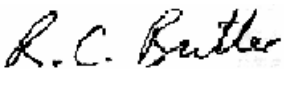
TITLE: # Planck LFI – Fine tuning of REBA parameters: Methods and Software

DOC. TYPE: TECHNICAL NOTE

PROJECT REF.: PL-LFI-OAT-TN–036 **PAGE:** I of IV, 31

ISSUE/REV.: 1.0 **DATE:** 10 August 2006

Issued by	**Michele Maris** **Marco Frailis** **Maurizio Tomasi**	**Date:** **Signature:**	**10 August 06** 
Agreed by	**A. ZACCHEI** **LFI DPC Manager**	**Date:** **Signature:**	**28 Oct 2006** 
Approved by	**R.C. BUTLER** **LFI Program Manager**	**Date:** **Signature:**	**28 Oct 2006** 
Approved by	**N. MANDOLESI** **LFI Principal Investigator**	**Date:** **Signature:**	**28 Oct 2006** 

DISTRIBUTION LIST

Recipient	Company / Institute	E-mail address	Sent
A.Zacchei	INAF-OAT	zacchei@oats.inaf.it	
M. Miccolis	ALCALTE/ALENIA	Miccolis@alenia.it	
R.C.Butler	INAF-IASF-Bologna	butler@iasfbo.inaf.it	
N.Mandolesi	INAF-IASF-Bologna	mandolesi@iasfbo.inaf.it	
L.Stringhetti	INAF-IASF-Bologna	Stringhetti@iasfbo.inaf.it	
M.Bersanelli	Univ. Milano	Bersanelli@fi.unimi.it	
M.Tomasi	INAF-IASF-Milano	Tomasi@iasfmi.inaf.it	
D.Mennella	Univ. Milano	Bersanelli@fi.unimi.it	

CHANGE RECORD

Issue	Date	Sheet	Description of Change	Release
0	2006 Jul 21		Starting of the document	0
0	2006 Jul 24		Description of optimization procedure added	1
0	2006 Jul 25		After A.Z. comments	2
0	2006 Aug 10		After testing and improvements of the procedure and related software during the days between July 31st and August 9th 2006. The optimization criteria has been changed. The list of suggested second_quant values for tuning has been updated. Operative explanations have been extended and updated. M. Tomasi has been joined to the authors.	3

TABLE OF CONTENTS

1 SCOPE 1
1.1 LIMITS OF APPLICABILITY 1
2 APPLICABLE/REFERENCE DOCUMENTS 2
2.1 APPLICABLE DOCUMENTS 2
2.2 REFERENCE DOCUMENTS 2
2.3 ACRONYMS LIST 3
3 INTRODUCTION 5
3.1 THE PROCESSING TYPE PTYPE 5 5
3.2 DEFINITIONS 6
3.2.1 COMPRESSION RATE, TARGET, PACKET, AVERAGED AND PERCENTILES 6
3.2.2 PACKET ENTROPY, AVERAGED AND PERCENTILES 7
3.2.3 NUMBER OF SYMBOLS 8
3.2.4 COMPRESSOR EFFICIENCY 8
3.2.5 R FACTOR 8
3.2.6 PROCESSING ERROR 9
3.2.7 QUANTIZATION ERROR GAIN 10
4 The Optimization Procedure 11
4.1 COMPRESSOR OPTIMIZATION ALGORITHM 13
4.1.1 COMPRESSOR EVALUATION 13
4.1.2 PARAMETERS SPACE ITERATOR 13
4.1.3 STATISTICAL OPTIMIZATION 14
4.1.4 OPTIMIZATION ALGORITHM 17
4.2 INPUT DATA 17
4.3 SOFTWARE 18
4.4 PRODUCTS 18
5 Procedures 19
5.1 REBA TUNING 19
5.1.1 TEST DATA ACQUISITION 19
5.1.2 OCA STARTUP 20
5.1.3 DTOIS GENERATION 21
5.1.4 CREATE THE SCRIPT FOR CRAB AND OPERATE CRAB 22
5.1.5 POST – PROCESSING OF CRAB 23
5.2 TESTING OF REBA PARAMETERS 23
5.2.1 SET-UP OF REBA PARAMETERS AND EXECUTION OF TESTING 24
5.2.2 LIVE ANALYSIS 24
5.2.3 CRAB/PMA ANALYSIS 24
6 A case Example 25
6.1 THE DATA SET 25
6.2 THE RESULTS 25
6.2.1 GMF1 OPTIMIZATION 25
6.2.2 GMF2 OPTIMIZATION 26
6.2.3 SECOND_QUANT OPTIMIZATION 26

6.2.4 OFFSET ADJUST 26
6.2.5 PROCESSING ERROR 26
7 Final Remarks 31

1 SCOPE

This document deals with the technique to optimize REBA parameters for the LFI.

This document supersedes the content of a preliminary description reported in [RD-1] and it is an evolution of the methods described in [RD-2]

1.1 LIMITS OF APPLICABILITY

Presently this document refers to the FM of LFI and to the Issue 2.0 of the compression software described in [AD-2].

The method outlined here in the present form refers to the FM – RAA test campaign.

This method represents an intermediate step toward the final method to be applied during the flight.

Tuning of N_average is outside the scope of this document and it will be not considered here. It is assumed N_average as already tuned with typical values in flight taken from [AD-1] or any other convenient value.

The methods outlined here just assumes N_average > 0 .

Being based on a simulator of the onboard compression code, the method described here does not allow to asses whether the REBA will be able to manage the required computational load (memory and cpu time).

Specialized tests are required for this purpose.

I M P O R T A N T

The content of Sect.6, as of other figures, is based on data taken during a test where the radiometer where not completely tuned.

The purpose of that section is merely illustrative of the method.

NO CONCLUSION SHOULD BE DRAWN ON THE PROPER WORKING OF LFI OR ANY OF ITS PARTS / SUBSYSTEMS / SOFTWARE.

Neither the report or the authors want to do it.

2 APPLICABLE/REFERENCE DOCUMENTS

2.1 APPLICABLE DOCUMENTS

[AD-1] Planck – LFI Communications ICD
M. Miccolis
PL-LFI-PST-ID-013, Version 13, January 2004 or successive

[AD-2] LFI Ground Software Compression Decompression Interface CON
Mónica Sánchez París
DS/SR-FIR/146.v3 Issue 3, April 10 2006

[AD-3] Planck/LFI – Characterization of the Compression Rate for the New Baseline for the Scientific Data Stream Coding
M. Maris, S. R. Hildebrandt
PL-LFI-OAT-TN-029, March 2004

[AD-4] Planck/LFI – Characterization of the On Board Processing Parameters
M. Maris, S. R. Hildebrandt
PL-LFI-OAT-TN-030, March 2004

2.2 REFERENCE DOCUMENTS

[RD-1] Determining REBA parameters in RAA tests
M. Maris
2006 May 12, PL-LFI-OAT-ME-017

[RD-2] Planck/LFI: Estimating the DM Compressor Performances Using the RAA Test Data
M. Maris, M. Guerrini
2006 Apr 7, PL-LFI-OAT-TN-035

