NA48 Note **October 1995**

Part I:

Simulation of a Parallel Scheme for the Online Calibration of the NA48 Electromagnetic Calorimeter

Part II:

Review of some Calibration Methods for Electromagnetic Calorimeters

Claudio Bruschini, **CERN/CN & GP-MIMD2 Project**

Don Cundy, **CERN/PPE** **and** ***Alan Norton,*** **CERN/ECP**

Abstract.

The first part of this paper describes and tries to assess the potentialities of a calibration scheme for the NA48 LKr calorimeter based on the use of (background) Ke3 events. It then focuses on an initial parallel implementation of such a scheme, carried out on the CERN Meiko CS-2 platform and targeted at quasi real-time use. Both analyses take as starting point the event-level parallel version of NA48's NMC fast simulation code.

The second part reviews some of the "standard" techniques employed to calibrate (large) electromagnetic calorimeters; they could be used in addition or as crosscheck to the one described in the first part.

Part I:

Simulation of a Parallel Scheme for the Online Calibration of the NA48 Electromagnetic Calorimeter

1. (LKr) Energy Resolution vs. Intercalibration Error

A calorimeter's resolution is usually parametrized in the following way:

$$\left(\frac{\delta E}{E}\right)^2 = \left(\frac{a_0}{E}\right)^2 + \left(\frac{a_1}{\sqrt{E}}\right)^2 + b^2 \quad \text{or} \quad \frac{\delta E}{E} = \frac{a_0}{E} \oplus \frac{a_1}{\sqrt{E}} \oplus b$$

where the first term corresponds to the electrical noise contribution (*noise term*), the second is due to statistics, e.g. fluctuations in the number of photoelectrons (*sampling term*), and the third contains the contribution due to systematic errors (*constant term*). The latter dominates at high energy and is made up of the cell-to-cell intercalibration error, geometry effects, physics noise, etc.

For a general "quantitative" estimate of the energy resolution deterioration due to the intercalibration error see [VIR91]. Concerning NA48 an analysis has been carried out in [CEC95] using the Saclay GEANT simulation of the LKr prototype calorimeter. The calibration constants of the 12 central cells were normally randomized for several values of σ, up to 1%, and the shower's energy was reconstructed with a 11×11 box for 80 GeV showers, where the largest contribution to the energy resolution comes from the constant term. According to this study a 1% error on the calibration constants induces a 0.4-0.5% contribution to the constant term. No effect is seen up to 0.2-0.3%, and the situation is probably tolerable up to σ =0.5%.

2. Intercalibration Error vs. Statistics

Now that we have an indication of which intercalibration error is tolerable we can ask ourselves how many events per cell are necessary to reduce the statistical error to the desired level (the systematic error is a different matter). This depends obviously on the precision of the quantity(ies) used to calculate the coefficients, such as (see examples in Part II) the pulse height (single cells), or the shower energy (minimization of the energy resolution), or the shower energy and the particle's momentum (*E/p* method), or the invariant mass (invariant mass fit). Quite generally we may state that $\sigma_{calibr} \geq \sigma_{measure}/\sqrt{n}$ where *n* is the number of events used. We have tried to stay conservative, not forgetting that additional factors may spoil the expected precision – such as the sharing of the energy deposited in the hit cell and its neighbours, or possible correlations between the calibration coefficients (of neighbouring cells for example, as suggested in [OGR91]).

3. NA48 LKr Calibration Scheme using Ke3 Events

A Ke3 calibration scheme for the NA48 electromagnetic calorimeter was outlined in [CUN93] and will be reviewed in the following paragraphs. It comes as no surprise that similar ideas show up also in the Fermilab KTeV Technical Proposal (see App. **A1**).

BASICS

Ke3 "background" events can be used as an abundant source of isolated calibration electrons, whose energy measured in the LKr calorimeter can be matched to the momentum from the spectrometer. This would permit a continuous *in situ* calibration, during normal data taking and/or in dedicated runs, without having to displace the calorimeter; it will probably be complementary to the use of calibration beams.

What makes this scheme even more interesting is the fact that (when no magnetic field is applied) photons from $2\pi^0$ decays and electrons from Ke3 have the same angular distribution and similar momentum distribution, and that for momenta up to 25 GeV/c the calorimeter is completely illuminated, even if not necessarily in an uniform way.

STATISTICAL NEEDS

For electrons in the energy band 15 to 25 GeV a (worst case) combined momentum[1] and shower energy error of 1.5% looks achievable. If we assume that an energy sharing effect holds true (see Part II for details), which introduces an additional "degrading" factor equal to the reciprocal of the average fraction of energy deposited in the central cell (around 50%), we can state that 100 events would allow to calculate a cell's coefficients at the 0.3% level, and 400 would be necessary to go down to the 0.15% level. Considering the whole calorimeter we are therefore speaking of several million events, possibly up to 10^7. At this point we have to ask ourselves if it is feasible to accumulate such a statistics in real-time, ideally every few hours, to have a fast feedback and check fluctuations on a short timescale (e.g. day/night effects, if any).

ACCEPTANCE STUDY (KINEMATICS)

It has been suggested to select calibration electrons from Ke3 events on the basis of the following criteria (we assume that an event has already been identified as Ke3 with a clearly tagged electron and pion):

- *z coordinate of the K_L decay vertex*: a somewhat looser cut looks feasible here (compared to the physics event selection), say $Z_{vtx} < 5000$ *cm.*
- *distance between electron and pion (track separation)*: they should be well separated when hitting the calorimeter (no overlap). This may require a rather severe cut, say $D_{e\pi} > 100$ *cm*, because pions can generate large and nonuniform showers.
- *electron energy*: the whole calorimeter should be illuminated by electrons of similar energy (see above). This points to an energy band around 20 GeV, say $15\ GeV < E_e < 25\ GeV$.
- *electron impact angle*: due to the calorimeter's pointing geometry photons from $2\pi^0$ decays hit it almost perpendicularly[2], whereas electrons are bent by the magnetic field and their shower core can be shifted towards a neighbouring cell. A cut such as $\theta_{e/pointing} < 10$ *mrad* was therefore suggested, because this would induce a maximum shift of the shower axis of about 1 cm (half a cell) at the calorimeter's end.

The acceptances for these cuts were calculated using the event-level parallel implementation of NMC v.14 (see App. **A5** and **A6** for more details) with access to the shower library, whose current version does in fact not permit to study the impact angle effect, all particles hitting the LKr front face perpendicularly. Ke3 events were generated over the whole K_L momentum range, i.e. $0\ GeV/c < p < 450\ GeV/c$, and with a Z_{vtx} from the exit of the final K_L collimator (*480 cm*) to the LKr front face (*12109.5 cm* as indicated by N. Doble, CERN).

Results are reported in the following table, with the first two columns referring to events that passed the first level trigger with nominal and reduced magnetic field respectively (p_T kick in MeV/c). The third column corresponds to events that passed also the second level charged trigger (normal data-taking, impossible to lower the magnetic field), with results in fact heavily depending on the decay range and invariant mass window choice. The first two rows contain the electron's average impact angle with respect to the z axis and in the pointing geometry. The following rows give the fraction of events that pass all previously mentioned cuts, checking also the case $\theta_{e/pointing} < 20$ *mrad*, and the corresponding rates per burst normalized to 100 kHz of Ke3 events. A detailed set of results, as well as the L1 and L2 trigger conditions, are reported in App. **A4**.

[1] We will see in the next paragraph that it might be necessary to run with a reduced value of the magnetic field.

[2] It may be better to say that a photon from an accepted $2\pi^0$ event hitting the cell center generates a shower that develops (on average) along the cell axis.

	L1, p_T 257 (R 31)	*L1, p_T 100 (R 32)*	*L2, p_T 257 (R 33)*
$\langle\theta_{e/z}\rangle$	14 ± 11	7.9 ± 6.3	12.0 ± 7.8
$\langle\theta_{e/pointing}\rangle$	12.5 ± 9.3	5.5 ± 4.9	10.5 ± 6.6
% [cuts & $\theta_{e/pointing}$<10 mrad]	0.3%	2.1%	0.03%
% [cuts & $\theta_{e/pointing}$<20 mrad]	2.1%	2.1%	0.21%
# [cuts & $\theta_{e/pointing}$<10 mrad]	700 / burst	5000 / burst	80 / burst
# [cuts & $\theta_{e/pointing}$<20 mrad]	5000 / burst	5000 / burst	500 / burst

The table suggests that the value of the magnetic field is indeed critical when cutting at 10 mrad, so that this limit should be carefully evaluated. Supposing to carry out a *dedicated run* one would record a maximum of about 5000 good calibration electrons per burst, equivalent to 14 10^6 events per day at 50% efficiency (includes SPS, tagging and identification, etc.). The calorimeter's most populated cells (annular region, see Fig. 8 below) would contain roughly *2500 hits/day*, and about a factor of 5 less, i.e. *500 hits/day*, at the edges and close to central hole because of the non-uniform illumination; in this scenario even recalibrating[3] several times per day looks feasible. Figures for the *normal data-taking* activity are available only for a straight implementation of the second level charged trigger (L2, M 257), and suggest that calibrating once per day might be achievable (possibly repopulating depleted cells). More work has obviously to be carried out in this direction.

A set of plots corresponding to a run with reduced magnetic field are shown in Figs. 1-8. Fig. 1 corresponds to the impact (true) energy distribution of all Ke3 electrons passing the first level trigger, and of all selected (calibration) electrons. Fig. 2 contains the corresponding histograms for the impact angle distribution with respect to the z axis and in the pointing geometry (note the cut at 10 mrad, bottom right), and Fig. 3 reports the track separation. The following plots contain all possible correlations between the four quantities (Z_{vtx} ,$D_{e\pi}$, E_e , $\theta_{e/pointing}$) on which a selection is applied, again for all events after L1 and all accepted ones. Fig. 8 contains, finally, a hitmap of the calibration electrons, with each pixel corresponding to a calorimeter cell.

