LHCb/2003-061
TRIG
August 14, 2003

Execution time optimization of the Level-1 algorithm

M. Witek

Institute of Nuclear Physics, Cracow
and
CERN, Geneva

Abstract

The measurement of the execution time of Level-1 algorithm is presented. The code has been reorganized to approach limits imposed by the predicted computing performance in 2007. The possible further optimizations are also discussed.

Contents

1 Introduction

An important constraint of the Level-1 (L1) algorithm is an upper limit on its execution time. The main role of the trigger, a suppression of minimum bias events while preserving high efficiency for signal channels, is related to the quality of the information for making the L1 decision. Short execution time affects the performance of the reconstruction leading to worse momentum resolution, worse track finding efficiency or higher ghost track rate and finally an overall performance of the L1 might be significantly degraded. Therefore proving that the L1 algorithm can be executed fast enough is indispensable to validate the physics results that are expected from the MC study.

So far the L1 code was developed inside an off-line environment. Due to the evolution of the software and the modifications of detector technology, it was kept in a maintainable form and only the critical blocks were benchmarked to check whether the timing were not too far from the limits. In this note the implementation of the algorithm that was used to simulate L1 decision for the analysis of the data from large production started in February 2003 is presented. The code has been reorganized and some technical modifications have been implemented to decrease execution time and to enable its reliable measurement.

In this note an emphasis is put on explaining the features strictly related to the optimization of the execution time. The various tasks of L1 algorithms have been already described elsewhere and the references are given in the text.

2 The logic of L1 algorithm

For a reliable time measurement, the different parts of the L1 code were grouped together to be executed in a sequence that is not interlaced with any other algorithm. The structure consists of a main L1 algorithm and a few slave tools that implement different tasks like matching VELO tracks to Level-0 (L0) objects or momentum determination by VELO-TT matching. This way a real time elapsed between start and stop of the algorithm is a reliable estimation of the total execution time. In particular the overhead due to communication via dynamic containers between the different parts of the code can be estimated and properly subtracted. This overhead of the order of milliseconds[1], is negligible for the off-line applications but becomes a sizable fraction of the time in the case of on-line environment where the average budget is below 6 ms per event.

Before starting the L1 processing all necessary data should reside in memory as in the case of real situation on the on-line farm. A dedicated algorithm reads the L1 input data into a computer memory, namely VELO clusters, TT clusters, L0 decision report and L0 objects (high p_T hadrons, muons, electrons and photons). The execution flow is organized as a sequence of steps, some of them are executed conditionally depending on the result of previous steps or a specific feature of the event.

---

[1] Unless explicitly stated all CPU timings in the note are quoted for 1 GHz Pentium III processor.

The processing chain for each L0 accepted event is the following:

1. **VELO initialization**
   All necessary information for VELO reconstruction is decoded, preprocessed and cached locally.

2. **2D VELO reconstruction**
   A fast 2D reconstruction in RZ projection (the RZ tracks are straight lines) is based on an automaton method. The details of the implementation can be found in Ref. [1].

3. **Primary vertex fit**
   Due to the segmentation of RZ VELO sensors into 45° sectors, the primary vertex (PV) can be precisely determined using only 2D tracks. The procedure is described in Ref. [1].

4. **2D track selection**
   Only a few VELO tracks contain the information useful to distinguish between the signature of $b\bar{b}$ and minimum bias events. Basing on the PV estimation from the previous step the tracks with 2D impact parameter in the range between 0.15 mm and 0.3 mm are selected for 3D reconstruction. In addition if there is a L0 muon candidate, a matching procedure to L0 objects is executed and the tracks passing a loose compatibility criteria are also marked for 3D reconstruction. The procedure of VELO track matching to L0 objects is described in Ref. [2]. The initial average number of 58.4 forward 2D tracks in minimum bias events passing L0 is reduced to only 8.5 after the selection.

5. **3D VELO reconstruction**
   For the selected 2D tracks the ϕ-sector information is linked and the 3D VELO tracks are constructed [1]. They are used as a basic set of tracks for making the L1 decision.

6. **Refit of 3D tracks**
   The momentum of the 3D track is not known at the stage of standalone VELO reconstruction. A dedicated method is used to obtain the optimal track parameters for extrapolation in both directions, to calculate their impact parameter to the PV and to search for their track segment in the TT station.

7. **VELO-TT initialization**
   The TT signals are preprocessed to collect locally all information for fast VELO-TT matching.