[RD-3] Reconfiguration for LFI on-board data processing and scientific telemetry
M. Miccolis, A. Mennella, M. Bersanelli, M. Maris
PL-LFI-PST-TN-037, Issue 1.0, March 2003

[RD-3] Planck LFI – Test Report on the TMH / TQL by Using a Known Signal Test Data
M. Maris, M. Frailis
2006 Mar 20, PL-LFI-OAT-RP-017

[RD-4] Prospects in simulating onboard data acquisition for LFI within the LS – L1 pipeline.
M. Maris
PL-LFI-OAT-ME-015, 21 Feb 2005

[RD-6] Planck LFI – The DTOI Generation Tool and IDL Handling Library
M. Guerrini, M. Maris
Report in preparation - 2006

[RD-7] A Novel Technique for compression enhancement in scientific data communication from satellites
M. Maris, M. Miccolis, C. Burigana
Paper prepared for SPIE 2005, Unpublished

2.3 ACRONYMS LIST

FD	Flight Dynamics
FS	Flight Simulator
L2	Second Lagrangian Point of the Sun-Earth System
LS	Level S
CSW	Compression Software
DTOI	Diagnostic TOI
TOD	Time Ordered Data
TOI	Time Ordered Information
DCSW	De-Compression Software
DPC	Data Processing Center
FM	Flight Model
TBC	To Be Confirmed
TBD	To Be Defined
ADU	Analog / Digital Unit
FM	Flight Model
FP	Floating Point
PType	Processing Type
QM	Qualification Model
TMH	Telemetry Handler
TMU	Telemetry Unscrambler
TQL	Telemetry Quick-Look
OBT	On Board Time
TOI	Time Ordered Information
TOD	Time Ordered Data
Cr	Compression Rate
CrEff	Compressor Efficiency

3 INTRODUCTION

3.1 THE PROCESSING TYPE PTYPE 5

The LFI has a number of modes to process the scientific data before storing them in packets and sending them to ground [AD-1].

According to the kind the kind of processing applied to the data before inserting them into the packets, different processing types (PTypes) are generated [AD-1].

PTypes are numbered from 0 to 6, a table of PTypes is in [AD-1] a short resume is in Tab. 3.1.

The nominal scientific processing, which will be used during the entire mission corresponds to PType 5. The other modes are considered diagnostic telemetry. In particular relevant to our analysis are: raw data (PType 0), downsampled data (PType 1), mixed data (PType 2). PType 1 and 2 are intermediate steps moving from PType 0 to PType 5.

PType 5 processing consists of [AD-1, RD-3]

1. acquire sky and load raw data (PType 0) sampled at 8192 Hz.;
2. Downsample them to a sampling rate equivalent to 1/3 of beam (PType 1);
3. mix sky and load and requantize the mixed quantities in order to asses a proper compression rate (PType 2);
4. compress the data (PType 5).

Details of the processing algorithm and its motivation are in [AD-1, RD-3].

A set of parameters named "*REBA parameters*" are used to fine-tune the processing. Indeed, the ability of the OnBoard compression software (CSW) to assess the required compression rate (Cr) 2 – 3 depends on the statistics of the PType 2 data entering the compressor which in turn are fixed by the fine tuning of REBA parameters and the statistic of the PType 0 data given in input.

The REBA parameters to be tuned are listed in Tab. 3.2. Names are in agreement with [AD-1] but other naming has been used in other documents and these alternative forms are also listed.

Since the statistical properties of the input signal changes from one detection to another chain, the parameters has to be tuned on a chain-by-chain basis (apart from N_average).

In general N_average has to be tuned in order to asses an equivalent sampling rate of PType 1 data equivalent to 1/3 of beam. Consequently its tuning is largely outside the scope of this document and it will not be considered here. Then it is assumed that N_average has been already tuned with typical values in [AD-1].

This document will deal with fine tuning of the remaining parameters GMF1, GMF2, SECOND_QUANT and OFFSET_ADJUST.

Tuning shall asses the following requirements

1. Asses the proper compression rate Cr.
2. Minimizes the processing error (quantization error).

Parameter	*Type*	*Other Symbol*	*Content*
N_AVERAGE	Integer	Naver	Number of raw samples to be coadded for downsampling
SECOND_QUANT	Float	Q	Quantization factor
OFFSET_ADJUST	Float	Offset	Zero point for the re-quantization of data
GMF1	Float	r1	Gain Modulation Factor 1
GMF2	Float	r2	Gain Modulation Factor 2

Tab. 3.2: Table of REBA Parameters according to [AD-1].
The second column denotes the type of the data, the third alternative names or symbols used in literature, other documentation of formulae.

3.2 DEFINITIONS

We shortly recall here some important definitions already described in other documents.

3.2.1 COMPRESSION RATE, TARGET, PACKET, AVERAGED AND PERCENTILES

The Compression Rate, Cr, is defined as the ratio of the length of input string Lin (usually expressed in bytes) and of the output string Lout

$$Cr = Lin / Lout .$$

Usually it is expected Cr to be equal to a particular target value: Cr_Target.

For a stochastic signal compressed in short packets, Cr is a random variable, which is a function of the packet, *pck*. Consequently Cr is replaced by its mean

$$\langle Cr \rangle = (Sum_pck\ Cr_pck)/Number_of_Packets$$

In addition it is convenient to define percentiles of the PDF of Cr_pck.

Cr05 := value of Cr for which less than 5% of the packets has Cr_pck < Cr05.

Cr95 := value of Cr for which less than 5% of the packets has Cr_pck > Cr95.

The optimization criteria is that

Cr_Target > Cr05 for at least 95% of the packets .

For packets the corresponding packet lengths to Lin and Lout are:

Lin_pck = input length of packet pck;

Lout_pck = output length of packet pck;

3.2.2 PACKET ENTROPY, AVERAGED AND PERCENTILES

The Cr is a function of the Shannon Entropy, H, of the signal.

We recall here the Shannon entropy of a data stream coded in binary data containing a list of distinct {Vs} values [1], here s=1, ..., Nsymbols identifies an element in {Vs} and Nsymbols is the number of elements in {Vs}. For each symbol Vs, the frequency of occurrence in the message is denoted by Fs, and of course $\Sigma_s Fs = 1$.

The Shannon entropy is defined as

$$H = -\Sigma_s Fs \cdot \log2\, Fs ,$$

where log2 *x* is the base 2 logarithm of *x*. Note that both for Fs = 0 or Fs = 1 => Fs · log2 Fs =1.

Assuming that all the symbols are coded with the same number of bits, Nbits, the theoretical Cr is

Cr_Theory = Nbits / H ,

for the LFI data sent to the compressor Nbits = 16.

For a normal distributed signal of variance σ^2 quantized with quantization step *q* (or alternatively a quantized normal signal with given σ/q) the entropy is $H \cong \log2[\sqrt{(2\pi e)}\ \sigma/q]$. Then for LFI the expected Cr for a pure white noise signal of given (σ/q)

$$Cr_The,wn(\sigma/q) = 11.090/(1.4189 + \log(\sigma/q)),$$

the expected rms of the quantization error relative to the rms of the signal wil be

[1] If thedata stream contains a constant K then {Vs} has only one member and {Vs}={K}, if even a single sample takes a value K' different from K then {Vs}={K,K'}.

qerr_rel,wn(σ/q) = 0.28868/(σ/q),

both formula are valid for $\sigma/q >> 1$, but they represents a good rule of thumb.

