Single-machine scheduling of deteriorating or delaying
stochastic jobs with dynamic priority.

DERBALA Ali *, **

*) University of Blida, ALGERIA. Faculty of Sciences, Mathematics Department. Bloc 14. BP 270, Route de Soumaa, Blida -ALGERIA. Tel & Fax: (+) 213 - 3 - 43 - 36- 42. E-mail : aliderbala@hotmail.com

**) Part of this research was conducted at the university François Rabelais, E3I, Computer Science Laboratory. 64, Avenue Jean Portalis. 37200 Tours, FRANCE.

Abstract: We consider a scheduling rule, called OPBM, which prioritizes jobs by an index computed for each job as a weighted sum of the proportion of time it has been processed and the proportion of time it has been waiting for processing. We prove that it is a well known deteriorating jobs or delaying jobs single-machine scheduling problems. An expression of this dynamic priority is interpreted as a net-reduction in the processing of a job and an other one as a delay caused or a time to repair a breakdown. Concerning the behavior of this scheduling scheme, two results are stated. OPBM is shown to be a deterioration function which yield to an index policy. The processor or the machine is not supposed to be subject to breakdown, jobs are independent and the preemption is allowed. An implementation of this priority is presented.

Keywords: completion (or Hazard) rate, mean residual lifetime, stochastic scheduling, index rule and OPBM.

1. INTRODUCTION

We present a scheduling rule, called OPBM, which prioritizes jobs by an index computed for each job as a weighted sum of the proportion of time it has been processed and the proportion of time it has been waiting for processing. Our motivation for studying these priorities is to answer a question raised in Browne and Yechiali [1990]. They were inquiring whether any deterioration functions yield to an index policy. OPBM is a deterioration function which yield to an index policy.
It has been shown in Haro and Proust [1992] that schedulers, some of which are employed by E3I’s operating system, at the Computer Science Laboratory at the university of Tours, in France, operate as hyperbolic schedulers. They defined this particular priority and implemented it in the kernel of an operating system. Multi-programmed system designers often implement priority disciplines expecting to improve system performances.

They showed that the choice of the bounds allow to derive some of the known strategies and prove that it makes the schedule equitable and its implementation still maintains this property.

They indicated and gave some efficient hints of the associated algorithm and an adaptation to obtain an equitable sharing scheduler is introduced. The processor or the machine is not supposed to be subject to breakdown, jobs are independent and the preemption is allowed.

N jobs are to be processed sequentially on a single machine. They are supposed to be within the shop at the time 0. With each job « i » is associated an instantaneous priority

\[ p_i(t) = \frac{M_i \cdot t_i^w(t) + m_i \cdot t_i^p(t)}{t} \quad \text{for} \quad t = 1, 2, \ldots \quad \text{and} \quad p_i(0) \in [m_i, M_i] \]  

(*)

where \(m_i\) and \(M_i\) are real-values. \(t_i^p(t)\) and \(t_i^w(t)\) are respectively the time spent processing job \(i\) and the time spent waiting for processing during \([0,t]\). At each time \(t\), we carry out the job with the highest priority. This means that when a job is to be selected among those waiting, the one with the highest priority is selected. We break the tie with Round-Robin rule. Scheduling jobs according to the formula (*) is said OPBM rule. This means in French « Ordonnancement par priorités bornées en Moyenne ».

The remainder of this paper is organized as follows: section 2 is devoted to the definition of Bandit processes. This tool will be used to prove that OPBM is a Gittins rule. The OPBM model is shown to be a deteriorating or delaying jobs scheduling problem. Concerning the behavior of these priorities, two results are stated in section 3. In section 4, an implementation of these priorities is given. Finally a summary and further work are provided in section 5.

2. BANDIT PROCESSES
A theoretical approach to stochastic scheduling which is entirely distinct from queuing theory recently has been developed by Gittins and his colleagues. Their method assigns a dynamic allocation index to each job and then schedules the jobs in decreasing order of this index. These indices are updated as the jobs are processed, thus allowing the schedule to be adapted to the actual pattern of arrivals and processing times. Papers of importance are Gittins [1979, 1989], Gittins and Glazebrook [Git77] and Whittle [Whi80].