8. **VELO-TT matching**
   For each 3D track a fast search for the segment in TT is done and, if successful, the momentum of these VTT tracks is determined with a precision in the range of 20%-40% from their deflection in the fringe magnetic field between VELO and TT. The procedure of VELO-TT matching at L1 is described in details in Ref. [3].

9. **L0 3D matching**
   The 2D L0 object matching is confirmed with 3D tracks using tighter acceptance conditions.

10. **Decision**
    The last step is the actual L1 decision. The discriminant variable for selecting $b\bar{b}$ events is constructed based on the VTT tracks and L0 objects. The details about the discrimination method can be found in Ref. [4].

3 Code optimization

The modifications fall into two categories: a structural one considering fast methods of reconstruction and the arrangement of tasks to reduce number of calculations before rejecting the event[2], and a pure technical optimization of the code.

The main structural modifications are the following:

- the description of specific fast solutions of the 2D and 3D VELO reconstruction are described in details in Ref. [1].
- conditional 2D and 3D matching to L0 muon candidates (only about 20% of L0 accepted minimum bias events contain a muon candidate).
- conditional 3D reconstruction for selected 2D tracks (high impact parameter or successful matching to L0 muon)
- VELO-TT matching: (i) a window size for TT hits search is limited by requiring the transverse momentum of the track to be at least 300 MeV. This is of particular importance for low angles where the density of hits is high; (ii) the first matched TT segment (the highest momentum one) of required quality is accepted.

The technical modifications are the following:

- dynamic containers were replaced by static ones. If the replacement was not possible the time difference between dynamic and static solution has been subtracted from the total execution time. This overhead was measured to be at the level of 20% (2 ms) of the total L1 execution time.
- usage of lookup tables. For VELO-TT matching, $\int Bdl$ and z coordinate of the "middle of the magnet" (half $\int Bdl$) was calculated once, at the beginning of the job, and stored into a table. The parametrization was made as a function of the z position of the primary vertex, the z coordinate of the VELO exit point and the y slope of the VELO track (the dependence on slope x was negligible). Another example is a table to store geometrical information like coordinates of first strip of a sensor, strip pitch or stereo angle, which are retrieved at the beginning of the job from the geometry database.

---

[2] On L0 accepted events 96% of L1 decisions is supposed to be negative.

- reduction of a number of calls to functions $sin(x), cos(x)$ or $sqrt(x)$ inside loops.

- collecting all information locally before starting combinatorial calculations.

- removal of all conditional debug printouts (even if condition is false and nothing is printed) from heavily used parts of the code.

4 Present performance

On PC Linux machines the method providing sufficient time measurement accuracy of about 1 μs is based on counting of processor cycles between the two points of the running code. The disadvantage is that the measurement is sensitive to the load of the machine. During the execution, the algorithm might be suspended by a system in favor of another concurrent process or due to an operation on a system page file. This effect can only increase the execution time[3]. The timing for different L1 steps is presented in Table 1. The main contribution comes from the initializations and the parts dealing with high combinatorics like 2D or 3D tracking. Relatively large contribution of 2D tracks selection is due to L0 muon matching (the matching procedure is fast but is executed for about 58 tracks).

Level-1 phase	time [ms]
VELO initialization	1.1
2D tracks	2.4
PV fit	0.5
2D selection	1.3
3D tracks	1.4
refit of 3D tracks	0.2
TT initialization	0.4
TT matching	0.5
L0 3D matching	0.4
Decision	0.1
Total	8.3

Table 1: The timing of various phases of the L1 algorithm as measured on a 1 GHz Pentium III Linux processor.

The average time for processing an L0 accepted event was measured to be 8.3 ms. The corresponding distribution is presented in Fig. 1. To estimate if LHCb can afford to execute this algorithm on 2007 CPU nodes one has to extrapolate the computer power over the next four years. It is connected with an uncertainty that is related to the limits the current processor technology is approaching and to the market

---

[3] On machines running only the L1 algorithm the spread of measurements for same input data was measured to be below 3%.