For a signal divided in packets the Entropy of each packet, H_pck, is a random variable changing packet by packet, then it has meaning to define its average and percentiles

<H> = (Sum_pck H_pck)/Number_of_Packets ,

H05 := value of H for which less than 5% of the packets has H_pck < H05.

H95 := value of H for which less than 5% of the packets has H_pck > H95.

Even Cr_Theory has its own packet definition

Cr_Th_pck = Nbits / H_pck .

3.2.3 NUMBER OF SYMBOLS

The number of different values in {Vs} is denoted *Nsymbols*.

Of course even Nsymbols is a random variable for data into a packet.

So Nsymbols_pck will denote the symbols into a packet. Together with this <Nsymbols>, Nsymbols05 and Nsymbols95 are respectively the number of symbols average, at 5% and 95% percentiles.

3.2.4 COMPRESSOR EFFICIENCY

It is interesting to compare Cr with Cr_Theory (or better Cr _pck v.z. Cr_Th_pck).

A comparison is provided by the Compressor Efficiency

Cr_Eff = Cr / Cr_Theory ;

or better

Cr_Eff_pck = Cr_pck / Cr_Theory_pck .

3.2.5 R FACTOR

Input to processing are sky and load signals: (Sky, Load)_t, where *t* is the time at which the sample is acquired. The r_factor is defined as

r_factor = mean(Sky_t) / mean(Load_t) .

3.2.6 PROCESSING ERROR

Input to processing are sky and load signals: (Sky, Load)_t, where *t* is the time at which the sample is acquired. After the processing PType 2 processing reconstructed sky and load signals are obtained (Rsky, Rload)_t.

The processing error is

Esky_t = Rsky_t - Sky_t

Eload_t = Rload_t - Load_t

differences between sky and load will be provided by the instrument

Diff = Sky_t – r_factor · Load_t .

For the processed data

RDiff = RSky_t – r_factor · Rload_t ;

leading to the definition of *total processing error*

EDiff = ESky_t – r_factor · Eload_t ;

Sample by sample the total processing error varies randomly. Statistical indicators are:

qbias_sky = mean(Esky) ;

qerr_sky = rms(Esky) ;

qbias_load = mean(Eload) ;

qerr_ load = rms(Eload) ;

cov_ esky_eload = cov(Esky,Eload) ;

qbias_tot = mean(EDiff) ;

qerr_tot = rms(EDiff) ;

computed over the TOI.

It also useful the following definition

qerr_mean_rms = (rms(Esky) + rms(Eload))/2 .

3.2.7 QUANTIZATION ERROR GAIN

One may wonder whether with the nominal PType 5 processing with optimized parameters it is possible to compress better the data respect to the simple case in which Sky and Load would be sent after simply subtracting a different offset and quantizing with the same quantization second_quant.

This processing is named: SIMPLE_PROCESSING and may be performed by OCA.

In this case:

OFFSET_SKY = - mean(SKY),

OFFSET_LOAD = - mean(LOAD),

and what is sent to the compressor is

Qsky = round((SKY+OFFSET_SKY)*second_quant),

Qload = round((LOAD+OFFSET_LOAD)*second_quant);

The reconstruction procedure is simply

Rsky = Qsky /second_quant + offset_sky,

Rload = Qload /second_quant + offset_load;

the quantization error is then evaluated as before giving

rdiff_simple = Rsky – r_factor * r_load,

qerr_simple = rms(rdiff_simple – diff) .

The gain of the NOMINAL strategy over the SIMPLE strategy is

γerr = qerr_tot/qerr_simple

but usually its inverse is reported in plots 1/γerr.

4 THE OPTIMIZATION PROCEDURE

The optimization is based on the use of the LFI Compression Software simulator to sample the Cr over a set of REBA parameters.

It is assumed that for each radiometer a representative TOI of PType 1 data is provided by the instrument working in conditions which are

1. representative of the operative conditions of the instrument while compressing the true data;
2. stable in time;

Representativeness

means that data shall contain the same kind of signal to be compressed, in particular, it is expected to have a signal with the same sky and load average, rms, correlation coefficient and drift-rates (if any).

Stability

means that the statistical features of the input data set does not change in a relevant way during the operations.

Fig. 4.1 outlines the optimization procedure.

