Informes Técnicos Ciemat

977
Octubre, 2001

Buffer of Events as a Markovian Process

J. Berdugo
J. Casaus
C. Mana

Departamento de Fusión y Física de Partículas Elementales

Toda correspondenica en relación con este trabajo debe dirigirse al Servicio de Información y Documentación, Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas, Ciudad Universitaria, 28040-MADRID, ESPAÑA.

Las solicitudes de ejemplares deben dirigirse a este mismo Servicio.

Los descriptores se han seleccionado del Thesauro del DOE para describir las materias que contiene este informe con vistas a su recuperación. La catalogación se ha hecho utilizando el documento DOE/TIC-4602 (Rev. 1) Descriptive Cataloguing On-Line, y la clasificación de acuerdo con el documento DOE/TIC.4584-R7 Subject Categories and Scope publicados por el Office of Scientific and Technical Information del Departamento de Energía de los Estdos Unidos.

Se autoriza la reproducción de los resúmenes analíticos que aparecen en esta publicación.

Depósito Legal: M-14226-1995
ISSN: 1135-9420
NIPO: 402-01-008-3

Editorial CIEMAT

CLASIFICACIÓN DOE Y DESCRIPTORES

S72
ASTROPHYSICS; MARKOV PROCESS; COMPUTER CODES; PARTICLES; PHYSICS

Buffer of Events as a Markovian Process

Berdugo, J.; Casaus, J.; Mana, C.
Astroparticle Physics Department, CIEMAT, 28040 Madrid, Spain

14 pp. 3 figs. 0 refs.

Abstract

In Particle and Asro-Particle Physics experiments, the events which get trough the detectors are read and processes on-line before they are stored for a more detailed processing and future Physics analysis. Since the events are read and, usually, processed sequentially, the time involved in these operations can lead to a significant lose of events which is, to some extent, reduced by using buffers. We present an estimate of the optimum buffer size and the fraction of events lost for a simple experimental condition which serves as an introductory example to the use of Markov Chains.

Buffer de Sucesos como un Proceso Markoviano

Berdugo, J.; Casaus, J.; Mana, C.
Departamento de Astrofísica de Partículas, CIEMAT, 28040 Madrid, Spain

14 pp. 3 figs. 0 refs.

Resumen

En experimentos de Astrofisica y Física de Partículas Elementales, los sucesos recogidos en los detectores se leen y procesan *"on-line"* antes de ser almacenados para una análisis posterior. Puesto que, normalmente, estas operaciones se realizan secuencialmente, el tiempo que conllevan puede dar lugar a una perdida significativa de sucesos que se puede reducir, hasta cierto punto, utilizando *"buffers"* de sucesos. Presentamos una estimación del tamaño óptimo del *"buffers"* y de la fracción de sucesos perdidos para una configuración experimental sencilla como ejemplo introductorio a la aplicación de Cadenas de Markov.

1 Introduction.

In a typical Particle or Astro-Particle Physics experiment, when an event gets through the detector the signals are read and the information is processed on-line before it is stored for a more detailed reconstruction and future Physics analysis. The events are usually processed one at the time so, when one is being processed, all other events which fall in the detector are, in principle, lost. These loses can be very large when the incoming rate of particles (λ) is large compared to the times involved in the process of reading (T_r) and processing (T_p) the events. These parameters depend on the type of experiment and detector. In an Astro-Particle Physics environment, for instance, typical rates are of the order of $\lambda \sim 10^3$ particles per second and times of the order of $T \sim 1$ ms. The loses are usually minimised using buffers in such a way that, when one event is being processed, a fraction of the events which fall in the detector are stored in the buffer and processed afterwards. Clearly, when $\lambda T << 1$ there is no need for a buffer. On the other hand, if λT is very large the buffer will almost always be full and thus, of no real use. The optimum buffer size depends on the characteristics of the experiment and is estimated to minimize the loses of events. However, the fraction of events which are lost has to be estimated in order to determine, for instance, the incoming flux of particles. Usually, this is estimated by a Monte Carlo simulation of the detector and the operational conditions so that subtle effects can be easily taken into account.

In this note, we present a simple approach to the problem which serves as an introductory example to the use of finite Markov Chains. We have considered an experimental situation in which the times involved in getting an event into and from the buffer are negligible compared to the processing time T_p. This is usually the case. We have assumed also that the time involved in reading the signals (T_r), during which the events can neither be stored on the buffer nor processed, can also be neglected. Depending on the complexity of the detector and the operations involved in the processing, this may not be a valid approximation. Non negligible reading times can be included in the present approach at the expense of having much more involved expressions. However, they do not modify the insights of the example and, for reasonable small values ($T_r < T_p/5$), the final results are not very different.