Now that we know that is feasible to accumulate such a statistics in real-time, we can ask ourselves how to carry out the calibration from a computational point of view. We will not address here the issue of how to identify a calibration electron and implement the cuts described above online (some are rather straightforward, others do probably require an interaction between the different triggers).

4. Implementation and Testing of a Parallel Calibration Scheme

EVENT GENERATION

The event generation was carried out, as above, with the event-level parallel version of NMC running on the CERN Meiko CS-2, a typical configuration featuring 16 dual-processor nodes (32 processors in total) and 2 event workers per node. Fiducial cuts, along the lines described in the previous chapter, were introduced on the pure MonteCarlo quantities between the end of the tracking phase and the shower simulation (jzSHOW subroutine). All events passing this stage were then filtered by the first level trigger, as usual. In this way it was possible to generate some 10^6 "accepted" Ke3 events on a timescale comparable to the experimental one, the generation of one good calibration electron requiring around 100÷150 msec. No event reconstruction was attempted; only one shower library file was used instead of all six usually mounted on a Parallel File System[4] (PFS), the latter being temporarily unavailable to avoid clashes with the NA48 Central Data Recording.

[3] We mean the calculation of the calibration coefficients, not the determination of E(x,y) (impact point dependence of the total energy).

[4] A parallel data storage scheme, whereby a file is physically distributed (striped) among several disks, guaranteeing higher data rates, in a transparent way for the user.

CALIBRATION ALGORITHM (MINIMIZATION OF THE ENERGY RESOLUTION):

We decided to calculate the calibration coefficients by minimizing the calorimeter's energy resolution, i.e. by comparing for each calibration electron (i-th event) the energy deposited in the calorimeter with a "reference" value, E_{ref}. The latter should be given by the tracking measurement, but we used directly the electron's true (MonteCarlo) energy to simplify things. The quantity to be minimized can therefore be written in the following way (more details are given in Part II):

$$\chi^2 = \sum_{events} \left(E_{ref}^{(i)} - \sum_{cells} a_k E_k^{(i)} \right)^2$$

where the k-th cell has calibration coefficient a_k and collects the energy E_k (we started directly from the cell energies as retrieved by the shower library). Note that in this initial implementation no error term $\sigma^{(i)}$ was considered, and no corrections on the energy were applied (e.g. for its dependence on the impact point).

Setting the derivative of χ^2 with respect to all M parameters a_k to 0 leads to a system of M linear equations, the so-called *Normal Equations* of the linear least-squares problem:

$$\frac{\partial \chi^2}{\partial a_k} = 0 \quad \forall k \quad \Rightarrow \sum_{cells} \alpha_{kj} a_j = \beta_k$$

where **[α]** is a square matrix of rank M and **[β],** the Right Hand Side (RHS) of the system of equations, is a vector of length M:

$$\alpha_{kj} = \sum_{events} E_k^{(i)} E_j^{(i)} \quad \text{and} \quad \beta_k = \sum_{events} E_{ref}^{(i)} E_k^{(i)} \quad .$$

In other words we have to deal with a system containing as many linear equations as cells in the calorimeter. Its solution in a *direct* way demands of the order of M^3 floating point operations.

CALIBRATION ALGORITHM IMPLEMENTATION

For the implementation of the calibration algorithm itself we decided to split the calorimeter into $N_{sectors}$ identical and non-overlapping *sectors*, which somewhat eased coding and problem handling; each of them was assigned to an *ad hoc* process, or *calibration worker*, that calculates the calibration coefficients of the corresponding cells. $N_{sectors}$ was fixed to 16, which translates into square sectors of 32 × 32 cells. We acknowledge that the calorimeter's subdivision in physical space has also been implemented by the CLEO collaboration (see the corresponding example in Part II); the handling of the sectors in parallel is probably unique to our approach, though.

The total number of processes, N_{procs}, and nodes, N_{nodes}, is set by the user when starting the code (with the *prun* command[5]), and the operating system assigns to each process a unique *process-id*, ranging from 0 to N_{procs}*-1*. In fact all processes are copies of one and the same executable, which decides at run time how to "behave" by checking it own process-id and acting accordingly in a predefined way. In particular it was decided that process 0 is the master, that the last $N_{sectors}$ processes, N_{procs}-$N_{sectors}$ to N_{procs}*-1*, are the calibration workers, and that all others are the event workers, ideally two per node (i.e. $2N_{nodes}$ in total). In case of a typical 16 nodes run this gives N_{procs} =1+16+2·16= 49 processes overall, three per dual-processor node. The processes themselves are distributed in a "round-robin" fashion among all nodes, i.e. process 0 to the first node, 1 to the second, ..., 15 to the 16th, 16 again to the first and so on, which provides an elementary form of load balancing. Note that we are implicitly using the possibility of loading multiple copies of the same executable on each node.

[5] This command, as well as most of the following figures, are CS-2 specific. The underlying concepts, however, are of general applicability to most parallel programming environments.

A messaging scheme was set up between event and calibration workers, whereby the event workers identify "good" Ke3 events, isolate the shower corresponding to the calibration electron using a 7×7 shower box around the hit cell, and send the cell energies to the calibration worker whose sector contains the hit cell. The latter deal with 32×32 cells each, as already mentioned, for a total of 1024 calibration coefficients; the full matrix of the corresponding linear least-squares problem, as encountered above, contains therefore 1024×1024 double precision entries (8 MBytes of data). A calibration worker calculates for each shower the crossproduct of the 49 cell energies, about 2500 multiplications if the symmetry is not exploited, and updates the matrix elements as well as the Right Hand Side of the system of linear (normal) equations. The system itself is then solved at the end of the run, and we note that during the run the calibration workers are far from being equally loaded because of the non-uniform calorimeter illumination.

SPARSE STORAGE

There is the possibility of storing the coefficient matrix in a sparse way (kind of "zero suppressed") because each shower has only a limited size (shower box), which means that each cell is only correlated to a given number of neighbouring ones. One can rely on sparse storage to use matrix solvers designed to deal with sparse systems, or to minimize storage requirements while calculating the matrix coefficients at each event. In the latter case sparse storage is used up to the solving stage, where a standard full matrix of N_{cells}^2 entries is filled in. A typical sparse storage scheme may consist of three vectors, the first one containing all matrix entries different from zero (DOUBLE PRECISION), the second and third containing the corresponding row and column indices (INTEGER). In our case we consider shower boxes of fixed size, which means that the problem itself – and therefore its storage – has quite a regular structure; in fact for each cell we know a priori which crossproducts with the others are going to be non null. In other words, one can define the structure of the vector at initialization (mapping of its entries to the cell energies crossproducts), and access it directly at each event without having to scan the indices vectors; it is rather easy to find the position in this vector of the matrix entry α_{kj}. There is clearly some space wasted for the cells close to the edges which have fewer neighbours, but this is the price to pay for a regular structure. The corresponding entries can in any case be removed before passing the vector to the solver.

Sparse storage is advantageous for large matrices and/or small shower boxes. In fact the degree of sparsity of a matrix, i.e. the inverse of the fraction of non zero matrix elements, changes because the shower box is fixed and gets better for larger matrices. To estimate the required storage space we note that for a given cell (not necessarily the one hit!) one has to calculate the crossproducts with all other cells "within reach", i.e. all cells that can share a shower box with the given one (we will call them *calibration cells*). Graphically one has to superpose the 4 shower boxes that have the given cell as its lower right, lower left, upper left and upper right corner cell respectively (clockwise). The total number of these calibration cells and the corresponding degree of sparsity are

$$N_{calibrcells} = (2\,ShowerBoxSize - 1)^2 \quad \text{and} \quad Sparsity = \frac{N_{cells}^2}{N_{cells} N_{calibrcells}} = \frac{N_{cells}}{N_{calibrcells}},$$

whereas the (**minimum**) amount of memory required to store a matrix in sparse format is equal to $8 N_{cells} N_{calibrcells}$ MBytes (double precision format), as represented in the following table for a number of interesting cases (storage in MBytes).

configuration	*Sector Size*	N_{cells}	*ShowBox Size*	$N_{calibrcells}$	*Sparsity*	*Sparse Storage*	*Full Storage*
sector	32	1024	7	169	6.1	1.3	8.0
sector with overlap	32+7+7	2116	7	169	12.5	2.7	34.2
full calorimeter	128	13500	7	169	79.9	17.4	1390
sector	32	1024	11	441	2.3	3.4	8.0
sector with overlap	32+11+11	2916	11	441	6.6	9.8	64.9
full calorimeter	128	13500	11	441	30.6	45.4	1390

The actual storage requirements may differ from the one reported above because sparse matrix solvers require additional vectors containing the indices of the non zero elements (as already pointed out), and possibly some extra space, the so-called "elbow room", which has to be determined on a case by case basis and can be relevant (factor 2 to 4). On the other hand this additional space is not used up to the solving stage, and by exploiting the inherent symmetry of the coefficient matrix one could save a factor two. We also note in passing that the sparse storage of the coefficient matrix for the full calorimeter requires less than 20 MBytes when using 7×7 shower boxes, which means that each event worker could keep a full matrix in memory with no messaging till the end of the run...

SOLVERS AND FIRST RESULTS

In order to solve the normal equations the following software was tried out (the input matrix is usually destroyed):

- the subroutine *DSEQN* (CERNLIB entry F102), which works on the full coefficient matrix stored in double precision. It implements a *LU* factorization via a (modified) Cholesky Decomposition (see Part II) and works therefore only for symmetric positive definite matrices. In fact the CERNLIB solver relies on the symmetry of the input matrix and it is sufficient to fill the elements α_{kj} of the lower triangle ($k \geq j$). The use of this subroutine is straightforward.
- the *MA28* subroutine interfaced to the Harwell sparse matrix package, as described in more detail in the Example on CLEO [see the references in Part II]. It is also based on the *LU* decomposition and is of general use, which means that it is not necessarily optimized for the solution of the normal equations. On the other hand it is quite refined, being able for example to proceed even in case of structurally singular matrices. Non zero entries of the coefficient matrix can apparently come in any order, and seem to be arranged by columns internally, which should be a fast operation. We note in passing that a new version (2) of the Harwell package has since been released and the MA28 code superseded [see the list of WWW links].

