{ij}(m) * inc\right)$$

(`alpha4`)

By way of illustration, Figure 1 shows a simulation of these four heuristics over 40 α-cycles with two different error distributions over cycles 1 to 20, and no errors during cycles 21 to 40. It can be seen that all four heuristics can provide the appropriate filtering action. It would be interesting to compare the heuristics from a dependability evaluation viewpoint. This would enable a motivated trade-off between the dependability-related performance of the heuristics vs. the simplicity and efficiency of the implementation. However, to be pragmatic, the implementation can adopt any of the four heuristics, whichever is the easiest to implement. Then, a dependability evaluation can be carried out (in parallel with the implementation) in order to fix the parameters of the heuristic in function of an assumed error distribution.

For example, in the case of (`alpha4`), let us assume without loss of generality that we have initially $\alpha_{ij}(0) = 0$ and $\alpha_{ij}(1) > 0$ (in other words, for the sake of notational simplicity, we translate the origin of the α-cycle index to the first cycle where channel *i* detects errors concerning channel *j*). Until $\alpha_{ij}(m)$ becomes equal to 0 again, we remain in a sequence of successive α-cycles (of length at least equal to 1 cycle) in which $\alpha_{ij}(m)$ is always strictly positive. In this sequence, the exact value of $\alpha_{ij}(m)$ is given by: **[DP2]**

$$\alpha_{ij}(m) = -m * dec + inc * \sum_{k=1}^{m} e_{ij}(k)$$

and the α-cycle index m in which $\alpha_{ij}(m)$ will eventually become equal to 0 again is the smallest index such that:

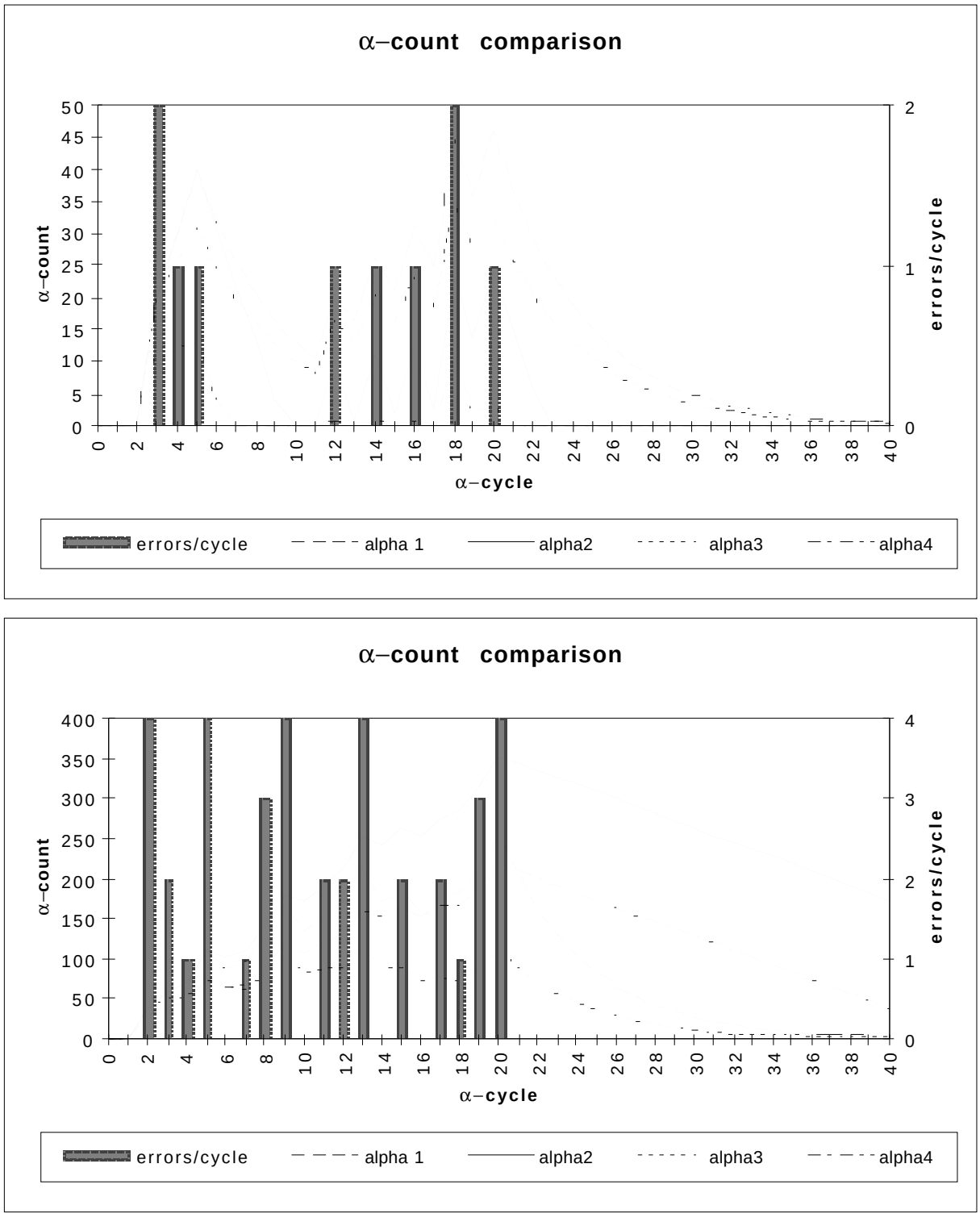
$$\sum_{k=1}^{m} e_{ij}(k) \leq m * \frac{dec}{inc}$$