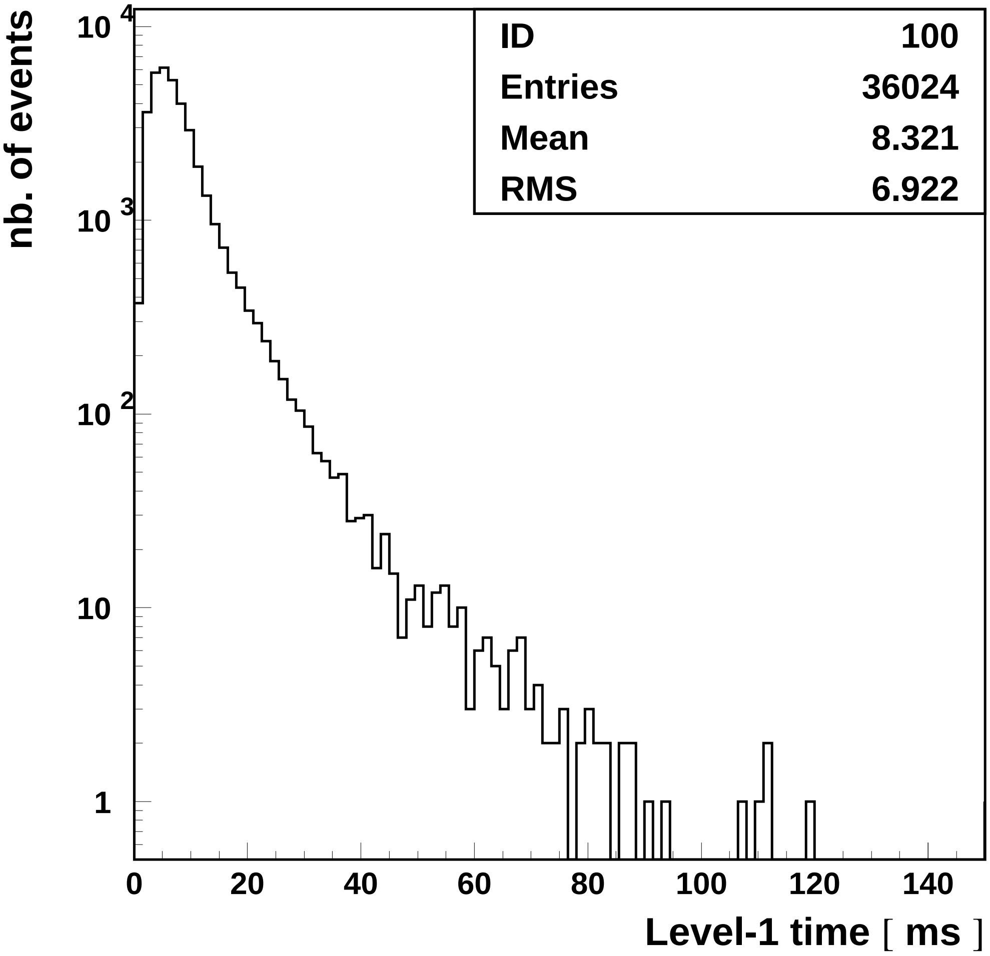

Figure 1: The distribution of the Level-1 execution time obtained on a 1 GHz Pentium III processor for minimum bias events accepted by Level-0.

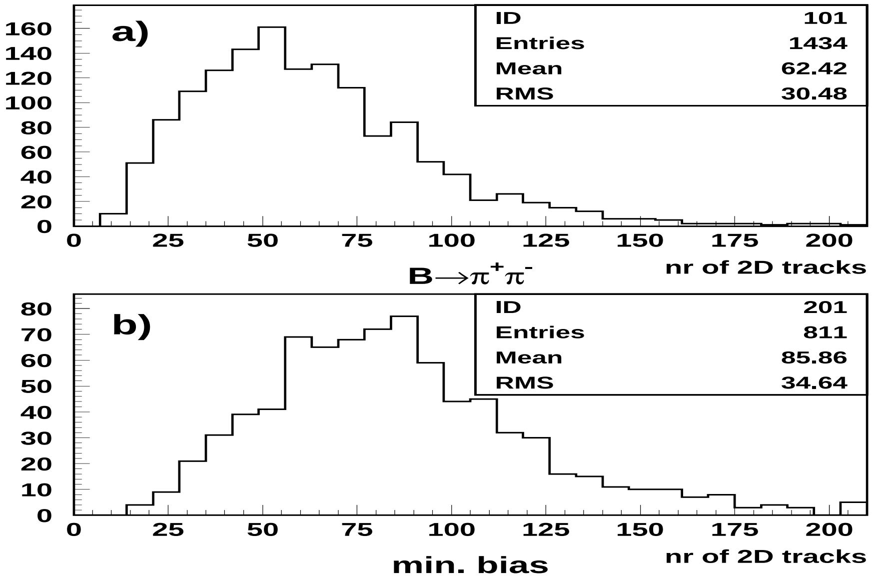

Figure 2: The distribution of number of forward reconstructed 2D tracks for L1 accepted events: $B_d^0 \rightarrow \pi^+\pi^-$(a) and minimum bias (b).

demands that are difficult to guess. Nevertheless from the prediction presented in [5] one can count on a gain factor of 6 going from 1 GHz Pentium III processor to a typical one in 2007. Hence the present code would require a farm of about 1400 nodes with a L0 rate of 1 MHz. Nevertheless, some further modifications to the code are described in the Sec. 5. Their implementation was postponed to freeze the code for physics studies of the large scale production data, where the synchronization between trigger and off-line selections required stable code.