In fact both solvers are not able, with the initial setup, to give results for any sector because they have to handle structurally singular matrices, due to the masking of the octagonal shape and the beam pipe (there are obviously no signals in the corresponding cells). The reason behind this is the simple cell numbering scheme employed whereby all cells in the 128×128 matrix containing the calorimeter are considered; a more realistic mapping that assigns a number only to the physically existent cells would clearly solve this problem. A quick fix was anyhow found by removing the masking of the LKr edges and hole in the showering subroutine, after it has been determined if a particle has hit the calorimeter or not. In this case both solvers were able to converge in the four central sectors around the beam pipe, and gave nearly identical results. The Harwell code is furthermore capable of dealing with all other sectors by toggling the "structurally singular" flag (see above), and delivers meaningful calibration coefficients at some distance from the edges.

Some results for one of the four central sectors are shown in Figs. 9 to 13; they refer to a run of about 50 10^6 generated and 2.8 10^6 accepted Ke3 events carried out in 5 hours. The solving time was of the order of a minute per sector, with the Harwell package about a factor 1.5 to 3 (extreme case) slower than the DSEQN subroutine in the sectors where both converged. We recall that no correction to the shower library cell energies was attempted, and that the results should therefore be considered from a qualitative more than a quantitative point of view (proof of feasibility). The solving times is consistent with a total of about $1024^3 \approx 10^9$ floating point operations, executed at roughly 10 Mflops (floating point operations per second).

The upper part of Fig. 9 contains the sector's hitmap in absolute coordinates and shows its placement within the full calorimeter, whereas the lower part gives an idea of the nonuniformity in the cell population. The "ringing" effect close to the edges, described also by CLEO (Part II), is clearly visible in Fig. 10, which shows a sector's slice (rows 26 to 32 and columns 4 to 29); the lower half is similar, having simply left off the last row. The oscillations seem to extend as far as one full shower box (7×7) away from the edges. Fig. 11 shows a map of the calibration coefficients in the centre of the sector, where each (colour) bar corresponds to a range of about 0.2%. The average is around 1.149-1.150 and should be equal to the ratio of the electron true energy vs. the total energy deposited in a 7×7 shower box. We note in passing that cells at the top of the scale seem to have often a neighbour at the bottom of the scale (correlation). A cell by cell plot of all 1024 coefficients, displayed one row after the other, is

shown at the top of Fig. 12, whereas at the bottom two rows have been extracted for a total of 64 coefficients. The large spikes correspond to cells close to the hole (or inside), the small ones to the oscillations near the edges. The first and last row, i.e. the first and last 32 coefficients, are entirely composed of "boundary" cells and give rise to the small plateaus at the leftmost and rightmost margin of Fig. 12 (top). The distribution of the coefficients is represented in Fig. 13, where the "good" ones are grouped in the central peak around 1.15 (from 1.14 to 1.16); all others are byproducts of the edge effects and can be interpreted as previously described.

SOME REMARKS AND OPEN ISSUES

Any final solution along this scheme (splitup in physical space) will have to deal with the edge effects, either by introducing appropriate MonteCarlo corrections to the deposited energy, or by employing overlapping sectors as also chosen by CLEO. In the latter case one may have to extend a sector by as much as a full shower box, or even more to have some safety margin, without forgetting that storage requirements go as N_{cells}^2 for the full matrix[6] and solving time as N_{cells}^3 roughly. Before doing this one should in fact understand if it is necessary to select for the calibration exactly the same (amount of) cells as for the reconstruction (as first pointed out by P. Calafiura, CERN); if this turns out not to be the case then one could use even somewhat smaller shower boxes, at the penalty of a higher statistics needed to achieve a given precision level. And, as already noted above, a more efficient cell numbering scheme will also have to be adopted.

Concerning the calibration workers it seems that their presence introduces quite some overhead, in the worst case consuming as much CPU as each of the two event workers sitting on the same node. The origin of this effect is unclear and may reside in the messaging mechanism, even if this is not supposed to happen according to the following estimate: a good calibration electron is generated by each event worker every 100 (to 150) msec, which implies a total rate of 300 Hz running with 32 processors. Even if all hits were concentrated in the four central sectors each of them would receive less than 100 showers per second, with such a short message (49 cell energies plus a few extra words) taking well under a 1 msec. The computation of the 2500 odd cell energy crossproducts and the corresponding accesses to memory should also contribute in a negligible way to the total CPU time, and this should not be too difficult to verify (just commenting one or two subroutine calls). As a final remark we note that one might start buffering messages to a given calibration worker, sending 10 showers at a time say, to reduce the messaging overhead if needed.

5. Conclusions

An initial parallel version of a technique suitable for the online calibration of the NA48 LKr calorimeter has been implemented on the CERN Meiko CS-2 platform, using as a starting point the event-level parallel implementation of NMC. Calibration coefficients were calculated in quasi real-time (some minutes) after having collected calibration events on a timescale comparable to the expected experimental one (several hours to "overnight"); many improvements are nevertheless still needed before being able to claim that this is "the way" to go. In the end several different techniques will be probably used (offline checks), possibly complementing the Ke3 measurements with those from suitable calibration beams.

In fact it is our feeling that even iterative methods might be used online on a machine such as the CS-2, provided one can somewhat reduce the event size and exploit the platform's parallel I/O capabilities. Let's consider for example 10^6 reconstructed Ke3 events, each containing a calibration electron, and suppose to be able to save for each of them only the information strictly needed to calibrate, such as the energies in each shower cell (having already reconstructed the pulses). A couple of KBytes per event would give a total of 2 GBytes, which can be readily stored on a Parallel File System and should be readable at (at least) 20 MBytes/sec, that is in about 100 seconds...

[6] But only linearly in N_{cells} in the case of sparse storage!

Part II:

Review of some Calibration Methods for Electromagnetic Calorimeters

In this part we will review some "standard" calibration techniques for electromagnetic calorimeters together with practical examples whenever possible, focusing on devices similar to the NA48 one (large number of channels, delivering a 2D information). Although some of these schemes can actually be used to obtain an absolute calibration we will consider in general only on the relative one[7].

One type of approach is based on beams of (high energy) particles, typically electrons or muons of a well known momentum, that scan the calorimeter's surface. Sometimes this is not practical or even impossible (e.g. at colliders) and one has to rely directly on physics processes such as (radiative) Bhabha events ($e^+e^- \rightarrow e^+e^-$) at CLEO and at LEP, or events with isolated electrons, for example from Z and W decays, at the Tevatron (and planned for LHC). In the NA48 case one might envisage to adopt both approaches, using electrons from Ke3 events as a continuos source of calibration together with dedicated Ke3 runs and calibration beams. What should not be forgotten is that the quality of the results is also dependent on the energy range being calibrated, compared to the final range of energies to be covered in the experiment; in fact extrapolating to different energies, or even to different particles such as from electrons to photons, is far from trivial (nonlinearities!).

1.1 Single Cell Methods

These methods treat each cell as a single, separated entity; the results obtained should therefore not depend on the behaviour of the neighbouring cells, which is particularly important whenever there are dead or malfunctioning channels. It might also be of advantage for cells closer to edges that have fewer neighbours. On the other hand these techniques do usually require larger statistics, partly because a single measure is less precise and partly because the deposited energy is often strongly dependent on the impact point, making one reluctant to accept events too far from the cell centre. Among the most important single cell methods we have:

PULSE HEIGHT DISTRIBUTION (MUONS)

In the simplest cases plotting the pulse height distribution of each cell and looking for the peak position, by fitting a polynomial for example, or calculating the mean value, might be sufficient to obtain a relative calibration estimate[8] (see for example [BLU86]).

In the particular case of muons, possibly from cosmic rays, or non interacting pions, energy is predominantly lost via ionization – whereas electrons induce showers – and one has to deal with a (modified) Landau distribution. Apart from the usual problem of accuracy (statistics) it is necessary to extrapolate the calibration coefficients to electrons having usually a much higher energy. The noise level is also very important, given the small energy release; these particles may nevertheless be useful to get initial sets of calibration constants (see also [BRA90], [NEW92] and Ch. 2.7 in [BAB93]).

The L3 experiment exploited its good muon tracking system by taking a couple of dedicated runs at LEP in 1991, looking for cosmics crossing opposite faces of a crystal while releasing about 20-30 MeV[9]. They collected several million triggers, which translates into some hundred muons per crystal (one muon traverses in fact many crystals). It was demonstrated that a 1.3% precision level could be attained in one month (equivalent time for a dedicated run). Limitations came from errors on the track length, pedestals and Landau fluctuations (see [BAK88], [BAK89-1], [BAK89-2], [BOR89], [ADE90], [BGO93], [KAR95]). It should be noted that this is only way of obtaining uniformity profiles along the crystal's major axis, i.e. measuring the light collection efficiency.

---

[7] The absolute (neutral) energy scale will in fact be fixed in NA48 using the position of the AKS counter...

[8] Some comments on the peak vs. mean choice are given, albeit in a different context, in the L3 example of the ITERATIVE PROCEDURES Section.

[9] A similar figure, around 30 MeV, is also reported in [KES95] concerning the KTeV CsI prototype calorimeter.

M. Martini (NA48, CERN) analyzed some muon data taken with the NA48 LKr prototype in 1992 and reached similar conclusions on the difficulty in fitting the resulting distribution (Vavilov?), which has a reported width of 10%, and the need for high statistics. This may not rule out the possibility of taking dedicated muon runs...