A bandit process is a Markov decision process \((\Omega, U, P, C, a)\) where:

- \(\Omega\) is the state space of the process and may be finite, countable or continuous.
- \(U\) is the decision set, consisting of just two elements \(c\) or \(1\) (continue) and \(w\) or \(0\) (wait).
- \(P\) is a probabilistic law of motion; that is, for each \(x \in \Omega\) and \(u \in U\), \(P(x,u)\) defines a probability distribution over some \(\sigma\)-field of subsets of \(\Omega\). Thus if \(x \in \Omega\) and the state of the process at time \(t\) is \(x(t)\), when decision \(u(t)\) is applied \(Pr\left[ x(t+1) \in X \mid x(t), u(t) \right]\) is well defined by

\[
\int_{x} dP(x(t), u(t)).
\]

Particularly \(\forall A \in P(\Omega), Pr[ A/ x \mid u \ ]\) is the probability that the state \(y\) of the process immediately after the time \(t\) belongs to \(A\) conditioning that at time \(t\) the process was at state \(x\) and the decision \(u\) (\(u \in \Omega(x)\)) was applied.

- \(C\) is a cost function from \(\Omega \times \Omega\) in \(R\).
- \(a\) the discount factor, a real number \((0 \leq a \leq 1)\).

A transition at time \(t\) from \(x(t)\) to \(x(t+1)\) involves an incremental cost \(a' \ c[ x(t), x(t+1) ]\).

The total cost accruing if the process describes a trajectory \(\{ x(t) : t = 0, 1, ..\} \) is

\[
\sum_{t \geq 0} a' \ c[ x(t), x(t+1) ].
\]

\(P\) and \(C\) must satisfy the following conditions:

\[
Pr\left[ x(t+1) = x(t) \mid u(t) = 0 \right] = 1, \forall t, x(t) \in \Omega. c[ x, x ] = 0, \forall x \in \Omega, \forall t.
\]

This definition may be extended to the case of semi-Markov decision processes. For some authors these processes are also called Adaptative control processes.

A family of alternative bandit processes is a set of bandit processes.
\{(\Omega_i, U, P_i, C_i, a): 1 < i < n\} \text{ such that at each time } t \ (t = 0, 1, 2, \ldots), \ \text{the decision element } "c"

(or 1) is applied to just one process \(i\), \(w\) being applied to the others.

The factor discount "a" is the same for all the processes.

A simple family of alternative bandit processes is a set of alternative bandit processes which are all always included in the decision set. It is denoted by SFABP. In stochastic scheduling problems, at time \(t\), the time spent processing job \(i\) models such family. The problem is to find an optimal strategy for the decision of a simple family of alternative bandit processes (i.e. a strategy for choosing \(i\) at each time \(t\)), in the sense of leading to minimal total expected cost. We suppose that the set of states is countable. After observing the state of the process an decision is chosen from the set \(U\). If the process is at state \("i\) at time \(t\) and the decision \("a\) is chosen, then a cost \(c(i, a)\) is incurred and the next state is chosen with probability \(p_{ij}(a)\).

GITTINS and JONES [1972] gave the first existence theorem of these priorities called dynamic allocation indices denoted Dai’s and called by Whittle[1980], GITTINS’ indices. These indices represent a ratio of an expected cost over an expected elapsed time in the workshop. They also represent a stopping cost. Nash [1973] was the first who gave a characterization of these indices.

Gittins and Glazebrook[1977], Gittins[1979] etc... have contributed in this field.