During the sequence, two cases may eventually arise:

- either $\alpha_{ij}(m)$ increases beyond the accusation threshold τ (in this case, a new diagnosis is carried out, potentially causing the accused channel to be removed from the pool),
- or $\alpha_{ij}(m)$ goes down to 0 again without ever exceeding the accusation threshold τ (in this case, we are put back in a situation in which our initial assumptions holds again).

By considering different distributions for $e_{ij}(m)$, it is possible to exactly tune the respective values of *inc*, *dec* and τ so as to obtain a given latency of ”remembering” and ”triggering”.

Note: it would be wise to foresee different values of the constants *inc* and *dec* (or κ) for α_{ii} (self-judgement) and α_{ij}, $i \neq j$ (cross-judgement).



$inc = 10 \qquad dec = 9 \qquad \kappa = 0.8$

Figure 1 — Simulation of four α-count heuristics

3. Interactive consistency

Since each channel may have a different perception of the errors created by other channels, the α-counts maintained by each channel must be viewed as private values. Each such value represents that channel's accusation either of itself or of another channel. In order for fault-free channels to have a consistent view of the status of the instance, these accusations must be exchanged through an interactive consistency protocol.

Since the α-count mechanism has the effect of "remembering" detected errors over several cycles, it is important to underline that "fault-free" in this context refers to channels that do not create errors during one execution of an interactive consistency exchange (or, equivalently, during a Byzantine agreement exchange). Consensus can be achieved with $C = 4$ (or with $C = 3$, since signed messages are used) if a single channel is faulty during execution of the protocol, but is provably impossible if more than one channel is faulty. So we must make the following assumption:

Assumption A — At most one channel may cause errors during the execution of the interactive consistency exchange.

As for the data exchanged and the corresponding consolidation, two options can be considered:

- direct exchange and consolidation of the α-counts,
- local comparison with α-count thresholds and consolidation of the resulting binary accusations. **[DP3]**

3.1 Consolidation of α-counts

At the end of the last slot of an α-cycle, a local copy α'_{ij} is made of each α-count α_{ij} maintained by channel *i*. These copies are necessary to "freeze" a value that does not change during the cross-channel consolidation and diagnosis phase. While the latter is being carried out (during the next α-cycle), error events can continue to be accumulated in α_{ij}.

At the beginning of an α-cycle, each channel *i* thus possesses a vector of α-counts, $\dot{\alpha}_i$. For *C*=4, we have:

$$\dot{\alpha}'_1 = [\alpha'_{11} \quad \alpha'_{12} \quad \alpha'_{13} \quad \alpha'_{14}]$$
$$\dot{\alpha}'_2 = [\alpha'_{21} \quad \alpha'_{22} \quad \alpha'_{23} \quad \alpha'_{24}]$$
$$\dot{\alpha}'_3 = [\alpha'_{31} \quad \alpha'_{32} \quad \alpha'_{33} \quad \alpha'_{34}]$$
$$\dot{\alpha}'_4 = [\alpha'_{41} \quad \alpha'_{42} \quad \alpha'_{43} \quad \alpha'_{44}]$$