FIT OF THE ENERGY DISTRIBUTION

A more refined technique consists in making a fit of the energy distribution as a function of the impact point using an appropriate parametrization, deduced for example from a MonteCarlo simulation. This was one of the schemes employed by the L3 experiment for the (pre)calibration of their BGO electromagnetic calorimeter at the SPS using electron beams (App. **A3**). They divided each crystal in squares of 1 mm^2 and calculated the average value of the signal, V(x,y), in each of these ([KAA89], [SOU90]). These data points were then fitted with the following function (parabola):

$$V(x,y) = V_0 - \beta\left((x-x_0)^2 + (y-y_0)^2\right) = V_0 - \beta R^2$$

where V_0 is the maximum of the signal (i.e. in the crystal centre, of coordinates (x_0, y_0)) and is used for the (inter)calibration. To get an absolute value one has to apply the same procedure to MonteCarlo events, obtaining an E_0 corresponding to the energy deposited in the centre of the crystal. The cell calibration coefficient is then: $C = E_0/V_0$. As indicated above, this method needs relevant statistics in the central part of the crystal.

Within NA48 a preliminary study has been carried out by J. Duclos (NA48) using the following function to parametrize the energy deposition in the hit cell:

$$\frac{E_{cell1}}{E_{total}} = K\left(1 - 0.31\left[\sqrt{(x-x_0)^2 + (y-y_0)^2}\right]^3\right) = K\left(1 - 0.31R^3\right) \qquad K \approx 0.6$$

The parameter K plays the same role as V_0 in the previous equation, and the resolution on K is of about 5.6% (may be improved). A cut at R=0.8 cm is suggested.

OTHER METHODS

In order to overcome the lack of calibration beams at colliders an interesting scheme based on a Radio Frequency Quadrupole (RFQ) accelerator has been devised and tested at L3, resulting in an accuracy of about 1% for 900K triggers collected in a few hours ([ZHU90], [KAR95]). A 1.85 MeV H beam is focused on a Li target, generating a great number of 17.6 MeV photons, visible as a “bump” (edge) in the energy spectrum at around 15-20 MeV. One has to deal with the usual problem of extrapolating the results to higher energies (nonlinearities). An alternative method that uses a fluoride target to generate on a short timescale (few minutes) a very large number of photons per beam pulse, “equivalent” to one high energy photon, is described in [ZHU90] and [NEW92].

For the ATLAS LAr calorimeter the possibility of controlling the cell gap uniformity and temperature gradient by measuring the drift velocity of each cell via a fit of the signal shape is also under study (L. Serin, Orsay).

1.2 Multiple Cells Methods

These methods consider each cell as part of a larger entity (shower), in the most simple case a square box centred around the hit cell. Correlations can therefore be introduced between neighbouring cells and interesting situations arise close to the edges[10], making it necessary to rely on appropriate corrections. On the other hand a single measure can be very accurate, reducing the overall statistics needed, and is less dependent on the impact point. Among the most important methods involving multiple cells we have:

MINIMIZATION OF THE ENERGY RESOLUTION:

Assuming a linear dependence of the total energy on the calibration coefficients (model depending linearly on its parameters) we can write the quantity to be minimized in our case as follows, using the notation of Ch. 15 in [PRE92]:

$$\chi^2 = \sum_{events} \left(\frac{E_{ref}^{(i)} - \sum_{cells} a_k S_k^{(i)}}{\sigma^{(i)}} \right)^2 = |\mathbf{A}\cdot\mathbf{a} - \mathbf{b}|^2 \quad \text{with} \quad A_{ij} = \frac{S_j^{(i)}}{\sigma^{(i)}} \quad \text{and} \quad b_i = \frac{E_{ref}^{(i)}}{\sigma^{(i)}}$$

where the k-th cell has calibration coefficient a_k and collects the signal S_k. The energy deposited in the calorimeter in the i-th event is compared with a "reference" value, E_{ref}, which might stay nearly unchanged from event to event (e.g. for calibration beams), or fluctuate within a large range (e.g. for physics processes such as Ke3 events). In the former case the measured error (standard deviation) on the i-th data point, $\sigma^{(i)}$, is usually not considered. In the latter case one often speaks of the *E/p method*, with *p* typically coming from the tracking measurement in a magnetic field, and $E=E_{calor}$ being the corresponding energy as measured in the calorimeter (see for example Ch. 8 in [AMO93]), and may decide to minimize the quantity $(E/p-1)^2/\sigma^2$. In the following we shall use indifferently the term *Energy Resolution Minimization* or *E/p Method*, which is probably correct provided that in each case the most appropriate error term $\sigma^{(i)}$ is used[11].

[NOTE: a commonly referenced *E/p* article is [ABE91], on the W mass measurement by CDF, which contains a paragraph on the relative and absolute electron calibration of their central calorimeter. The latter has in fact a rather coarse granularity and their *E/p* approach seems somewhat computationally simpler (relative calibration via gain equalization in individual calorimeter cells)]

Appropriate corrections on the energy have usually to be applied, taking into account its dependence on the impact point, leakage effects, etc., in order to achieve high precision results[12].

The matrix **A** is called the *design matrix* of the fitting problem (linear least squares problem) and has size $N \times M$, where N is the number of events (data points) and M the number of calibration coefficients (a_1 to a_M), one per cell. In fact we have implicitly assumed that a cell numbering scheme has already been defined, assigning to each cell of coordinates (i,j) (assuming a 2D configuration in physical space) a unique number $k=F_{index}(i,j)$, for example simply indexing by rows. The design matrix is schematically as follows:

[10] Remember that the cell energies, and therefore indirectly the calibration coefficients, are also used for the impact point determination....

[11] Otherwise two events with the same relative error (E/p-1) might have a different absolute error (E-p), and the two approaches could produce different results (Example: event 1 with p=10, E=9; event 2 with p=20 and E=18).

[12] It is not clear if σ depends also on the number of events per cell, i.e. if one has to take into account a non-uniform event distribution, as hinted to for example in [CEC95].

$$\begin{pmatrix} \frac{S_1^{(1)}}{\sigma^{(1)}} & \frac{S_2^{(1)}}{\sigma^{(1)}} & \cdots & \frac{S_M^{(1)}}{\sigma^{(1)}} \\ \frac{S_1^{(2)}}{\sigma^{(2)}} & \frac{S_2^{(2)}}{\sigma^{(2)}} & \cdots & \frac{S_M^{(2)}}{\sigma^{(2)}} \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ \frac{S_1^{(N)}}{\sigma^{(N)}} & \frac{S_2^{(N)}}{\sigma^{(N)}} & \cdots & \frac{S_M^{(N)}}{\sigma^{(N)}} \end{pmatrix}$$

Several different techniques may be used to find this minimum. As far as we understood they all share the advantage that it is not necessary to iterate over the data[13] (single pass solution). We will discuss two of them:

- MINIMIZATION VIA SINGULAR VALUE DECOMPOSITION: A solution which is the best approximation in a least-squares sense can be found via the *Singular Value Decomposition* (SVD) technique, working directly on the design matrix **A**. SVD is in theory very stable and cannot fail, and is in fact the recommended way to do it. On the other hand this technique can turn out to be significantly slower than solving the normal equations (see below), and seems to require an extra array of size $N \times M$ to store the whole design matrix (!), which looked prohibitive for the NA48 calibration exercise.

- MINIMIZATION VIA THE NORMAL EQUATIONS: Setting the derivative of χ^2 with respect to all M parameters a_k to 0 leads to a system of M linear equations, the so-called *Normal Equations* of the linear least-squares problem:

$$\frac{\partial \chi^2}{\partial a_k} = 0 \quad \forall k \quad \Rightarrow \sum_{cells} \alpha_{kj} a_j = \beta_k$$

where **[α]** is a square matrix of rank M

$$\alpha_{kj} = \sum_{events} \frac{S_k^{(i)} S_j^{(i)}}{\left(\sigma^{(i)}\right)^2} \quad \text{or} \quad [\alpha] = \mathbf{A}^T \cdot \mathbf{A} ,$$

and **[β],** the Right Hand Side (RHS) of the system of equations, is a vector of length M

$$\beta_k = \sum_{events} \frac{E_{ref}^{(i)} S_k^{(i)}}{\left(\sigma^{(i)}\right)^2} \quad \text{or} \quad [\beta] = \mathbf{A}^T \cdot \mathbf{b} .$$

In matrix form one would write the normal equations as follows[14]:

$$[\alpha] \cdot \mathbf{a} = [\beta] \quad \text{or} \quad \left(\mathbf{A}^T \cdot \mathbf{A}\right) \cdot \mathbf{a} = \mathbf{A}^T \cdot \mathbf{b} .$$

It is interesting to note that a cell pair of coordinates (in physical space) (i_1,j_1) and (i_2,j_2), having respectively indices $k1=F_{index}(i_1,j_1)$ and $k2=F_{index}(i_2,j_2)$, gets mapped onto the matrix entry α_{k1k2}. On the other hand it is not difficult to see, considering for example two cells separated by a couple of rows belonging to one and the same shower, that neighbouring cells in physical space might nevertheless be numerically "distant", i.e. have a large value of $|k1-k2|$, thus getting mapped to an

[13] At least if one doesn't start too far off from the real constants... see again [CEC95].
[14] See [BIN86] and Ch. 2.3 in [BAB93] for a modified version which uses mean amplitudes.

entry far away from the diagonal of the matrix **[α]**. The cell numbering scheme can therefore have far-reaching consequences on the structure of **[α]** and the corresponding solving strategy/results.