5 Future optimizations

Recently a new implementation of VELO reconstruction has been developed for the high level trigger [6]. The time for 2D and 3D track reconstruction with similar performance was reduced to 0.6 ms and 0.3 ms respectively. Taking this into account the total execution time of the L1 code drops to 5.4 ms. It can be lowered by further modification to other parts of the code. One possibility is a treatment of events with high number of 2D tracks and high number of 2D tracks selected for 3D reconstruction. The multiplicities of signal and minimum bias events accepted by L1 differ slightly. As can be seen in Fig. 2 and 3, the average number of 2D tracks reconstructed and selected for $B_d^0 \rightarrow \pi^+\pi^-$ is smaller. It was shown that removal of busy events does not affect the final efficiency of off-line selected events. A good example is the SPD multiplicity cut applied already at the L0 [7]. The compensation mechanism responsible is the following. The cut at a given value of multiplicity reduces both signal efficiency and minimum bias retention. Retuning of minimum bias retention to a nominal 40 kHz brings back the signal efficiency close to

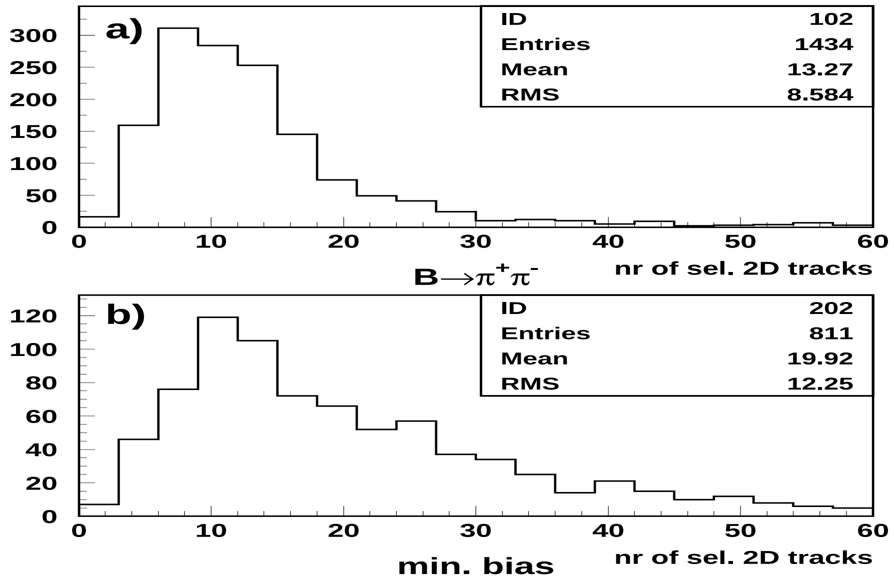

Figure 3: The distribution of number of selected 2D tracks for L1 accepted events: $\mathrm{B_d^0} \to \pi^+\pi^-$(a) and minimum bias (b).

	2D reconstructed	2D selected	execution time [ms]
No cut	-	-	8.3
Cut 1	120	40	7.7
Cut 2	110	35	7.5
Cut 3	100	30	7.2
Cut 4	90	25	6.7

Table 2: The variation of mean L1 execution time for different cuts on number of 2D reconstructed tracks and number of 2D selected tracks.

the original value. Moreover, the busy events are more exposed to a reconstruction errors which tends to wash out the difference between the physics features of signal and minimum bias events. After removal the overall signal efficiency might be better then the initial one for certain channels. The execution time dependence on multiplicity cuts are listed in Table 2. The time shown in the last column includes the contribution from all steps necessary to apply the cut. In case of 2D tracks multiplicity, the time up to the end of 2D reconstruction is counted. Similarly for a cut on number of 2D selected tracks, the selection step is included. As can be seen in Fig 4 the tail of initial distribution is gradually reduced for tighter cuts. The average execution time can be significantly reduced for relatively loose cuts.