After the interactive consistency exchange of the α-count vectors, each fault-free channel *k* has available a globally consistent α-count matrix, $\boldsymbol{\alpha}''(k)$:

$$\boldsymbol{\alpha}''(k) = \begin{bmatrix} \alpha''_{11}(k) & \alpha''_{12}(k) & \alpha''_{13}(k) & \alpha''_{14}(k) \\ \alpha''_{21}(k) & \alpha''_{22}(k) & \alpha''_{23}(k) & \alpha''_{24}(k) \\ \alpha''_{31}(k) & \alpha''_{32}(k) & \alpha''_{33}(k) & \alpha''_{34}(k) \\ \alpha''_{41}(k) & \alpha''_{42}(k) & \alpha''_{43}(k) & \alpha''_{44}(k) \end{bmatrix}$$

such that, for fault-free channels *i*, *j*:

$$\boldsymbol{\alpha}''(i) = \boldsymbol{\alpha}''(j) \qquad \text{(agreement)}$$

$$\forall k \in [1, C] \; : \; \alpha''_{jk}(i) = \alpha'_{jk} \qquad \text{(validity)}$$

The precise building of matrix $\boldsymbol{\alpha}''(k)$ on channel k follows the general rule for interactive consistency and is done as follows. Row *k* of the matrix is the direct (local) copy of vector $\boldsymbol{\alpha}'_i$. The other rows are obtained through classical majority voting functions. More precisely, for each couple $(i, j), i \neq j$, the consolidated values α''_{ij} are computed in these rows from the received α'_{ij} according to the following schemes:

- For $C = 4$, if at least 2 values out of the 3 received α'_{ij} are equal, then α''_{ij} is set to this majority value, otherwise α''_{ij} is set to the default value "*",
- Similarly, for $C = 3$, if the two received values α'_{ij} are equal, then α''_{ij} is set to this majority value, otherwise α''_{ij} is set to the default value "*".

If a channel is faulty during the first round of the interactive consistency exchange, the corresponding line of $\mathbf{G}(k)$ may have arbitrarily erroneous (non-negative) values (or "*" if inconsistent behaviour was detected). The problem that then occurs is how to combine the α-counts in the columns of the matrix (excluding possibly the diagonal "self-judgements" which must trigger self-isolation of the channels concerned should their values be higher than the self-accusation threshold) in order to make a consistent accusation about each channel. A simple addition of the α-counts will not work, since an arbitrarily faulty channel could dictate its opinion. For the case *C*=4, one possible solution would be to take the median of the three α-counts not on the diagonal. However, this would not be possible for *C*=3, so it is not a generic solution.

A generic solution can be found by converting the consolidated α-count matrix $\boldsymbol{\alpha}''(k)$ into an "accusation matrix" $\mathbf{A}''(k)$ as follows:

$$\forall i \in [1, C], \; A''_{ii}(k) = \begin{cases} F & \text{if } \alpha''_{ii}(k) > \tau_s \\ C & \text{otherwise} \end{cases}$$

$$\forall i, j \in [1, C], \; i \neq j, \; A''_{ij}(k) = \begin{cases} F & \text{if } \alpha''_{ij}(k) > \tau_c \\ C & \text{otherwise} \end{cases}$$

where *F* and *C* denote "faulty" and "correct", and τ_s and τ_c are α-count thresholds for self-accusation and cross-accusation.

Note that the resulting matrix $\mathbf{A}''(k)$ is a consolidated matrix since it is computed from consolidated input values. In particular, for fault-free channels *i*, *j* we have:

$$\mathbf{A}''(i) = \mathbf{A}''(j)$$

due to the agreement property of the interactive consistency exchange: $\boldsymbol{\alpha}''(i) = \boldsymbol{\alpha}''(j)$.

We also have:

$$\forall k \in [1, C] : \mathbf{A}''_{jk}(i) = \begin{cases} F \text{ if } \alpha'_{jk} > \tau \\ C \text{ otherwise} \end{cases}$$

(with $\tau = \tau_s$ if $j = k$ or $\tau = \tau_c$ otherwise) due to the validity property of the interactive consistency exchange: $\forall k \in [1, C] : \alpha''_{jk}(i) = \alpha'_{jk}$.