Note that the errors on the parameters are given by:

$$\sigma^2(a_j) = C_{jj} = [\alpha]^{-1}_{jj}$$

i.e. they are the diagonal elements of the *covariance matrix*. More in general, the off-diagonal elements C_{jk} are the covariances between the parameters a_j and a_k, where one has to remember that the formal covariance matrix has a clear quantitative interpretation only if (or to the extent that) the measured errors are normally distributed. We also note in passing that the solution of the least-squares problem directly from the normal equations may be susceptible to accumulated roundoff errors, which is the more likely to happen the closer a set of equations is to singular – think at the presence of dead or malfunctioning channels for example. Direct substitution back into the original equations helps to detect this type of errors, and the solution might be iteratively improved (one pass is usually enough). This procedure requires some extra storage but only $O(M^2)$ floating point operations (FLOPs[15]).

The system can be solved (Ch. 2 in [PRE92]):

⇒ in an *iterative* way – attempting to converge to the desired answer in however many steps are necessary. An example is the *Conjugate Gradient Method*, which refers to the matrix only through its multiplication of a vector, or the multiplication of its transpose and a vector, and is attractive for large sparse systems. It is guaranteed to converge in at most M steps (sic!) and works by generating successions of search directions and improved minimizers (estimates of the solution).

⇒ in a *direct* way – requiring a predictable number of operations. We may remark that, unless the errors on the parameters are required (see above), it is in fact not necessary to invert the matrix **[α]** to solve the system. This is for example the case of the *LU decomposition*, which factorizes a matrix into the product of a lower (**L**) and an upper triangular (**U**) one, relying on the fact that the solution of a triangular set of equations is quite easy. The vector **a** is then obtained via a procedure called *backsubstitution*. Overall about $2/3M^3$ floating point additions and multiplications are required, or $2M^3$ FLOPs if the matrix inversion is desired. The *LU* technique is in fact quite general and does not use the properties of **[α]** = $\mathbf{A}^T \cdot \mathbf{A}$, which is symmetric, and positive definite if **A** is non singular. A special triangular decomposition, the *Cholesky decomposition*, exists and is about a factor 2 faster than *LU* decomposition. It is very stable numerically, and a failure usually indicates that the matrix is not positive definite.

As an example of a large scale problem solved in a *direct* way we will illustrate the basics of the (offline) Bhabha calibration of the *CLEO* CsI electromagnetic calorimeter (see also App. **A3**); a smaller scale implementation was probably carried out by Crystal Barrel, as hinted at in [BRA90]. The CLEO group constrained the shower energy from a Bhabha electron to the beam energy and minimized the corresponding energy resolution, having to deal with a matrix of rank 7800 characterized by a high degree of sparsity, given that their showers are rather small[16] and that the product $S_k S_j$ is non-zero only if crystal k is physically close to crystal j ([OGR90], [NOR91], [OGR92], [KUB92]). To solve the corresponding system of linear equations they used the Harwell MA28 sparse matrix package[17], and adopted a special indexing scheme ("potato-peel") to make a crystal as numerically close to its neighbours as possible. This ensures that the corresponding matrix is approximately diagonal.

[15] Total, not per second!
[16] The shower energy is computed from a cluster's most energetic N crystals (N varies logarithmically up to 17).
[17] C. O'Grady, B. Heltsley, *private communication*; [OGR91]; see also the list of WWW links.

The solving package was tested injecting into MonteCarlo events "fake" calibration constants following a Gaussian distribution with mean 1.0 and $\sigma = 0.14$; then they checked that the correct solution had indeed been found by taking the calculated solution, multiplying it by the matrix and comparing it with the original right hand side. Following this a plot of the ratio of the injected to the calculated constants was made, controlling that its RMS was small (would obviously have been 0 if the constants had been calculated perfectly). Finally, it was verified that the calibration process did indeed narrow the energy distribution.

Moving to 5 GeV electrons introduced a higher number of off-diagonal elements (compared to the 1 GeV initially tried out), and the matrix solver turned out to be unable to find a full solution even after 2 hours of VAX 8600 CPU time[18]. The detector was therefore split up in pieces and the behaviour at the edges of these regions investigated, expecting that at some distance inside the edge the calculated constants would become the "actual" constants. To quantify this statement they solved a large matrix (3584×3584) and a smaller one (1792×1792) located physically inside the first. The difference between the constants obtained in this way was plotted in the region corresponding to the smaller matrix; they agreed indeed to within 0.1% at a distance of 3 (z-)rows from the edge (which obviously depends on the size of their showers). In fact, the constants show an oscillatory ("ringing") behaviour near the edges due to the partial absence of neighbouring cells, which forces the constants of the last row to be high in order to bring the shower energies up. This in turn drives the constants of the second last row down, and so on.

In the end the calorimeter was split into 3 regions of 3000 to 3500 cells each with an overlap of 10 (z-)rows, thus ensuring that every crystal was calibrated at least 5 rows away from the edge of a matrix. Comparing different sets of calibration constants obtained with real data an accuracy of about 0.6% was estimated (statistics driven, about 30 events per crystal), sufficient for their purposes.

ITERATIVE PROCEDURES

These methods do usually start from an initial set of calibration constants, run through all the data and at the end produce improved constants that are used for the following iteration, which loops again on all events (therefore they are more suited to offline processing). The procedure is repeated till the desired level of accuracy is reached, where the number of iterations might well be influenced by the goodness of the initial constants. Deviations of several % (say up to 10%?), should not constitute a problem, in the sense that the method is still able to converge to the correct results (provided one has a reasonable number of events per cell...).

As an example we will describe the iterative procedures employed for the (pre)calibration of the L3 BGO calorimeter at the SPS in alternative to the single cell method previously described ([SOU90], [BGO95], [KAR95], and App. **A3**). Initial values were taken from cosmic ray data, and for the calibration only events from the central region of each crystal were used, thus analyzing in total about 1000-1500 events per crystal (with this statistics one has indeed to understand the systematics very well!). Corrections to the energy in a 3×3 shower box, due to rear and lateral leakage as well as the impact point position, were taken into account via a GEANT-based MonteCarlo simulation.

One approach, described only in [SOU90], consisted in minimizing for each crystal the following quantity by setting $\partial\chi^2 / \partial a_1 = 0$ (keeping fixed the calibration coefficients a_2 to a_9 of the crystals around the hit one):

$$\chi^2 = \sum_{events} \left(E_9^{MC}(x,y) - \sum_{k=2}^{9} a_k V_k - a_1 V_1 \right)^2$$

Since it is unclear whether this method went beyond a MonteCarlo study we will not go into details.

[18] This depends probably also on the memory requirements, apart from the CPU performance.

The second approach consisted in calculating, on an event by event basis, the calibration constant of the hit crystal assuming again the others fixed:

$$a_1 = \frac{E_9^{MC}(x,y) - \sum_{k=2}^{9} a_k V_k}{V_1}$$

where V_1 is the measured signal in mV seen on the central crystal, V_k is the measured signal in one of the 8 crystals surrounding the central one, a_1 is the calibration constant for the central crystal, $E_9^{MC}(x,y)$ is the expected average energy deposited in the nine crystals. Having processed all events one can determine each crystal's calibration constant either by taking the *mean* value or the *peak* value (using a Chebyshev polynomial parametrization for example) of the a_1 distribution. The mean value is the most precise estimator as it makes use of all statistics available, but it could be sensitive to the tails of the distribution, and therefore to the cuts applied. The peak is free from the effect of these cuts but its determination depends on a reduced sample of events and its statistical accuracy is correspondingly reduced; it may also depend on the choice of the analytical representation of the measured distribution. Results obtained with the two methods (peak vs. mean) agreed within 0.2%.

A fluctuation in the calibration constants' values of less than 4 10^{-4} (!?) was registered after 2 to 3 iterations, which may be due to the goodness of the initial values and to the fact that the fraction of energy deposited in the central crystal is rather large, up to about 75% of the total. The global systematic uncertainty on the calibration constants has been estimated at 0.62%, with contributions from the beam momentum (0.2%), the event selection (0.1%), the temperature corrections (0.5%), and (the choice of) the method (0.3%). The last value comes from a comparison of the results obtained taking the peak value with the ones from the single crystal method[19]. Note that the *in situ* (at LEP) calibration of L3 is summarized in App. **A3**.

Other iterative techniques *(sliding windows)* might consider just a portion of the calorimeter, calculating the calibration coefficients inside it assuming the others fixed. At each iteration the window is moved to a neighbouring (overlapping?) area, and the procedure is repeated until the desired level of accuracy is reached (L. Serin, Orsay, *private communication*).

INVARIANT MASS FIT

Invariant mass fits can also be used to determine calibration constants, especially when no calibration beams are available, and seem for example to have allowed the most accurate calibration of Crystal Barrel ([BRA90]; see also Ch. 2.9 in [BAB93] and [NEW92]). The ATLAS experiment has carried out an interesting preliminary study on the Z^0 -> e^+e^- channel ([AMO92] and [AMO93]). They point out that this technique has the advantage of measuring the calorimeter's performance (characteristics) directly for processes of the same nature as the ones for which it is intended. One of the problems might be in the statistics itself, i.e. the number of available events on a short timescale. In the NA48 case a suitable channel could be K^0 -> $\pi^+\pi^-\pi^0$ -> $\pi^+\pi^-\gamma\gamma$.