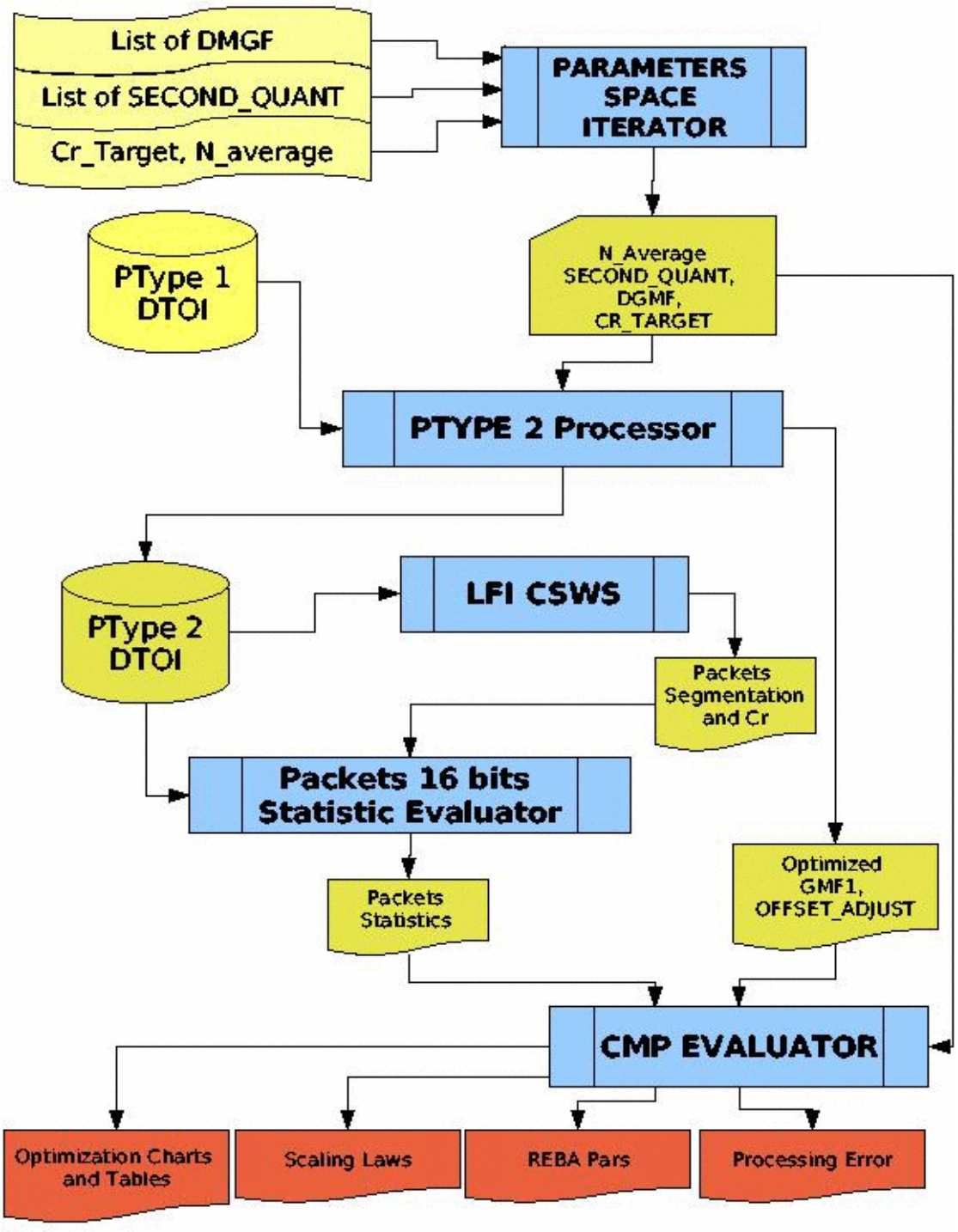


Fig 4.1 Outline of the optimization process.

4.1 COMPRESSOR OPTIMIZATION ALGORITHM

4.1.1 COMPRESSOR EVALUATION

For a fixed set of REBA parameters: (N_average, offset_adjust, second_quant, gmf1 and gmf2) a DTOI of PType 1 data for the radiometert to be optimized is transformed into a DTOI of PType 2 data. Warnings are issued if the so-generated PType 2 data are overflowed or underflowed.

The DTOI of PType 2 data is compressed by the LFI CSWS. The CSWS provides for each packet the list of uncompressed samples in the packet as the Cr.

From the list of compressed samples, a packet-by-packet statistical evaluator splits the DTOI of PType 2 data and for each packets provides the needed 16-bits statistics so to allow correlation of Cr with the statistical properties of data in each packet.

4.1.2 PARAMETERS SPACE ITERATOR

The scope of the Parameter Space Iterator is to provide the list of REBA parameters to be tested and to operate on them the compressor.

In the following the symbol {.} will denote a list of values.

The iterator determines the REBA parameters according to the following scheme:

As already discussed N_AVERAGE is assumed to be known.

The GMF1 is optimized for the input DTOI in particular

GMF1 = r_factor .

The GMF2 is obtained through a list of GMFs differences

DGMF := GMF2 – GMF1;

the user provides the list of DGMF and once GMF1 has been fixed the list of GMF2 is generated

{GMF2} = GMF1 + {DGMF} .

The OFFSET_ADJUST is automatically generated after GMF1 and GMF2 are known

OFFSET_ADJUST = - <sky> +0.5·(GMF1+GMF2) <load> .

The SECOND_QUANT is taken from a list of {SECOND_QUANT} values also provided by the user. Typical values for SECOND_QUANT ranges between 0.05 and 8.

The number of combinations to be tested is then

$$NSYM = ||\{SECOND_QUANT\}|| \times ||\{DGMF\}||$$

where ||{x}|| is the number of elements of list {x}.

The iterator compiles for each radiometer a table of statistical measures of the performances of the CSWS.

Statistical indicators are defined in Sect. 3.2.

4.1.3 STATISTICAL OPTIMIZATION

Statistical optimization is obtained from the table of statistical indicators of performances.

The tables of statistical indicators are used to generate a set of "*Optimization Charts*".

A type of Optimization Chart is plotted in Fig. 6.1.

The following features are evident:

1. The Cr has a maximum for GMF1 ~ GMF2 as expected from theory.
2. Taken separately the rms of sky and load quantization errors
    1. are monotonously increasing or decreasing (no minima or maxima are present),
    2. diverge as 1/|GMF2 – GMF1| for GMF2 – GMF1 going to 0,
    3. have two crossing points or "nodes" for
        GMF2 – GMF1 = -(1 + GMF1) , or GMF2 = -1,
        GMF2 – GMF1 = 1 – GMF1 , or GMF2 = 1,
    4. at the nodes
        qerr_sky = qerr_load = qerr_rms_mean .
3. The quantization error for differentiated data qerr_tot, in the case GMF1 = r_factor is independent from GMF1 and GMF2 and their difference and has value

    qerr_tot(second_quant | GMF1 = r_factor) = 1/(second_quant*sqrt(12)) ,

4. Both qerr_tot, qerr_sky, and qerr_rms_mean are asymmetrical with respect to DGMF = 0, smaller errors are obtained for DGMF < 0, or GMF2 < GMF1.
5. The log(Cr) and log(qerr_tot) scales proportionally to log(SECOND_QUANT) as clearly displayed in the plot of Fig 6.3.

The constancy of qerr_tot(second_quant | GMF1 = r_factor), is strictly related to the hypothesis

GMF1 = r_factor

during the optimization procedure the r_factor is estimated on a short data chunk (typically minutes), while during the data analysis long data chunks will be used to determine the better r_factor (typically hours or days). The some difference between the r_factor and GMF1 has to be expected.

We define with ε, the relative difference between GMF1 and the r_factor

$$\varepsilon = (GMF1 - r_factor)/r_factor$$

and then qerr_tot will be a function of ε.

From this the sensibility to qerr_diff to variations of the r_factor is quantified by the susceptibility to r_factor variations

$$Sr(\delta, \varepsilon) = \{[qerr_tot(GMF1, \delta, \varepsilon) - qerr_tot(GMF1, \delta, \varepsilon=0)] / [\varepsilon \cdot qerr_tot(GMF1, \delta, \varepsilon=0) \}] ,$$

where $\delta = (GMF2 - GMF1)/GMF1$. Sr is at first order independent of GMF1, ε, or second_quant

$$Sr = -1/\delta .$$

A Sr = 1 means that a 1% difference in the r_factor with respect to GMF1 will cause a 1% variation in qerr_tot with respect to the ideal case.

In order to maximize the compression rate one would like to take DGMF as small as possible (ex.: $|DGMF| \sim 10^{-2}$) but this will cause |Sr| to be of order 10^2 or greater, leading to an extreme sensitivity to the effective value of the r_factor.

The the most conservative solution is to take

$$GMF1 = r_factor \sim 1 , DGMF < -1$$

a convenient solution is to take DGMF at the intersection of qerr_rms_mean and qerr_tot i.e. to solve the equation

qerr_rms_mean(DGMF) = qerr_tot(DGMF)

for given GMF1 = r_factor. The solution is independent with good accuracy on second_quant.

As a rule of thumb, having for all of the radiometers r_factor ~ 1 the optimal GMF2-GMF1 will be

$$GMF2 - GMF1 \sim -1.23 ,$$

equivalent to

GMF2 ~ -0.2 ,

with

Sr ~ 0.8 ,

and with

qerr_tot ~ 0.28/ second_quant ,

qerr_mean_rms ~ 0.28/ second_quant ,

qerr_sky ~ 0.24 / second_quant ,

qerr_load ~ 0.33 / second_quant ,

cov_esky_eload ~ 0.047 / second_quant^2 .