The resulting binary accusation matrix $\mathbf{A}''(k)$ is a test syndrome that can be processed by a diagnosis algorithm to generate the final passivation decisions (see Section 4).

3.2 Consolidation of binary accusations

The alternative to carrying out interactive consistency on α-count vectors is to first carry out a local comparison with α-count thresholds and then consolidate the resulting binary accusations. Each channel *i* computes its local accusations (at the end of the last slot of each α-cycle) as follows:

$$A_{ii} = \begin{cases} F \text{ if } \alpha_{ii} > \tau_s \\ C \text{ otherwise} \end{cases}$$

$$\forall j \in [1, C],\ i \neq j,\ A_{ij} = \begin{cases} F \text{ if } \alpha_{ij} > \tau_c \\ C \text{ otherwise} \end{cases}$$

At the beginning of an α-cycle, each channel *i* thus possesses a vector of accusations, $\dot{A}_i$, i.e., a local test syndrome. For *C*=4, we have:

$$\dot{A}_1 = [A_{11} \quad A_{12} \quad A_{13} \quad A_{14}]$$
$$\dot{A}_2 = [A_{21} \quad A_{22} \quad A_{23} \quad A_{24}]$$
$$\dot{A}_3 = [A_{31} \quad A_{32} \quad A_{33} \quad A_{34}]$$
$$\dot{A}_4 = [A_{41} \quad A_{42} \quad A_{43} \quad A_{44}]$$

After the interactive consistency exchange of the local accusation vectors, each fault-free channel *k* has available a consolidated accusation matrix, $\mathbf{A}''(k)$:

$$\mathbf{A}''(k) = \begin{bmatrix} A''_{11}(k) & A''_{12}(k) & A''_{13}(k) & A''_{14}(k) \\ A''_{21}(k) & A''_{22}(k) & A''_{23}(k) & A''_{24}(k) \\ A''_{31}(k) & A''_{32}(k) & A''_{33}(k) & A''_{34}(k) \\ A''_{41}(k) & A''_{42}(k) & A''_{43}(k) & A''_{44}(k) \end{bmatrix}$$

such that, for fault-free channels *i*, *j*:

$$\mathbf{A}''(i) = \mathbf{A}''(j) \qquad \text{(agreement)}$$

$$\forall k \in [1, C] : \mathbf{A}''_{jk}(i) = A_{jk} \qquad \text{(validity)}$$

As in the previous case, we obtain a consolidated binary accusation matrix $\mathbf{A}''(k)$ that can then be processed by a diagnosis algorithm to generate the final passivation decisions (see Section 4).

Discussion

Although both approaches guarantee agreement and validity between non-faulty channels, they are not entirely equivalent. Starting from the same initial α-count values, the approaches may lead to different values for the elements of the matrix that concern a channel that is faulty during the first round of the interactive consistency exchange. The difference stems from the binary versus non-binary nature of the variables exchanged by interactive consistency. When binary accusations are exchanged, then for *C*=4, each of the other three channels can only receive one of two values, even if the source channel behaves inconsistently. There is thus always a majority value and thus no default value "*" like in the case when α-counts are exchanged.

4. Diagnosis

Both formulations lead to a classic diagnosis problem, in which all channels (called "units" in the general literature on diagnosis) can test all other channels, and $\mathbf{A}''(k)$ is the resulting test syndrome, as perceived by channel *k*. Since we have shown that interactive consistency ensures that all non-faulty channels have identical copies of the test syndrome, we can now drop the index *k*, without loss of generality.

The diagnosis problem has been extensively studied in the literature. In the current case, the inter-channel tests have imperfect coverage (due to the α-count "filtering") so a faulty channel is not necessarily accused by all correct channels [Lee & Shin 1990, Blough *et al.* 1992, Postma *et al.* 1996].

A diagnosis algorithm is a function that takes as input the test syndrome $\mathbf{A}''$ and returns a global diagnosis vector $\vec{D}$ whose elements D_i represent the diagnosed state of each channel:

$$D_i = \begin{cases} C \text{ if channel } i \text{ is diagnosed as correct} \\ F \text{ otherwise} \end{cases}$$

Note that "correct" (resp. "faulty") in this context means "not requiring passivation" (resp. "requiring passivation").