2 Description of the process.

We shall consider the buffer, which can store up to k_M events, as the system to be analyzed. As we process events, the number of events in the buffer will change with time. Each time we process one event, the number of events in the buffer may change. This will correspond to one step in the time evolution of the buffer. When we start processing the n^{th} event (at step n) the buffer can be at any state e_k with $k \in [0, k_M]$ events. Therefore, the space $\mathcal{E}$ of the possible states $\{e_0, e_1, \ldots, e_{k_M}\}$ of the system is of dimension $\dim(\mathcal{E}) = k_M + 1$, finite. The events are processed one at the time and each one bears a processing time T_p

during which the events falling in the detector are stored in the buffer until it is full. Once the buffer is full, the remaining events are lost. Thus, while processing the n^{th} event, the system can move from state e_j to state e_i with a transition probability $P(i|j)$ which depends on the number of events which fall during the processing time T_p. We shall assume that the events fall at a constant rate λ following a Poisson model and that the processing time is the same for all type of events (mean processing time). Therefore, the probability to receive m events at the step n (while processing the event n) is:

$$P_{oisson}(m|\mu = \lambda T) \; = \; e^{-\mu}\,\frac{\mu^m}{m!}$$

Suppose that, at the beginning of step n, the buffer is at e_0; that is, has no events stored. Then, we shall process directly the first event falling in the detector so, for $i = 0, 2, \ldots, k_M - 1$, we have that:

$$P(e_0 \rightarrow e_i) \equiv P(i|0) \; = \; P_{oisson}(i|\mu)$$

If we get k_M or more, the system will move to the state e_{k_M}, so:

$$P(e_0 \rightarrow e_{k_M}) \equiv P(k_M|0) \; = \; 1 \; - \sum_{m=0}^{k_M-1} P(m|0)$$

On the other hand, if at the step n the buffer is in state e_{j+1} we shall automatically process one event from the buffer, so the transition probabilities correspond to those from state e_j to state e_i; that is:

$$P(e_j \rightarrow e_i) \equiv P(i|j) \; = \; \begin{cases} P_{oisson}(i - j + 1|\mu) \equiv P_{i-j+1} & \\ \qquad\qquad\qquad i = j - 1, j, j + 1, \ldots, k_M - 1 & \\ \qquad\qquad\qquad j = 1, 2, \ldots, k_M & \\ 0 \quad \text{otherwise} & \end{cases}$$

Finally, the buffer will move from state e_j to e_{k_M} if we get $k_M - j$ or more events during the processing time so:

$$P(k_M|j) \; = \; 1 - \sum_{m=0}^{k_M-1} P(m|j) \qquad\qquad j = 0, 1, \ldots k_M$$

The probability to go from state e_j to e_i at the step n depends on the state of the system at the step $n - 1$ and not on the state of the system at previous steps, so the evolution is described by a finite Markov Chain. On the other hand, since the transition probabilities are independent of the step, the Markov Chain is stationary.

With these transition probabilities we can define the corresponding transition matrix **B**, of dimension $\dim(\mathbf{B}) = (k_M+1)\otimes(k_M+1)$, as:

$$\mathbf{B} \;=\; \begin{pmatrix} P(0|0) & P(1|0) & P(2|0) & \cdots & P(k_M-1|0) & P(k_M|0) \\ P(0|1) & P(1|1) & P(2|1) & \cdots & P(k_M-1|1) & P(k_M|1) \\ P(0|2) & P(1|2) & P(2|2) & \cdots & P(k_M-1|2) & P(k_M|2) \\ P(0|3) & P(1|3) & P(2|3) & \cdots & P(k_M-1|3) & P(k_M|3) \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ P(0|k_M) & P(1|k_M) & P(2|k_M) & \cdots & P(k_M-1|k_M) & P(k_M|k_M) \end{pmatrix} =$$

$$= \begin{pmatrix} P_0 & P_1 & P_2 & \cdots & P_{k_M-1} & N(P,k_M-1) \\ P_0 & P_1 & P_2 & \cdots & P_{k_M-1} & N(P,k_M-1) \\ 0 & P_0 & P_1 & \cdots & P_{k_M-2} & N(P,k_M-2) \\ 0 & 0 & P_0 & \cdots & P_{k_M-3} & N(P,k_M-3) \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & P_0 & N(P,0) \end{pmatrix}$$

with

$$N(A,k) = 1 - \sum_{m=0}^{k} A_m$$

ensuring the stochastic character of the transition matrix.