4.1.4 OPTIMIZATION ALGORITHM

From this the optimization algorithm follows

1. Unscaled qerr_mean_rms(DGMF), qerr_tot(DGMF) and cr(DGMF) curves (denoted as qerr_mean_rms1, qerr_tot1 and Cr1) are derived from the Optimization Chart removing the scaling law.
2. The DGMF corresponding to the intersection
   qerr_mean_rms1(DGMF_best) = qerr_tot1(DGMF_best)
   is determined.
3. The unscaled Cr_best1 = Cr1(DGMF_best) is derived.
4. From the Cr_best1 and the scaling law the optimal SECOND_QUANT for the given cr_target is derived.
5. The procedure is repeated for Cr05, <Cr> and Cr95 to provide an optimal set of parameters and their uncertainties.
6. A verification stage is performed repeating the full processing (mixing, quantization, compression and reconstruction) to asses the goodness of the solutions recovered by the code.

Combinations of REBA paramters leading to overflow or underflow in the onboard processing have to be avoided.

4.2 INPUT DATA

About 15 – 30 minutes of representative data has to be provided.

Input data has to be provided in the form of DTOI of PType 1.

Data has not to be per

The procedure to have these data is:

1. acquire for a subset of radiometers an uninterrupted data stream in PType 1 format with the required N_average;
2. operate TQL/TMH to generate TOI;
3. operate the DTOI generator in OCA to have DTOI;
4. repeat steps 1, 2 and 3 till all the radiometers are acquired.

This procedure is needed since, at least during RAA tests, It is not possible to acquire PType 1 data for all of the radiometers at once.

For step 1 we suggest to operate TQL during the set-up step for the acquisition, then stop the TQL and restart without to stop the REBA. In this way an unperturbed data stream will be generated.

In addition we suggest to generate as many separate tests as many groups of radiometers have to be acquired.

4.3 SOFTWARE

The software for this procedure is part of the CRAB/CMPS toolkit, which is part of OCA.

OCA is part of the LAMA package which is included in the LIFE environment.

OCA can be operated both as a part of LIFE/LAMA and as a standalone product. Nevertheless it is distributed as par of LIFE/LAMA.

CRAB also supports another method (CRAB/MODEL) based on a model of Cr_pck as a function of H_pck and N_symbokls_pck, but the method outlined in this document is more general.

4.4 PRODUCTS

The procedure provides a set of outputs:

1. Compression Optimization Charts;
2. Packet statistics, in particular, Cr_Eff_pck as a function of H_pck;
3. TOI statistics;
4. Optimized REBA parameters;
5. Estimates of expected processing noise.

The outputs are provided in the following formats:

1. Latex report, with all the relevant tables, and on the request, the relevant figures (to be compiled by the operator);
2. Postcript version of all of the plots generated by the analysis, one set for any detector;
3. csv file containing all the statistical information, the optimization information and the related parameters; the file contains one row for each detector, columns are separated by “,”;
4. cvs file containing only the optimized REBA parameters and the related FH, RAD and DTC to be used by scripts automatically generating the telecommands for the REBA.
5. .save files containing the data base of simulations

The database of simulations is used to repeat optimization, plotting and reporting without to repeat the simulation which is time consuming.

To do this the data base has to be moved inside the source directory containing the data and the appropriated flags in the scripts has to be modified. Then the test will be runned, skipping simulations but reading data from the database.

5 PROCEDURES

This section describes from the operative point of view the procedure for the REBA optimization and the evaluation of the goodness of REBA optimizzation.

5.1 REBA TUNING

REBA tuning is performed for each radiomenter in a given "status".

The "status" of the radiometer is defined by all the environmental parameters (ex.: temperatures in the chambers), electrical parameters (ex.: bias and gains of amplifiers), set-up parameters (ex.: Naverage or ADC setup) able to determine the statistical properties of the signal.

Any change in the status will require a new tuning of REBA parameters.

5.1.1 TEST DATA ACQUISITION

Representativeness requires to have the same setup during acquisition that will be used during the operations.

Stability requires that no transients or gaps due to set-up procedure, or any accidental reason, shall be present in the data.

About 15 – 30 minutes of representative data has to be provided.

Input data has to be provided in the form of DTOI of PType 1.

The procedure to have these data is:

1. acquire for a subset of radiometers
    1. an uninterrupted data stream in PType 1 format;
    2. with the required N_average;
    3. in the Processing Group A;
2. operate TQL/TMH to generate TOI;
3. operate the DTOI generator in OCA to have DTOI;
4. repeat steps 1, 2 and 3 till all the radiometers are acquired.

This procedure is needed since, at least during RAA tests, It is not possible to acquire PType 1 data for all of the radiometers at once.

For step 1 we suggest to operate TQL during the set-up step for the acquisition, then stop the TQL and restart without to stop the REBA. In this way an unperturbed data stream will be generated.

In addition we suggest to generate as many separate tests as many groups of radiometers have to be acquired.

5.1.2 OCA STARTUP

In order to start OCA it is sufficient to start LIFE and then LAMA, after which the IDL command

```
IDL> start_oca_session
```

without parameters will automatically start a session of OCA adding the proper paths.

No graphical interface is provided with OCA.

The `start_oca_session` is a procedure which allows to specify the path for input and output of data. Unless specified at startup it is assumed that

- input are taken from a directory test_data in the same directory tree of life,
- utput are written into the oca subdirectory within the directory created by LAMA at startup a directory `test_data` in the same directory tree of life,
- the operative system is LINUX [2].

The startup procedure may be runned as many times as wanted (as an example to setup a different ROOT_IN or ROOT_OUT) but the packets will be linked to LAMA at the first run only.

We suggest also to run once the command `@make_oca.pro` if some modification has been done to any part of the code during a session.

Example:
The user `life_operator` with the following home directory

```
/home/life_operator
/home/life_operator/life
/home/life_operator/test_data
```

starts a LAMA session "`lama-Wed-09aug06-11:44:00`" and then starts OCA without any specification it will create

```
/home/life_operator/lama-Wed-09aug06-11:44:00/oca
```

which will contain the results, while OCA will assume to read input data from

```
/home/life_operator/test_data
```

[2] OCA has been created to be operated under LINUX. At LFI /DPC Any kind of support to WINDOWS users is ***refused*** by the authors. Use under MS-WINDOWS in principle is possible but it is discouraged and it is at the risk of the user. Some third-party executables are not distributed under LINUX.

A message showing the root for input data (ROOT_IN) and for output data (ROOT_OUT) is printed at startup.

If the test "XXX_0037" will be analysed the data will be readed from

```
/home/life_operator/test_data/XXX_0037
```

and final results will be stored in

```
/home/life_operator/lama-Wed-09aug06-11:44:00/oca/XXX_0037/analysis
```

while temporary results will be stored in

```
/home/life_operator/lama-Wed-09aug06-11:44:00/oca/XXX_0037/tmp
```

or in another directory usually destroyed at the end of the operations.

For installation and other details about the startup see the README file in the distribution of OCA.

Templates of self-explanatory scripts needed for the operations described below are stored in

OCA/SCRIPTS_TESTS

the directory is divided in subdirectories according to the package to be used.

Scripts have to be modified to analyse specific tests.

In most cases it is enough to specifying the test to be operated and the list of feed-horns to be analyzed.