An ideal diagnosis should be both *correct* and *complete*:

- a diagnosis is *correct* if any channel which is diagnosed as faulty is indeed faulty,
- a diagnosis is *complete* if all faulty channels are diagnosed as faulty.

Unfortunately, it does not seem possible for a diagnosis to be both correct and complete when arbitrary faults can occur (since Byzantine faults may cause inconsistencies that lead to situations where diagnosis becomes undecidable).

Evidently, a channel must be faulty if it accuses itself. Also, under the assumption that only a minority of channels are faulty, a channel must be faulty if a a majority of channels accuse it of being so [Rabéjac 1997]. By applying these two observations, we obtain the following diagnosis algorithm, for C=3 or 4:

$$\forall i \in [1,C],\ D_i = \begin{cases} F \text{ if } \left(A_{ii} = F\right) \vee \left(\left|\left\{A_{ji} = F, j \neq i\right\}\right| > 2\right) \\ C \text{ otherwise} \end{cases}$$

However, this algorithm is only correct and complete under a rather restricted set of assumptions.

One algorithm that admits even arbitrary faults is that given in [Postma *et al.* 1996]. The algorithm is based on four reduction rules (see [Postma *et al.* 1996] for the formal definitions). Rule 1 is applied first, then rules 2 to 4 can be applied repeatedly in any order:

1. a channel must be faulty if it accuses itself;
2. a channel must be faulty if it is accused by, or if it accuses, more than *T-f* non-faulty channels, where *f* is the current number of channels accused of being faulty;
3. a channel must be correct if it is not accused by any correct channels, and it does not accuse any correct channels;
4. a channel for which it is not possible to find a set of other channels such that, if the considered channel is faulty and the other channels correct, results in a consistent solution, must be considered as faulty.

After repeated execution of this set of reduction rules, the authors of [Postma *et al.* 1996] claim that all channels judged to be faulty are indeed faulty, and all channels that are judged to be correct are indeed correct. However, it does not guarantee that a judgement can be made on all channels, so there exists a set of *indeterminate* channels (that contains at least one faulty channel). This set of indeterminate channels can be assumed to be faulty (sacrificing correctness of the diagnosis) or correct (sacrificing completeness of the diagnosis) according to which aspect should be given priority. This depends on the dependability requirements of the considered application.

5. Acknowledgements

This note benefited from discussions held in the GUARDS consortium. The authors acknowledge the contribution given by Fabrizio Grandoni and Felicita Di Giandomentico.

6. References

[Blough *et al.* 1992] D. M. Blough, G. F. Sullivan and G. M. Mason, "Intermittent Fault Diagnosis in Multiprocessor Systems", *IEEE Transactions on Computers*, 41 (11), pp.1430-41, November 1992.

[Bondavalli *et al.* 1997] A. Bondavalli, S. Chiaradonna, F. D. Giandomenico and F. Grandoni, "Discriminating Fault Rate and Persistency to Improve Fault Treatment", in *27th Int. Symp. on Fault-Tolerant Computing (FTCS-27)*, (Seattle, WA), pp.354-62, IEEE Computer Society Press, 1997.

[Lee & Shin 1990] S. Lee and K. G. Shin, "Optimal Multiple Syndrome Probabilistic Diagnosis", in *20th Int. Symp. on Fault-Tolerant Computing Systems (FTCS-20)*, (Newcastle upon Tyne, UK), pp.324-31, IEEE Computer Society Press, 1990.

[Postma *et al.* 1996] A. Postma, G. Hartman and T. Krol, "Removal of Faulty Nodes from a Fault-Tolerant Service by Means of Distributed Diagnosis with Imperfect Fault Coverage", in *2nd European Dependable Computing Conference (EDCC-2)*, (A. Hlawiczka, J. G. Silva and L. Simoncini, Eds.), (Taormina, Italy), Lecture Notes on Computer Science, 1150, pp.385-402, Springer Verlag, 1996.

[Rabéjac 1997] C. Rabéjac, *Inter-Channel Fault Treatment Mechanism*, Matra Marconi Space, France, Guards Report, N°D1A3 AO 2014 B, March 1997.