Finally, at each step in the evolution (that is, each time we process one event), the state of the system will be described by a distribution π over $\mathcal{E}$, of dimension $\dim(\pi) = \dim(\mathcal{E}) = k_M + 1$, where π_i denotes the probability for the system to be in state e_i. At the beginning of the process, when we start the data taking period, the buffer is empty so the system is in the state e_0 with probability 1 and $\pi^{(0)} = (1,0,0,\dots,0)$. At the end of the step n (or beginning of the step $n+1$), the status of system will be described by:

$$\pi^{(n)} \;=\; \pi^{(n-1)}\,\mathbf{B} \;=\; (\pi^{(n-2)}\mathbf{B})\,\mathbf{B} \;=\; \dots \;=\; \pi^{(0)}\,\mathbf{B}^n$$

3 Characteristics of the Markov Chain.

The main features of the process can be obtained from an inspection of the transition matrix. In this section we shall describe them briefly.

Since

$$\mathbf{B}_{i,j} \;\neq\; 0 \qquad \forall\, i,\; j = i-2,\dots,k_M+1$$

we can go from any state to any other in a finite number of steps. Thus, all the states communicate. Since the communication between states is a relation of equivalence, all the possible states of the system belong to the same class of equivalence and, therefore, the stationary Markov Chain is irreducible.

In general, the states which belong to the same class of equivalence are either recurrent (if starting in state e_i, the probability that the the system will pass again by state e_i is one), or transient (if the aforementioned probability is less than one). If the Markov Chain is finite, all recurrent states are positive recurrent; that is, starting in state e_i, the expected time until the system returns to state e_i is finite. Of course, we can have also transient states in finite Markov Chains (which belong to a separate class of equivalence) but, for finite irreducible Markov Chains, all states are recurrent and, therefore, positive recurrent. Being the Markov Chain irreducible and all the states positive recurrent, there is a unique stationary distribution $\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{B}$, so:

$$\boldsymbol{\pi}\,\mathbf{B}^n = (\boldsymbol{\pi}\,\mathbf{B})\,\mathbf{B}^{n-1} = \boldsymbol{\pi}\,\mathbf{B}^{n-1} = \ldots = \boldsymbol{\pi}$$

Last, $\mathbf{B}_{ii} \neq 0 \;\forall\, i = 1, 2, \ldots k_M + 1$ so, starting at state e_i, the system can go back to e_i at any step of the evolution. Thus, all the states are aperiodic (property shared by all the states which belong to the same class of equivalence). Since all the states are positive recurrent and aperiodic, the Markov Chain is ergodic having, among others, the following properties:

1)

$$\lim_{n\to\infty}\mathbf{B}^n = \begin{pmatrix} \pi_0 & \pi_1 & \cdots & \pi_{k_M-1} & \pi_{k_M} \\ \pi_0 & \pi_1 & \cdots & \pi_{k_M-1} & \pi_{k_M} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \pi_0 & \pi_1 & \cdots & \pi_{k_M-1} & \pi_{k_M} \end{pmatrix}$$

2)

$$\lim_{n\to\infty}(\mathbf{B}^n)_{ii} = \pi_i / E[N_{ii}]$$

with $E[N_{ii}]$ the mean number of steps to go back to state e_i starting from e_i.

Therefore, asymptotically the buffer will be in a state described by the stationary distribution $\boldsymbol{\pi}$ whose elements represent the probability to find the buffer in a particular state e_i. They are easily obtained solving the linear system $\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{B}$. In particular, since all the components π_i are proportional to π_0, a simple way could be to use a recursive procedure:

$$\pi_0\,P_0 = \pi_{n-1} - \pi_0\,P_{n-1} - \theta(n-1)\sum_{m=1}^{n-1}\pi_{n-m}\,P_m$$

for $n = 1, 2, \ldots k_M$ and obtain π_0 from the normalisation condition:

$$\pi_0 + \pi_1 + \ldots + \pi_{k_M} = 1$$

For large buffer sizes, this becomes a lengthy task, but since π_0 is always different from 0, we can fix it to any value in $(0, 1)$, implement the above procedure and scale all components to match the normalisation condition.