To run scripts use the idl command

IDL>.run SCRIPT_NAME

if the adapted scripts are left in the subdirectory containing their templates they are automatically identified by IDL.

5.1.3 DTOIS GENERATION

After the generation of TOIs (see TQL/TMH user manual) DTOI has to be generated by OCA.

A self explanatory script template is in

OCA/SCRIPTS_TESTS/MAKE_DTOI

DTOIs are automatically moved into

```
test_data/TEST_NAME/dtoi
```

at the end of their generation.

Example:
the DTOIs for test XXX_0037 of the previous example will be stored into

```
/home/life_operator/test_data/XXX_0037/dtoi
```

5.1.4 CREATE THE SCRIPT FOR CRAB AND OPERATE CRAB

Parameters optimization is performed by CRAB.

A templates script is in

OCA/SCRIPTS_TESTS/CRAB

the user is required to specify not only the test and the list of feed-horns, but for each feed-horn it has to list

1. the minimum and maximum index of the samples in the DTOI to be used (usually 1 and a number below the number of samples in the TOI);
2. the Naverage to be applied to each feed-horm;
3. the target Cr (Cr_Target);
4. the list of GMF1 values (usually forced to [!VALUES.D_NAN]);
5. the list of OFFSET values (usually forced to [!VALUES.D_NAN]);
6. the list of list of DGMF values to be explored (LST_DGMF, at least 2);
7. the list of second_quant values to be explored (LST_QUANT, at least 2) .

In fixing the list of indexes specifying the portion of TOI to be used it is helpful to explore preventively the various TOIs in order to avoid transients.

In addition a list of flags are provided to modify the operations of CRAB the most important are

1. the index in the list of the first and the last feed-horns to be analysed (usually 0 and the number of fee-horns – 1)
2. whether or not
    1. to RESTORE the simulation data base
    2. to perform SIMPLE analysis
    3. to RESTORE the data base of simple analysis
    4. to include figures in latex

other flags are illustrated in the template.

We suggest to name the script as the test, as an example: crab_XXX_0037.pro. Which is runned with

IDL> .run crab_XXX_0037.pro

Depending on the number of samples, the number of second_quant and DMGF tested CRAB may takes one or two hours to complete its operations.

In fixing LST_ DGMF consider that usually proper values of DGMF are in the range -1.5, -0.5. So a fast analysis may be performed with

LST_DGMF = [-1.5, -1.25, -1., -0,75, -0.5],

In fixing LST_QUANT consider that usually proper values of second_quant are in the range 0.1 – 8, and that the accuracy of the optimization quickly degrades if it has to extrapolate the optimal second_quant out of the explored range of values.

An iterative procedure may be used in which a first run is performed to identify the interesting region of parameters and then a refinement is made.

5.1.5 POST – PROCESSING OF CRAB

After the processing the data are stored into the "analysis" subdirectory within the output path.

In the example above

```
/home/life_operator/lama-Wed-09aug06-11:44:00/oca/XXX_0037/analysis
```

The user is responsible to compile the latex report and to convert it in PS or PDF.

In addition the user is responsible to store the .save files containing the data base of simulations into the test_data subdirectory. For the example above the database has to be moved to test_data/XXX_0037/analysis with the command

cp -r `oca/XXX_0037/analysis /home/life_operator/test_data/XXX_0037`

which will overwrite the old version of the database (if any).

5.2 TESTING OF REBA PARAMETERS

This step is performed using the REBA.

Its minimal task is to asses that REBA is able to reach the expected performances.

5.2.1 SET-UP OF REBA PARAMETERS AND EXECUTION OF TESTING

The following procedure is required

1. REBA parameters are extracted from REBA_pars.csv file and set to the REBA [3].
2. Data are acquired for about 15 minutes in PType 5 on Processing Group A.

It is expected that all of the detectors may be acquired at once.

In addition it should be possible to acquire at least one feed-horn using PType 1 on Processing Group B together with PType 5.

5.2.2 LIVE ANALYSIS

The rate by which packets are displayed in the TQL are a measure of the Cr obtained. The comparison with the data acquired in PType 5 processing will give a suggestion of the Cr.

5.2.3 CRAB/PMA ANALYSIS

A more quantitative analysis is the CRAB *Post Mortem Analysis* (CRAB/PMA) .
For the CRAB/PMA TOIs and DTOIs are generated after the end of the test.
Then the CRAB/PMA packed is operated by using a script into SCRIPTS_TESTS/PMA.
The script is a simplified version of the CRAB script and does not need any further explanation.
The scripts generates plots and a latex report to be compiled.

[3] The definition of the mechanism for this action is out of the scopes of this document.

6 A CASE EXAMPLE

We discuss here a preliminary example obtained by using data from the FM, RAA test XXX_0078.

The test HAS NOT BEEN DESIGNED for the purpose of compression testing / validation or optimization. Consequently not all of the radiometers has been properly tuned.

Nevertheless this data set is sufficiently good to exercise at the DPC with the overall procedure.

6.1 THE DATA SET

The XXX_0078 data set contains data acquired at the FM testing site 3 August 2007.

The data stream is acquired with PType 1, and nominal N_average for all of detectors.

The data stream contains one hour of data, with some drifts and transients in some of the detectors, and detectors acquired in two groups.

DTOIs have been generated for all of the data and the full data set.

After visual inspection of the data stream we select an unperturbed segment of DTOI containing up to 10^5 samples, equivalent to about 30 minutes of data.

The optimization has been carried out for 5 values of second_quant (0.5, 1, 2, 4, 8) and 20 values of DGMF ranging from -4 to +4 with the range -2.5 ÷ -1.5 sampled with step 0.1.

The assumed Cr_Target to be reached by at least 95% of the packets has been

Cr_Target = 2.4 .

6.2 THE RESULTS

The optimization code requires about 2.5 hours to generate a complete set of optimization charts on a normal PC.

6.2.1 GMF1 OPTIMIZATION

The Tab. 5.1 shows the results for the GMF1 optimization obtained according to the convention

GMF1 = r_factor

the range of r_factors is quite narrow; therefore GMF1 in the range 0.95 ÷ 1.05 are selected.

6.2.2 GMF2 OPTIMIZATION

Analogously to the GMF1 parameter, also optimal DGMF and then GMF2 range over a narrow interval.

In this data set the optimal DGMF is between -1.5 ÷ -1.1 and the optimal GMF2 is -.2 ÷ -.1.

6.2.3 SECOND_QUANT OPTIMIZATION

Due to the higher variation in the RMS of the data, SECOND_QUANT largely varies from detector to detector. In cases of detectors with anomalously low noises very high values of SECOND_QUANT (in the range 10 ÷ 20) are required.

Values of CrEff in the range 0.75 – 0.8 are reached in agreement with [RD-2].

6.2.4 OFFSET ADJUST

The procedures optimizes the offset_adjust to take its values in the range -15000 ÷ -6000. No overflow or underflows has been detected in the data.

6.2.5 PROCESSING ERROR

With optimized GMF1 and GMF2 the processing error is typically in the range 0.1 ÷ 0.5 ADU in some cases 1 or 2 ADU are reached. No conversion to temperature has been possible at the epoch of the test, but it can be compared this data with a rms of the white noise of ~6 ADU. The quantizatoin noise however is always 10% - 20% of the white noise. The 1/γerr is in the range 1.5 – 2.5.