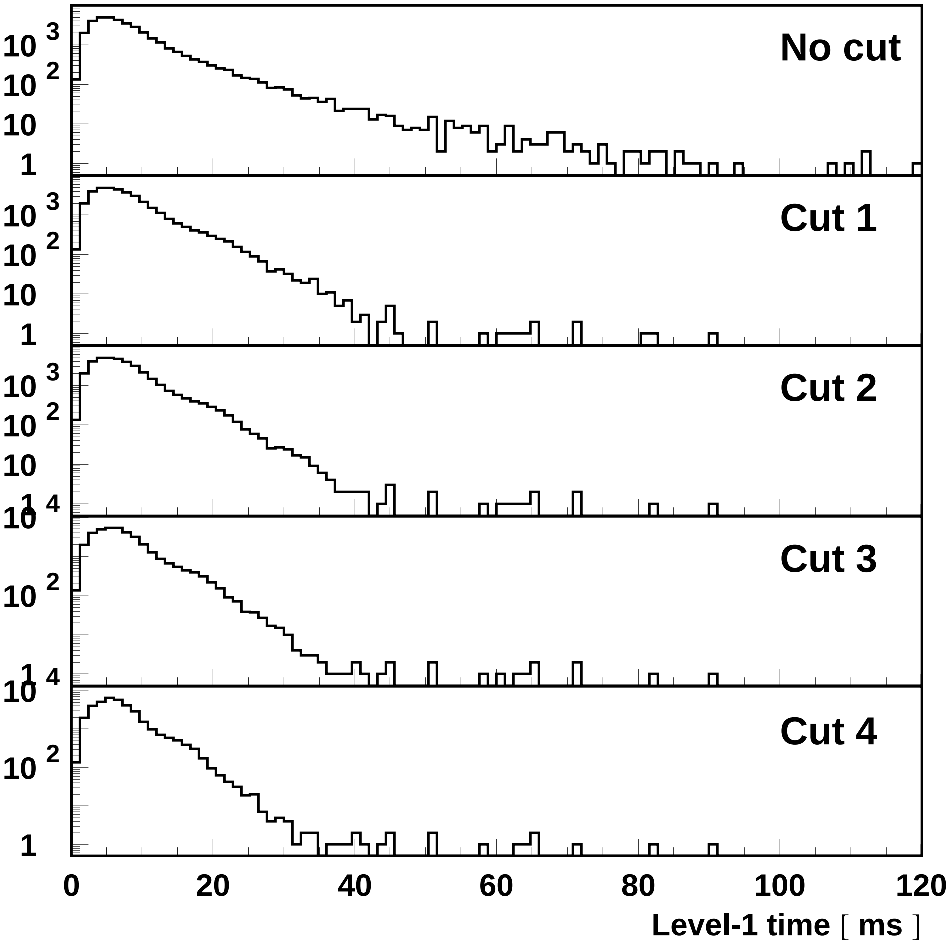

Figure 4: The distributions of L1 execution time for the cuts listed in Table 2 for minimum bias events accepted by L0.

6 Summary

The execution time of the complete software implementation of the Level-1 algorithm has been measured. The code has been reorganized to minimize the mean execution time for minimum bias events passing Level-0. The main modifications aimed to reduce the number of calculations before rejecting event, as 96 % of L1 decisions is supposed to be negative. Some purely technical modifications have also been made. The code has not been fully optimized to provide a stable environment for the data analysis of the mass production that started February 2003. The performance of the code turned out to be satisfactory close to the nominal design value. The mean execution time was measured to be 8.3 ms. Further optimizations are in progress and indicate that a value of 5.1 ms is possible, which would correspond to a L1 farm of 850 nodes in 2007. Further improvements are being studied like the treatment of the high multiplicity events.

References

[1] *The LHCb Level-1 Trigger: Architecture, Prototype, Simulation and Algorithm*, V. Lindenstruth, D. Atanasov, K. Giapoutzis *et al.*, LHCb 2003-064, Chapter 5.

[2] *Matching VELO tracks to L0 objects*, N. Tuning, LHCb 2003-039

[3] *VELO-TT matching and momentum determination at Level-1 trigger*, M. Witek, LHCb 2003-060

[4] *The use of the TT1 tracking station in the Level-1 trigger*, H. Dijkstra, T. Schietinger, F. Teubert, M. Witek, D. Wiedner, LHCb 2002-045

[5] *Processors, Memory and Basic Systems*, Working Group A, PASTA 2002 ed. http://lcg.web.cern.ch/LCG/PEB/ PASTAIII/pasta2002Report.htm

[6] *Velo tracking for the High Level Trigger*, O. Callot, LHCb 2003-027

[7] *Effect of Multiplicity Cuts on the L0 and L1 Triggers*, M. Ferro-Luzzi, LHCb 2003-047