In order to constrain the invariant mass spectrum to be around the known value of the Z^0 mass one can use a *least squares fit*, but a more rigorous approach, consisting in applying the *maximum likelihood method* (assuming that the shape of the probability distribution is known), was chosen. This leads, even in the simplest case, to a set of cubic equations in $\sqrt{a_k}$, which are then expanded for simplicity in Taylor series around the value 1. The resulting system of linear equations is of order N_{cells} and can be solved, for example, as discussed above for the energy minimization technique. The set of calibration coefficients thus obtained are still far from the correct ones, but they can be used to redefine the signal values (S_k -> a_k S_k), leading to an improved solution of the second iteration since the expansion is now performed nearer to a_k=1. The final value of the calibration constant in each cell is then given by the product of the corresponding results for each iteration. One obtains for the resolution on the calibration constants, under the main assumption that a "significative part" of the energy is deposited in the central cell:

[19] In fact the values given in the three references above differ slightly.

$$\sigma_k \approx \frac{2R\sigma_m / m_Z}{\left\langle g_1^1 \right\rangle \sqrt{n}}$$

where σ_m / m_Z is the width of the Z^0 peak (natural peak convoluted with detector resolution), n the number of events used, $\left\langle g_1^1 \right\rangle$ the average fraction of energy deposited in the central cell, and R is an empirical coefficient of value between 1 and 2, depending on which probability distribution has been used. There is an additional factor 2 because the invariant mass depends (to the first order) upon the product of the two calibration constants relative to the cells where e^+ and e^- have hit. The above formula suggests in fact that *only the energy collected by the central cell is useful in the calibration process (!),* and a similar degradation can be proved for the standard *E/p* method [which uses any isolated electrons that can be produced in several processes]. This result was obtained analytically and verified via a MonteCarlo simulation of a calorimeter's subsection (15 × 15 cells max), whereby the procedure converged correctly even for (very) poor initial constants.

The accuracy of this method is mainly dominated by the natural shape of the Z^0 and the additional factor 2. It requires therefore a larger number of electrons per cell than the standard *E/p* technique and may be numerically more demanding, given that the use of the invariant mass (in this way) establishes correlations among all cells (large matrices, not sparse); on the other hand the tracking itself would be needed only to identify the electrons but not to measure their momentum. Using isolated electrons (i.e. *E/p*) demands also, in fact, some extra cuts – mostly to protect from radiative losses – such as an isolation requirement and the selection of a window around the *E/p* peak, say of $\pm 2\sigma$, to isolate the Gaussian part of the distribution [NIK95]. This because Bremsstrahlung in the tracker and the presence of a magnetic field can separate the resulting photon and the electron at the calorimeter surface; the photon energy would be collected in the calorimeter itself but may not be included in the electron's tracking momentum measurement.

RUNNING AVERAGE

Running average schemes update the calibration coefficients in real time event after event and are particularly suited to online applications because of their small computing requirements. In the case of the GAMS calorimeter (NA12/WA102) the following recurrent expression was used to update the coefficient of the k-th cell within the n-th calibration event (shower), as reported in [BIN86] and [ALD87] (apparently the electron impact point was not used):

$$\frac{1}{a_k^{(n)}} = \frac{1}{a_k^{(n-1)}} \left(1 - \frac{E_e - E_0^{(n)}}{E_e} \frac{w_k^{(n)}}{\sum_{\nu=1}^{n} w_k^{(\nu)}} \right) \quad \text{with} \quad w_k^{(\nu)} = \left(\frac{E_k^{(\nu)}}{E_e} \right)^2$$

where E_e is the energy of the calibration electron, $E_k^{(n)} = a_k^{(n-1)} S_k^{(n)}$ is the energy deposited in the k-th cell, E_0 is the sum over all cell energies in the shower[20], and w_k is a weight factor whose form has been chosen to reduce the hadronic background (it was found out empirically to give better results than $w_k = E_k / E_e$).

For the calibration of the GAMS-4000 calorimeter 300 to 500 electrons per cell were collected in about 10 hours with a calibration beam. This method looks useful for a quick crosscheck of constants obtained with other techniques, but if desired one can increase its precision offline by cycling again through the data.

---

[20] In [ALD87] the total energy E_0 is defined as: $E_0^{(n)} = \sum_j E_j^{(n)} = a_k^{(n-1)} S_k^{(n)} + \sum_{j \neq k} a_j^{(\nu)} S_j^{(n)}$ where k is the index of cell under calibration, and ν refers to the "current" value (whatever that means) of a_j.

Acknowledgements:

A relevant part of this work has been carried out within the framework of the ESPRIT project P7255 *GP-MIMD2* (General-Purpose MIMD machines 2 [GPM93]), funded by the European Union. One of us (C. Bruschini) would therefore like to thank in particular *Fabrizio Gagliardi* (Project Management) and *Mike Metcalf* (CN/ASD/PAR Section Leader), who made his presence at CERN possible, without forgetting *Ken Peach* (Edinburgh), one of the first who believed in this type of (parallel) applications. This work would not have been possible without relying on the efforts of many friends and colleagues*: Laura M. Bertolotto* and *Paolo Calafiura* (parallel NMC and shower library), *John Apostolakis*, *Eric McIntosh* and *Bernd Panzer-Steindel* (system management, problem solving, many general questions), and *Ginette Bocquet-Jorat* (NMC). Special thanks go to *Les Robertson* (CERN Project Management).

Many interesting discussions on the facets of calorimeter calibration took place with Augusto Ceccucci (Torino, NA48), Fred James (CERN/CN), Yannis Karyotakis (LAPP Annecy/CERN, L3), Laurent Serin (Orsay, ATLAS), and Alexander Singovski (IHEP, GAMS calorimeter in NA12/WA102), as well as via E-mail with Christopher O'Grady and Brian Heltsley (Cornell).

References[21]*:*

CALORIMETRY CONFERENCES

[BRA90] K. Braune, The Crystal Barrel Collaboration, "The Electromagnetic Calorimeter of the Crystal Barrel Detector", Proceedings First International Conference on Calorimetry in High Energy Physics, World Scientific, p. 203, 29 Oct - 1 Nov 1990, FNAL, Batavia, Illinois (U.S.A.). [CERN C 92/105, Batavia 1990]

[ZHU90] R. Zhu, "Limits to the Precision of Electromagnetic Calorimeters", Proc First Intnl Conf on Calorimetry in HEP, p. 25.

[VIR91] T.S. Virdee, "Performance and Limitations of Electromagnetic Calorimeters", Proceedings Second International Conference on Calorimetry in High Energy Physics, World Scientific, p. 3, 14 - 18 Oct 1991, Capri, Italy. [CERN C 92/876, Capri 1991]

[AMO92] A. Amorim, L. Poggioli, A. Maio (RD1 Collaboration), "Calibration of an Electromagnetic Calorimeter for LHC using Z^0 -> e^+e^- Events", Proceedings Third International Conference on Calorimetry in High Energy Physics, World Scientific, p. 625, 29 Sep - 2 Oct 1992, Corpus Christi, Texas (U.S.A.). [CERN C 93/745, Corpus Christi 1992]

[NEW92] H.B. Newman, "A BaF_2 Crystal Precision Electromagnetic Calorimeter for GEM at the SSC", Proc Third Intnl Conf on Calorimetry in HEP, p. 229.

CLEO

[BLU86] E. Blucher, *et al.*, "Tests of Cesium Iodide Crystals for an Electromagnetic Calorimeter", NIM A: 249 (1986) 201-227.

[OGR90] C. O'Grady, The CLEO Collaboration, "Status of the CLEOII CsI Calorimeter", Proc First Intnl Conf on Calorimetry in HEP, p. 209.

[NOR91] E. Nordberg, "Performance of the CLEOII CsI Calorimeter", CLNS 91/1096 Note, Wilson Lab, Cornell University, Ithaca, N.Y. (U.S.A.). [CERN SW9206]
[Subm to the Workshop on Physics and Detectors for KEK Asymmetric B-Factory, Apr 15-18 1991]

[OGR91] C. O'Grady, "Bhabha Calibration of the CLEOII CsI Calorimeter", CBX 91-20 Note, Wilson Lab, Cornell University, Ithaca, N.Y. (U.S.A.).

[KUB92] Y. Kubota, *et al.*, "The CLEO II Detector", NIM A: 320 (1992) 66-113.

GAMS

[BIN86] F. Binon, *et al.*, "Hodoscope Multiphoton Spectrometer GAMS-2000", NIM A: 248 (1986) 86-102.

[ALD87] D. Alde, *et al.*, "Data Reduction Procedures for Large Multicellular Electromagnetic Calorimeters", CERN-EP/87-28, 12 Feb 1987. [Subm to NIM]

KTEV

[ARI92] K. Arisaka, *et al.*, "KTeV Design Report: Physics Goals, Technical Components, and Detector Costs", FERMILAB FN-580 Report, 22 Jan 1992, FNAL, Batavia, Illinois (U.S.A.). [CERN SW9352, LEP Library]

[CHI92] S. Childress, "KTeV CsI Crystal Calorimeter", Proc Third Intnl Conf on Calorimetry in HEP, p. 253.

[KES95] R.S. Kessler, *et al.*, "Beam Test of a Prototype CsI Calorimeter", FERMILAB-Pub-95/108, EFI-95-24 (Enrico Fermi Institute), May 1995, FNAL, Batavia, Illinois (U.S.A.). [Subm to NIM]

L3

[BAK88] J.A. Bakken, *et al.*, "Ionization Loss in BGO", NIM A: 270 (1988) 397-402.

[BAK89-1] J.A. Bakken, *et al.*, "High Energy Cosmic Muons and the Calibration of the L3 Electromagnetic Calorimeter", NIM A: 275 (1989) 81-88.

[BAK89-2] J.A. Bakken, *et al.*, "Study of the Energy Calibration of a High Resolution Electromagnetic Calorimeter", NIM A: 280 (1989) 25-35.

[KAA89] P.A. Kaaret, Thesis, Princeton University (1989), DoE/ER/3072/50.

[BOR89] B. Borgia, M. Diemoz, S. Morganti, "A Method to Calibrate the Light Response of a High Resolution Electromagnetic Calorimeter with Cosmic Rays", NIM A: 278 (1989) 699-704.

[ADE90] B. Adeva, *et al.*, "The Construction of the L3 Experiment", NIM A: 289 (1990) 35-102.

[21] NIM A stands for Nucl. Instr. Methods Phys. Res., A.