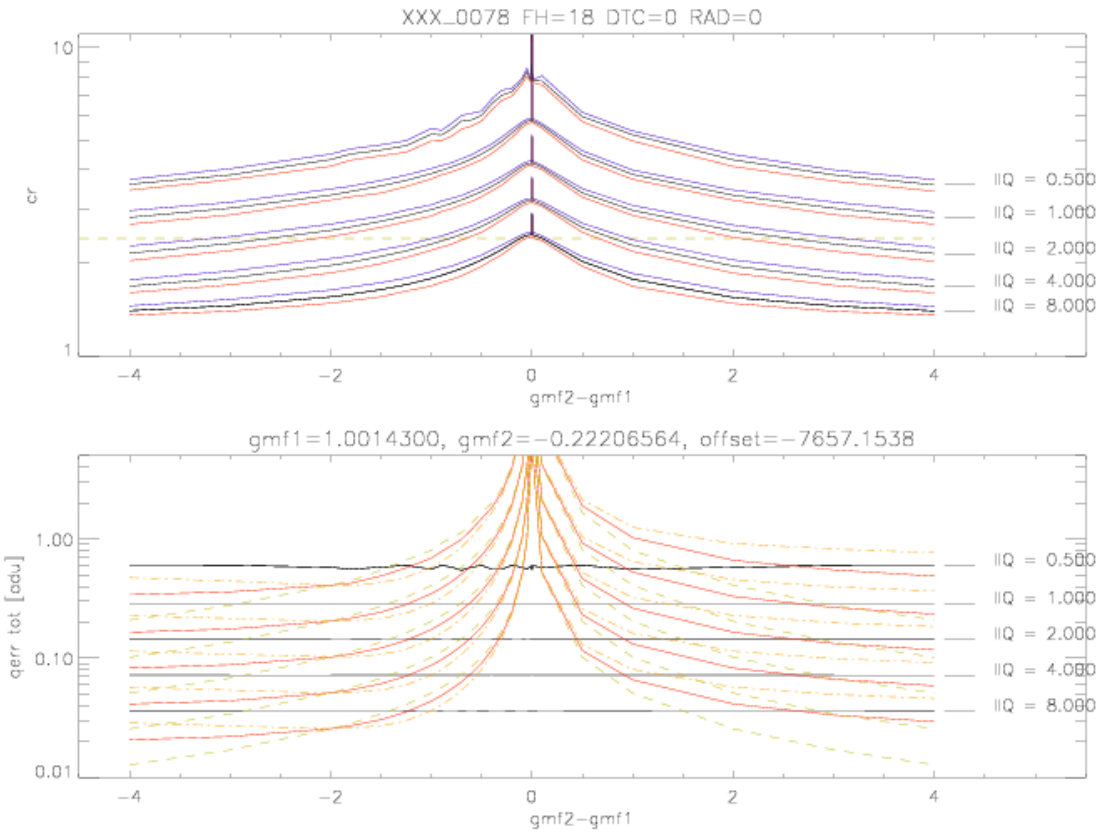


Fig 6.1 Example of optimization chart.
The chart is for test XXX_0037, FH = 18, DTC = 0, RAD = 0.
The upper frame is Cr as a function of GMF2 – GMF1. The plot reports <Cr> (full-black), Cr05 (full-red) and Cr95 (full-blue). Different values of SECOND_QUANT (abbreviated with IIQ) are reported. The yellow-dashed line is the targer Cr = 2.4.
The Lower Frame are total quantization errors in adu as a function of GMF2-GMF1 and for the same range of SECOND_QUANT. Data shown are: qerr_tot (full black), qerr_sky (dot – dashed yellow), qerr_load (dashed green), qerr_mean_rms (red full). The the optimal GMF1, gmf2 and OFFSET are also displayed.

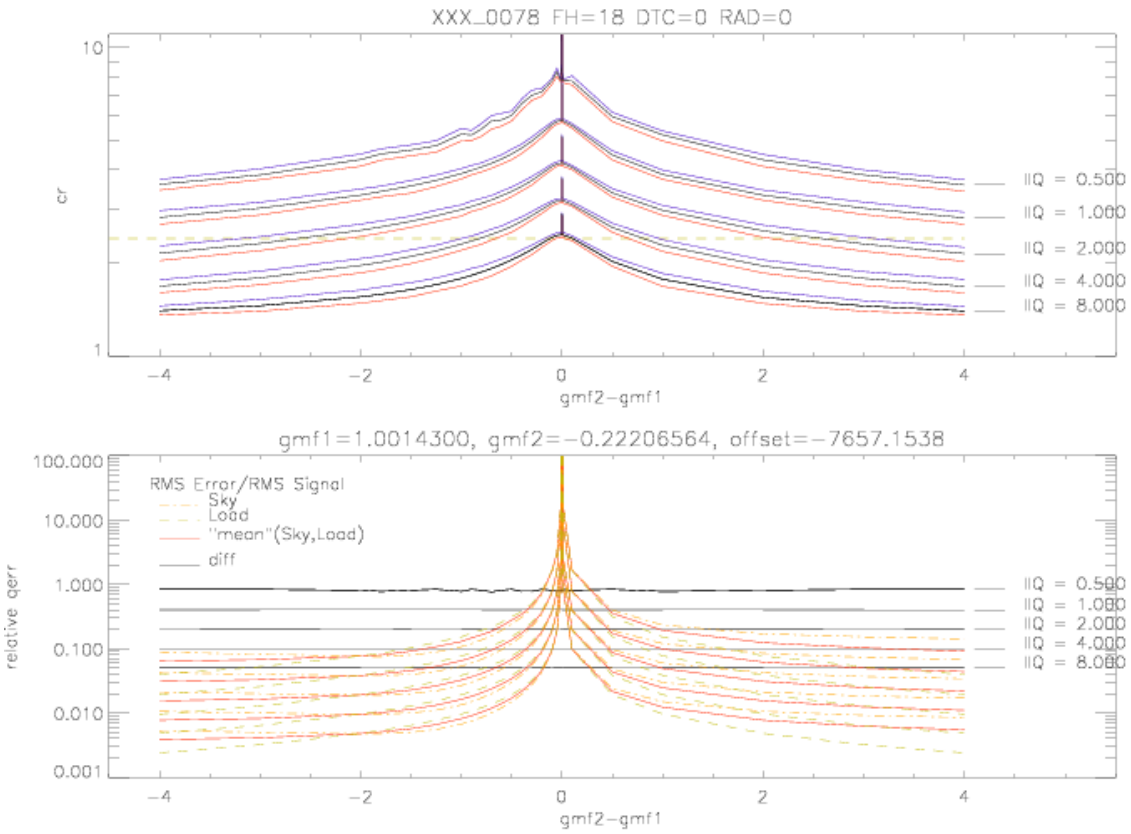


Fig. 6.2 Example of optimization chart with relative errors.
The same of Fig.5.1 but here processing errors relative to the rms of the data is displayed.