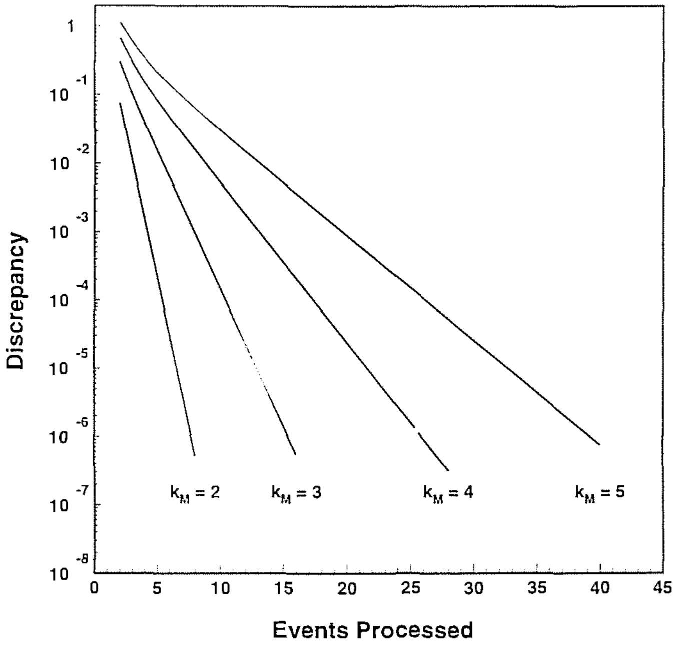

Fig. 1: Discrepancy δ between the distribution of the buffer and the asymptotic distribution as function of the number of events processed for $\mu = \lambda T_p = 1$ and buffer sizes of 2,3,4 and 5.

4 Estimation of the fraction of events lost.

As the system evolves during the event processing, the probability distribution which describes the buffer tends to the asymptotic distribution. In practice, after few events are processed, the discrepancy between the actual distribution ($\pi^{n)}$) and the asymptotic one (π) is very small. To quantify this difference we can use, for instance, the Kullback-Leibler measure of discrepancy ($\delta\{\mathbf{p},\mathbf{q}\}$) between two probability distributions. Considering π as an approximation to $\pi^{n)}$, we have that:

$$\delta\{\pi,\pi^{n)}\} = \sum_{i=0}^{k_M} \pi_i^{n)} \log \pi_i^{n)}/\pi_i$$

Relative differences between π_i and $\pi_i^{n)}$ of about 1 per mil would give values of $\delta\{\pi,\pi^{n)}\}$ around 10^{-3}. The discrepancy is shown in fig. 1 as function of the number of steps (events processed) for buffer sizes of $k_M = 2,3,4,5$ and and a value of $\mu = \lambda T_p = 1$.

Usually, data taking periods are very long so a large number of events are processed and we can safely consider the asymptotic distribution to describe the

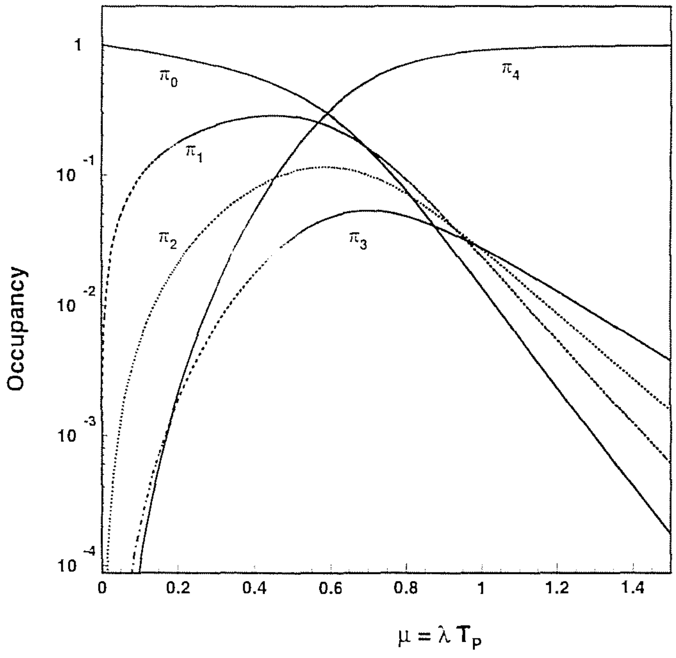

Fig. 2: Buffer occupancy as function of $\mu = \lambda T_p$ for a buffer of size $k_M = 4$.

buffer after few steps. Thus, at the step n of the evolution, we have a probability π_k to find the buffer at the state e_k; that is, with k events. The dependence of these probabilities (buffer occupancy) with the incoming rate is illustrated in fig. 2 for a buffer of size $k_M = 4$ and $\mu = \lambda T_p = 1$. As it is expected, for low incoming rates it is more likely to find the buffer empty when we start processing an event. On the other hand, when the rate is large ($\lambda T_p \geq 1$) the buffer is full most of the times.