[SOU90] C. Souyri, “Calibration du Calorimètre Électromagnétique de L3 – Etude du Canal e^+e^- -> Z^0 -> $e^+e^-(\gamma)$”, Thèse (Université de Savoie), (EX) LAPP-T-90-03, May 1990. [CERN SW9107]
[BAY92] A. Bay, *et al.*, “The Xenon Monitor of the L3 Electromagnetic Calorimeter”, NIM A: 321 (1992) 119-128.
[BGO93] J.A. Bakken, *et al.*, The L3 BGO collaboration, “Results on the Calibration of the L3 BGO Calorimeter with Cosmic Rays”, CERN-PPE/93-184, Oct 22, 1993. [CERN SW9348; Subm to NIM]
[BGO95] (1987-1988) L3 BGO collaboration, J.A. Bakken, *et al.*, “Calibration of the L3 Electromagnetic Calorimeter in Electron Beam”, L3 Note 1712, Mar 1995.
[CERN WWW Home Page -> Experiments -> L3 -> Publications and Document.-> Internal Notes]
[KAR95] Y. Karyotakis, “The L3 Electromagnetic Calorimeter”, LAPP-EXP-95.02, Feb 1, 1995.
[CERN SCAN 9505077, SW 9520; Subm to the Intnl Calorimetry Symposium, Beijing Oct 94]

NA48

[CUN93] D. Cundy, “Calibration of Krypton Calorimeter”, Minutes of the 1993 NA48 Plenary Meeting, Apr 1993, Cambridge (U.K.).
[CEC95] A. Ceccucci and G. Lasio, “The NA48 Electromagnetic Calorimeter: Some Simulation Results”, NA48 Note 95-27, 6 Sep 1995.

PARALLEL NMC, GP-MIMD2 PROJECT

[GPM93] Project P7255 GP-MIMD2, described in “High Performance Computing and Networking, Summaries of Projects (Synopsis)”, Jan 1993, CEC DG-XIII.
[APO94] J. Apostolakis, *et al.*, “First Results from the Parallelisation of CERN's NA48 Simulation Program”, High-Performance Computing and Networking in Europe 1994 (HPCN’94) Proceedings, pp. 371-376, Apr 1994, Munich - Germany.
[BER94] L.M. Bertolotto, *et al.*, “Feasibility studies for a High Energy Physics MC Program on Massive Parallel Platforms”, Computing in High Energy Physics 1994 (CHEP’94) Proceedings, pp. 374-378, Apr 1994, San Francisco - U.S.A.
[JAM94] F. James, RANLUX: “A Fortran implementation of the high-quality pseudorandom number generator of Lüscher”, Computer Phys. Commun. 79 (1994) 111-114.
[BRU95] C. Bruschini, *et al.*, “Deliverable 3.1.3.a: Report on Event Simulation on the CS-2”, EU Project P7255 (GP-MIMD2) Internal Report, Feb. 1995.

VARIOUS

[ABE91] F. Abe, *et al.*, CDF Collaboration, Phys. Rev. D43 (1991) 2070.
[PRE92] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, “Numerical Recipes in C: The Art of Scientific Computing”, 2nd Edition, Cambridge University Press, 1992.
[AMO93] A. Amorim, L. Poggioli, A. Maio, “Calibrating the Electromagnetic Calorimeter using Z^0 -> e^+e^- Events”, ATLAS Internal Note PHYS-NO-015, 27 Jan 1993.
[BAB93] V.V. Babintsev, “Some Calibration Methods for Calorimeters”, IHEP 93-73 OHF, Protvino, 26 May 1993. [CERN SW9420]
[LIN93] D. Lincoln, G. Morrow, P. Kasper, “A Hidden Bias in a Common Calorimeter Calibration Scheme”, FERMILAB-Pub-93/394, Dec 1993. [Subm to NIM]
[NIK95] A. Nikitenko, P. Verrecchia, D. Bomestar, CMS TN/95-019, 1 Feb 1995.

WWW LINKS

- *CLEO* Home Page (Wilson Lab, Cornell): *http://w4.lns.cornell.edu*
  [or CERN Home Page -> High-Energy Physics Laboratories (under “About Physics”) -> Wilson Lab/CLEO]
- *NAG* Home Page: *http://www.nag.com*
  -> Numerical (under “NAG Products and Services”) -> Specialist Libraries and Packages -> Harwell Sparse Matrix Library (HSML)
- *Rutherford Appleton Laboratories* Home Page: *http://www.rl.ac.uk*
- *RAL Numerical Analysis Group*: *http://www.rl.ac.uk/departments/ccd/numerical/numerical.html*
  -> Harwell Subroutine Library -> Matrices and linear algebra (compressed PS file)
- *Netlib* software repository: *http://www.netlib.org*
  -> The Netlib Repository -> Harwell

Appendix:

A1. Some Words on the KTeV Strategy

The Fermilab KTeV experiment E832 ([ARI92], [CHI92], [KES92]) will use an electromagnetic calorimeter composed of 3256 CsI crystals, subdivided into 2232 inner crystals of $2.5 \times 2.5 \times 50$ cm, and 1024 outer of $5 \times 5 \times 50$ cm, for a total of $2 \times 2 \times 0.5$ m and 27 X_0 (radiation lengths). Three techniques will be combined *in situ* to track individual gain variations with time and rate, understand nonlinearities and changes in individual crystals (light output, transparency):

- *Laser calibration (flasher system):* it is aimed at monitoring variations in the photomultiplier response that occur on a timescale too short for the other two methods. It gives an independent measurement of photomultiplier/ADC nonlinearities over the full dynamic range. The laser monitoring system will be designed to operate at the 0.1% level.
- *Special e^+e^- calibration runs*: they will be implemented by letting the photons convert into pairs and then using appropriate "splitting magnets"; the latter will be automatically sequenced by the data acquisition system. These runs should provide a snapshot of the calorimeter's performance and will be useful to understand the initial setup, and to monitor changes in individual crystals due to radiation damage. They will be taken weekly and last some hours, possibly with the addition of an intelligent L3 trigger to enhance less populated blocks; a rate of 1000 usable events per block per 2 hours is expected (!). It is envisaged, for certain applications, to interpolate between these runs with the laser monitoring system.
- *Large samples of Ke3*: they should give the ultimate measure of performance (such as tuning the electron energy scale) and are meant to be recorded simultaneously with data taking, i.e. in real operating conditions, to be then analyzed offline. About 40000 Ke3 events are planned to be taken per 23 sec spill, for a total of 40000 Ke3 per block per week (or 250 per block per hour). It is not specified how many of these events would be usable for calibration purposes, and one has to note that the outer CsI crystals have the same surface as six NA48 LKr towers.

A2. A Hidden Bias in a Common Calorimeter Calibration Scheme

The energy minimization technique contains in fact a hidden bias, as described in [LIN93] (and pointed out by A. Ceccucci, Torino) and Ch. 2 of [BAB93]. This is best illustrated by considering the calibration of only one cell (not restrictive), with calibration coefficient a, using N particles of fixed energy E, each giving a signal S_k. The average and RMS of the distribution of S_k are:

$$\overline{S} = \frac{1}{N}\sum_{k=1}^{N} S_k \quad , \quad \sigma^2 = \frac{1}{N}\sum_{k=1}^{N}\left(S_k - \overline{S}\right)^2 = \frac{1}{N}\sum_{k=1}^{N} S_k^2 - \left(\overline{S}\right)^2 ,$$

and the quantity to be minimized is:

$$\chi^2 = \frac{1}{N}\sum_{k=1}^{N}\left(aS_k - E\right)^2$$

Finding the a_{min} that minimizes χ^2 leads to the following result:

$$\frac{\partial\chi^2}{\partial a} = 0 \Rightarrow a_{\min} = \frac{E\overline{S}}{\sigma^2 + \overline{S}^2}$$

whereas the intended one is $a_{\text{expected}} = E / \overline{S}$, and so the ratio of minimized to expected constant is:

$$\frac{a_{\min}}{a_{\text{expected}}} = \frac{\overline{S}^2}{\overline{S}^2 + \sigma^2} \le 1.$$

This interesting effect can fake a nonlinearity and is important at low energy and/or for hadronic calorimeters. The error estimated in the NA48 case is the following:

$$\frac{\sigma}{E} \approx \frac{3.5\%}{\sqrt{E}} \Rightarrow \frac{a_{\min}(E)}{a_{\text{expected}}(E)} = \frac{E^2}{E^2 + 0.035^2 E} = \frac{1}{1 + 0.035^2 / E} = \frac{1}{1 + 10^{-3} / E}$$

A3. Characteristics of Some Large Electromagnetic Calorimeters

L3 (BGO)

The L3 electromagnetic calorimeter is composed of 10734 tapered BGO crystals, 2×3840 in the barrel and 2×1527 in the endcaps. Each crystal is 24 cm long, covering 2×2 cm at the front and 3×3 cm at the back, and its light output is strongly temperature dependent (-1.55 % per °C, monitored by 1792 thermal sensors). The Moliere radius is 2.3 cm and 93% of the energy is deposited in a 3×3 shower box.

Calibration of all barrel and part of the endcap crystals, with the final electronics and support structure, was carried out with electron beams extracted from the SPS, at energies of 2, 10, 20 and 50 GeV, after initial measurements and precalibration with cosmic rays. The total effort was of about 3 years.

The *in situ* calibration at LEP (time dependent corrections) is carried out using a Xenon light monitor to illuminate each of the crystals, which are grouped for this purpose in geographical regions. The intercalibration itself is carried out by adjusting the constants to get the same answer for all crystals in the same region, whereas the absolute scale is fixed using electrons from non radiative Bhabha decays, achieving an absolute calibration precision of 0.9%. Partly because of the limited statistics available at LEP (this type of event is peaked in the forward direction) a degradation of the resolution in the barrel is registered, compared to the one achieved at the SPS. Temperature fluctuations also play an important role, as well as the radiative part of Bhabha events. It has to be noted that isolated electrons can be used for calibration purposes only with difficulty given the poor TEC (Time Expansion Chamber) momentum determination efficiency at high energies (Y. Karyotakis, *private communication*).