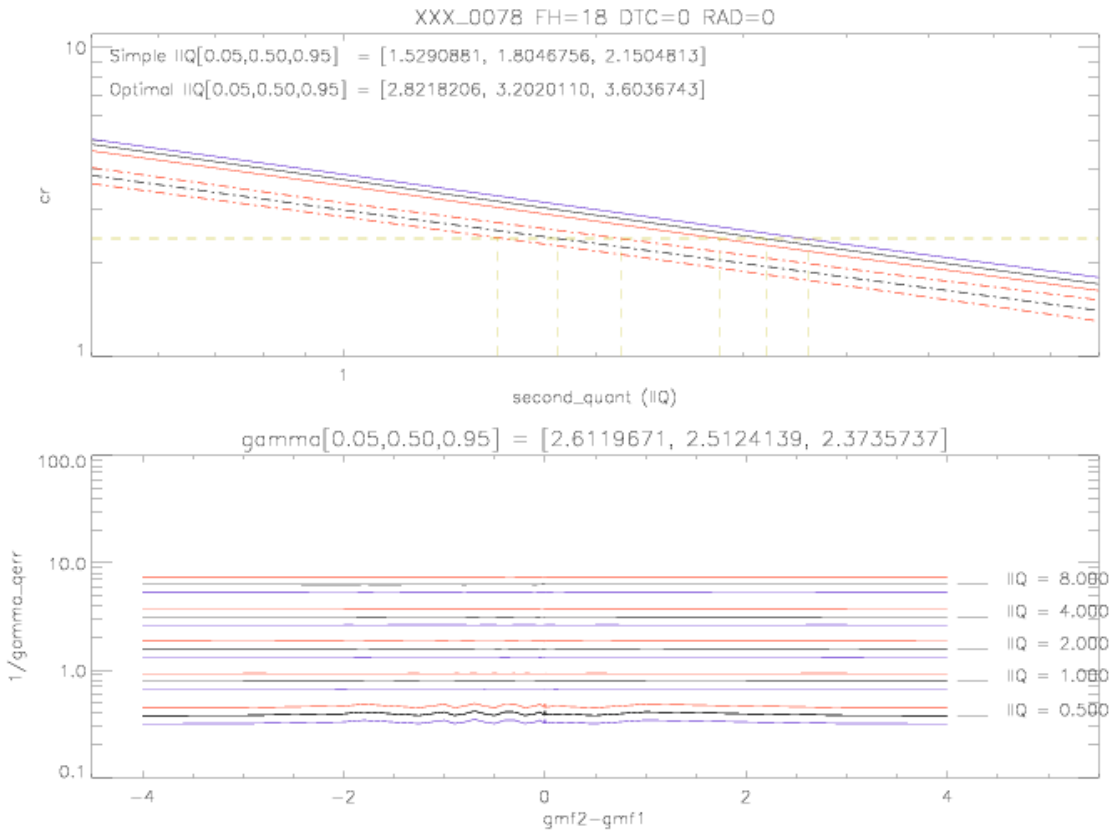


Fig. 6.3 Scaling factor for Cr and gain for the quantization error.
The chart is related displays how the compression rate scales with second_quant. Full lines for the nominal processing. Dashed lines for the SIMPLE processing. Red for Cr05, Black for Cr50 and blue for Cr95. Below the ratio between the quantizatoin errors for the simple and the nominal processing. The numbers between [.] in the top frames are the quantization errors.

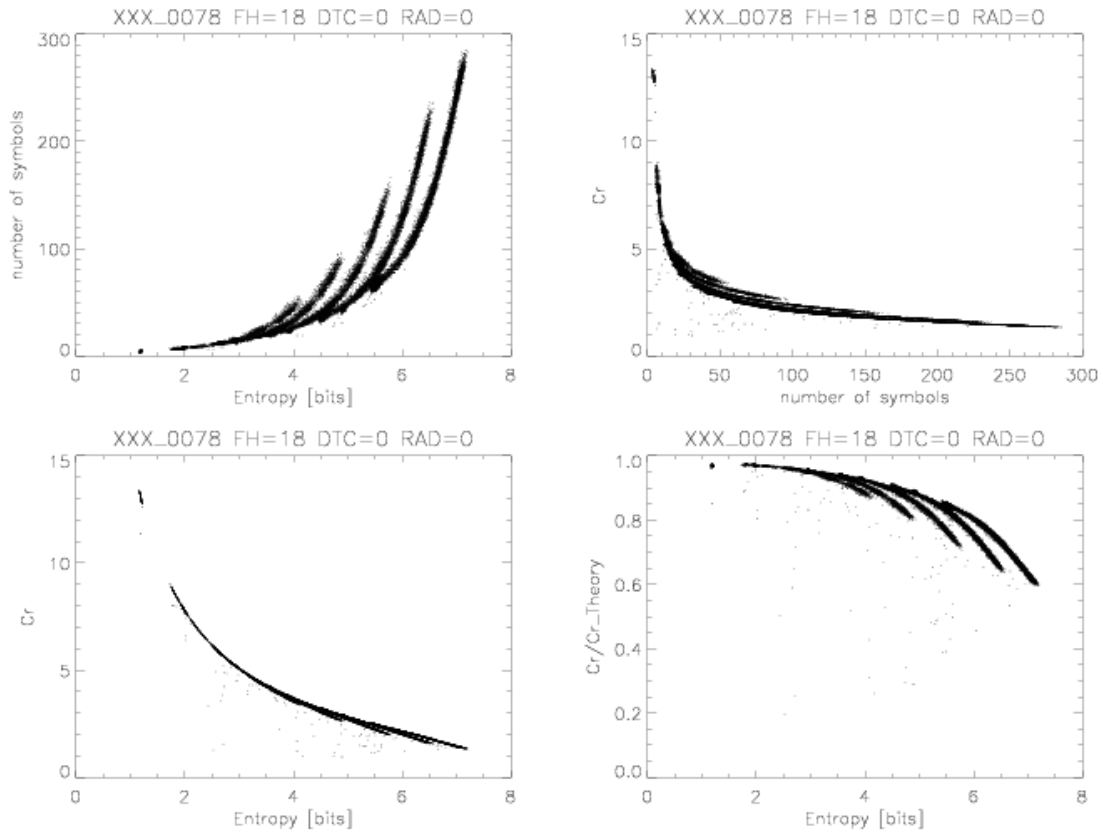


Fig. 6.4 Number of symbols in each packets as a function of entropy of the packet, Cr for the packet as a function of entropy and Number of symbols, Efficiency of the compressor for each packet.
The "fingers" structure corresponds to different values of second_quant. Each structure gives an idea of the level of variability of Cr from packet to packet.

7 FINAL REMARKS

This preliminary analysis shows that it is possible to optimize the REBA parameters with the methods and the software described in this report. i.e. starting from a data set representative of the data to be compressed.

The method assumes that the instrument is stable for time scaled longer than the duration of the optimization cycle which typically is of about 6 hours including the acquisition of the data set, the generation of TOI and DTOI, the execution of the optimization code and the application of optimized parameters.

In addition it shows that, at least using the compression simulator, it is possible to reach the target compression rate, keeping a limited processing error.

The method is not able to provide any indication about the CPU load or the memory consumption.

Specific tests will be required to asses that the same results can be achieved with the compressor on-board the REBA.