Consider, for instance, that the buffer is at the state e_k with $k \geq 1$. Then, one of the events from the buffer is sent to be processed so, at the beginning, the buffer moves to state e_{k-1} having room for $k_M - (k-1)$ new events to be collected while one is being processed. All events we get in excess to those are lost since they can not be stored in the buffer. Thus, we shall lose m events if we get $k_M - (k-1) + m$ events, which will happen with probability:

$$P_{oisson}(k_M - (k-1) + m|\mu)$$

Then, if at the n_{th} step the buffer is at the state e_k, the average number of events

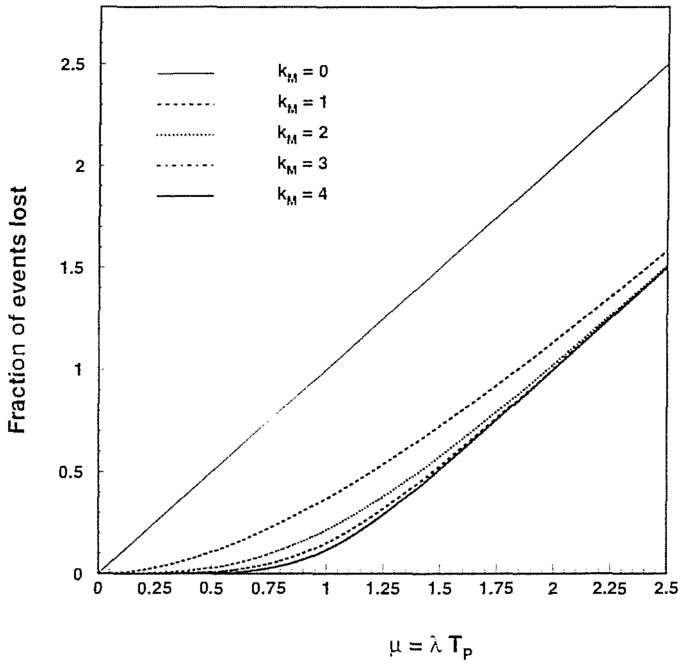

Fig. 3: Fraction of events lost as function of $\mu = \lambda T_p$ for buffer sizes of $k_M = 0, 1, 2, 3$ and 4.

we shall lose is:

$$\begin{aligned} n_L(k) &= \sum_{m=0}^{\infty} m\, P_{oisson}(m + (k_M - k + 1)|\mu) = \\ &= \mu - (k_M - k + 1) + e^{-\mu} \sum_{j=0}^{k_M - k} (k_M - k + 1 - j)\, \mu^j / j! \end{aligned}$$

Clearly, for the state e_0 we have that $n_L(0) = n_L(1)$. Therefore, the average number of events lost for each processed one is:

$$n_L = \sum_{k=0}^{k_M} n_L(k)\, \pi_k$$

so, after processing N_p events, we lost a fraction $f_{k_M} = n_L$ of them with a buffer of size k_M.

It is straight forward to implement the previous procedure on a computer. However, for small buffer sizes, compact expressions can be obtained. For instance, for sizes up to $k_M = 4$ we obtain:

$$f_1 = \mu - 1 + P_0$$

$$
\begin{aligned}
f_2 &= \mu - 1 + \frac{P_0^2}{1-P_1} \\
f_3 &= \mu - 1 + \frac{P_0^3}{(1-P_1)^2 - P_0 P_2} \\
f_4 &= \mu - 1 + \frac{P_0^4}{(1-P_1)^3 - 2(1-P_1)P_0 P_2 - P_0^2 P_3}
\end{aligned}
$$

with $P_n = P_{oisson}(n|\mu)$. The fraction of events lost is shown in fig. 3 as function of the incoming rate for $\mu = 1$ buffer sizes of$k_M = 0, 1, 2, 3$ and 4.