CLEO (CsI)

The CLEO electromagnetic calorimeter is installed at the Cornell e^+e^- storage ring, peaked at the Y (Upsilon) mass, and totals 7800 CsI crystals, 6144 of them in the barrel and 828 in each of the two endcaps.

GAMS (LEAD GLASS)

The GAMS-4000 electromagnetic calorimeter is composed of 4092 identical lead-glass counters of square cross-section (3.8×3.8 cm) and 45 cm long (16 radiation lengths), assembled in a 64×64 matrix covering a surface of 6 m^2 (!).

A4. Ke3 Acceptance, detailed table:

A number of parallel runs with higher statistics were executed to estimate the acceptance for Ke3 events (see Ch. 3) using the full K_L momentum range and decay range, which translates into the following Generation Limits (GLIM): *0 GeV/c < p < 450 GeV/c* and *480 cm < Z_{vtx} < 12109.5 cm*. The multiple scattering was also switched on (type 1), although it did not produce a large effect.

The Level 1 trigger was used with: *Calibration factor for HAC: 1, Energy cut for E_{tot} (GeV): 20, Min # space points in DCH1 (used if optNDETSIM=1): 2, Number of HOD counters for extension: 2, and Cut activated (=1) on HOD QX, Mu veto, (AKS), Antis*.

The Level 2 charged trigger parameters were left untouched and are as follows: *Drift time window: 50.0 to 200.0 ns, Sum of drift distances match within 0.20 cm, Maximum drift time: 200.0 ns, Number of bins for U/V LUT: 2048, Maximum extrapolation error: 5.00 cm, Vertex z coordinate window: 200.0 to 3100.0 cm, Two-track mass window: 0.475 to 0.520 GeV/c^2, Triplets data reduction performed, min. coord. separation: 0.50 cm.*

Results were obtained for all combinations (logical AND) of the following sets of cuts:

quantity [unit]	*description*	*condition #1*	*condition #2*	*condition #3*
Z_{vtx} *[cm]*	z coord of the K_L decay vertex	< 3000	< 5000	< 7000
$D_{e\pi}$ *[cm]*	distance between e and π	> 50	> 70	> 100
E_e *[GeV]*	electron energy	0 ÷ 9999	15 ÷ 25	
$\theta_{e/pointing}$ *[rad]*	electron impact angle	< 0.01	< 0.02	

with the underlined values referring to the "final" Ke3 acceptance conditions. We will concentrate on three runs (31, 32, 33), each of them generating 10^6 events. Their general characteristics are summarized in the following table, where the fourth column – *total # K^0* – is relative to the total number of events analyzed, i.e. passing the first or second level trigger[22]. The last four columns report the number of Ke3 events after each cut for the "final" Ke3 acceptance conditions described above. All numbers have to be normalized to the Ke3 rate of about 100 kHz, one million events being therefore roughly equivalent to four bursts.

Run #	*magnetic field (p_T)*	*trigger level*	*total # K^0 (Ke3)*	Z_{vtx}	Z_{vtx}•$D_{e\pi}$	Z_{vtx}•$D_{e\pi}$•E_e	Z_{vtx}•$D_{e\pi}$•E_e•$\theta_{e/pointing}$
31	nominal (257)	L1	201796 (199243)	131741	53587	21048	2949
32	reduced (100)	L1	210740 (208440)	136879	55455	21125	21102
33	nominal (257)	L2	9677 (9637)	9637	5032	2060	326

The following table contains all numbers relative to runs 31 (and 32 in brackets):

Z_{vtx} *[cm]*	$D_{e\pi}$ *[cm]*	E_e *[GeV]*	$\theta_{e/pointing} < 0.01$ *[rad]*	$\theta_{e/pointing} < 0.02$ *[rad]*
< 3000	> 50	ALL	36611 (64038)	58821 (67147)
< 3000	> 50	15 ÷ 25	3985 (20505)	19796 (20505)
< 3000	> 70	ALL	26641 (51686)	47199 (54657)
< 3000	> 70	15 ÷ 25	3345 (18730)	17955 (18730)
< 3000	> 100	ALL	12684 (30733)	28073 (33106)
< 3000	> 100	15 ÷ 25	1965 (13106)	12807 (13106)
< 5000	> 50	ALL	60207 (112020)	102182 (120206)
< 5000	> 50	15 ÷ 25	7485 (37549)	36140 (37573)
< 5000	> 70	ALL	41246 (87044)	79301 (94877)
< 5000	> 70	15 ÷ 25	5807 (32971)	31756 (32995)
< 5000	> 100	ALL	17973 (48512)	44612 (54725)
< 5000	> 100	15 ÷ 25	2949 (21102)	21047 (21125)
< 7000	> 50	ALL	73897 (146962)	133306 (163188)
< 7000	> 50	15 ÷ 25	10541 (51931)	49825 (52350)
< 7000	> 70	ALL	48066 (109160)	100138 (124512)
< 7000	> 70	15 ÷ 25	7427 (43052)	42091 (43470)
< 7000	> 100	ALL	19788 (56625)	53290 (68036)
< 7000	> 100	15 ÷ 25	3386 (25001)	25942 (25322)

[22] The number in brackets *(Ke3)* refers to events where, in addition, both the electron and pion are requested to hit the LKr calorimeter (this condition is present in the code for "historical reasons").

Same as above for run 33:

Z_{vtx} [cm]	$D_{e\pi}$ [cm]	E_e [GeV]	$\theta_{e/pointing} < 0.01$ [rad]	$\theta_{e/pointing} < 0.02$ [rad]
< 3000	> 50	ALL	4802	8100
< 3000	> 50	15 ÷ 25	534	2803
< 3000	> 70	ALL	3842	6960
< 3000	> 70	15 ÷ 25	483	2643
< 3000	> 100	ALL	1998	4398
< 3000	> 100	15 ÷ 25	314	1988
< 5000	> 50	ALL	4967	8414
< 5000	> 50	15 ÷ 25	554	2913
< 5000	> 70	ALL	3966	7223
< 5000	> 70	15 ÷ 25	500	2746
< 5000	> 100	ALL	2056	4555
< 5000	> 100	15 ÷ 25	326	2060
< 7000	> 50	ALL	4967	8414
< 7000	> 50	15 ÷ 25	554	2913
< 7000	> 70	ALL	3966	7223
< 7000	> 70	15 ÷ 25	500	2746
< 7000	> 100	ALL	2056	4555
< 7000	> 100	15 ÷ 25	326	2060

A5. Task-Farming Parallelisation

The task farming parallelisation scheme, in which each process(or) generates a full event, was implemented using several tasks – actually copies of the same executable, whose behaviour is established at run time as described in Ch. 4 – that communicate with each other via message passing interfaces, i.e. (see [APO94], [BER94], [BRU95]):

- one *master program*, called *Source*, which first initializes the working environment of the message passing interface. During execution it synchronizes the slave processes and distributes the relative work loads by instructing each worker on how many events it should generate (number of the first and last one). These work "packets" are handed out until the production is finished; their size can be predefined or, even simpler, be equal to $N_{packet} = N_{events} / N_{workers}$. No communication with the source takes place during the simulation of the N_{packet} events: the message passing overhead can therefore be kept quite low and the source can in principle run on any processor. Note that the treatment of random numbers is also quite important since they have to be generated and distributed correctly[23], irrespective of the number of processors in the system. The solution we decided to adopt, namely to use a different sequence of random numbers per each event, is outlined in App. ***A6***.
- several independent *slave programs*, or *Event Workers*, which do the actual event generation and do not communicate with each other. A typical CS-2 configuration is characterized by having (at least) *two Event Workers per dual-processor node,* in order to fully load the machine.

Concerning histograms in particular it was initially decided to rely on a file locking/unlocking mechanism (J. Apostolakis, CERN) to merge all contributions from the single event workers. This scheme works fine but is quite time consuming when dealing with a large number of event workers (say 20), because one after the other has to lock a file, read it in, add its own contributions and write it out[24]. A valid alternative is provided by an HBOOK subroutine called HMERGE which has become available quite recently: it merges identical histogram files from a given list into one output file. In this case one has to make sure that all individual files exist and are accessible when HMERGE is called (it may become necessary to introduce an appropriate delay in the code).

[23] In a way such that each worker will really generate different events.
[24] In fact what makes this procedure heavy is that each worker writes out its own histograms as well as the merged ones, and the output file gets correspondingly large. This might be avoided if necessary.

A6. Handling of Random Numbers

A number of requirements have to be met when handling random numbers in a MonteCarlo program such as NMC. In particular ([BRU95]):

- it must be possible to reproduce a sequence of events regardless of some conditions which can be modified by the user; that is, the kinematics of each event and thus the detector signals must not change. For example accidental events can then be added without modifying the signal events that follow.
- a large number of independent random numbers may be used in the generation of a single event, and they have to be computed rapidly.
- due to the desired high precision of the experiment, large samples of events must be generated without any correlation.
- the serial and the parallel versions of NMC must generate exactly the same events. This is of fundamental importance when cross-checking the two codes and certainly increases the user's confidence in the parallel version.

The random generator RANLUX (KERNLIB V115, [JAM94]) was chosen after having investigated several possibilities; this routine generates pseudorandom numbers uniformly distributed in the interval (0,1). Each call produces an array of single-precision real numbers of which 24 bits of mantissa are random, and the user can choose a so-called "luxury level" which guarantees the required quality of the generated random numbers. The period of the generated pseudorandom numbers is greater than 10^{165} and they can be extracted from any of 900 million independent sequences[25]. The latter feature in particular was exploited to initialize a different sequence for each signal event, corresponding to the event number. Such a sequence provides all the necessary random numbers for that event and for its accidentals, if any. This and the other characteristics mentioned above make RANLUX an ideal random generator for our purposes.

[25] Switching from one to another takes only 0.3 msec on the CS-2.