**Theorem GJ[1972]**

Given a family of alternative bandit processes \{ (\Omega_i, U, P_i, C_i, a): 1 < i < n\}

there exist functions \{ \nu_i : \Omega_i \rightarrow \mathbb{R}, 1 \leq i \leq n\} such that the choice \(u_i(t) = 1\) is optimal when the states of the processes at time \(t\) are \(x_i(t)\) \((1 \leq i \leq n)\) if and only if

\[
\nu_i(x_i(t)) = \min_{1 \leq j \leq n} \nu_j(x_j(t)).
\]

Provided that for the set of strategies such that this optimal strategy is applied from some time \(t\) onwards, with the possible exception of a single decision-time, the contribution to the expected loss which arises after time \(t\) tends uniformly to zero as \(t\) tends to infinity.

The characterization of these indices \(\nu_i\) \((1 \leq i \leq n)\) is given by the result below.

**Theorem N : [1973]**
If $\nu_i(x_i)$ is defined and finite for all $x_i \in \Omega_i$, then $\nu_i(x_i)$ is the unique finite solution of the two equations:

$$
\nu_i(x_i) = \frac{\sum_{t=0}^{t-1} a^t C_i[x_i(t), x_i(t+1)]}{1 - E a^t} \quad (N_1)
$$

and $t^* = \min \{ t \mid \nu_i(x_i(t)) \geq \nu_i(x_i) \} \quad (N_2)$ and where $x_i = x_i(0)$

A rule of scheduling jobs according to the theorem of Gittins and Jones [1972] and the theorem of Nash [1973] is called an index rule.

3. THE MODEL OPBM : A DETERIORATING OR DELAYING PROBLEM

OPBM scheduling rule can be modeled as two well-known problems. It can be expressed as a problem of deteriorating jobs and jobs subject to delay.

3.1 OPBM : A single-machine scheduling problem of deteriorating stochastic jobs

We call jobs whose processing times increase over time deteriorating jobs.

At each time $t$, $t_i^p(t) + t_i^w(t) = t$. The formula (*) become, for all $t$, $t = 1, 2, ...$

$$
p_i(t) = M_i + (m_i - M_i) \frac{i^p(t)}{t} = M_i - \Delta_i \frac{i^p(t)}{t}
$$

Hence $A_i(t) = \frac{t p_i(t)}{m_i} = (1 + \alpha_i) \frac{t_i^p(t) - \alpha_i}{t} = t_i^p(t) - \alpha_i(1 - t_i^p(t))$

Where $\alpha_i = -\frac{M_i}{m_i} \in \mathbb{R}^+$. We suppose that $P_i$ is the processing requirement of job $i$ at $t = 0$. For all $i$, it is assumed to be independent of $P_j$, $i \neq j$.

The constant $\alpha_i = -\frac{M_i}{m_i}$ is the deterioration rate of job $i$. If job $i$ is not yet complete, its residual processing grows at rate $\alpha_i$.
$\alpha_i (t - t_i, p(t))$ is the increase in i’s residual processing as a result of processing given to other jobs during $[0, t]$.

$A_i(t) = t \frac{p_i(t)}{m_i}$ can be interpreted as a « Net reduction » in job i’s residual processing as a consequence of all processing during $[0,t]$, where $p_i(t)$ is its priority.

This model is used in many industrial applications, for instance in the planning of machine maintenance or service and in steel production where the material will cool during the waiting periods.

Inventory model for deteriorating items with instantaneous supply, a perishable product, linear or non-linear increasing demand and shortage. Items deteriorate while in storage. Decay in radioactive elements, spoilage in food grain storage, pilferage from on-hand inventory. In scheduling steel rolling mills, if the temperature of an ingot, while waiting in a buffer between the furnace and the rolling machine, has dropped below a certain temperature, then the ingot needs to be reheated to bring it up to the temperature required for rolling. The re-heating time of the ingot depends on its waiting time in the buffer and is thus schedule-dependent.

Assume that time evolves in discrete epoch of $N$, the set of integer-values.

OPBM belongs to the class of preemptive linear deterioration models in which the processing requirement of a job grows linearly as it awaits processing $i$. It can be considered as a model which belongs to the class of Markov decision processes where:

1) $U = \{c, w\}$ the set of the decisions

2) Jobs : The jobs are individual .

3) Decision , decision epochs : The time evolves in discrete steps of size1. Job completion’s will not necessarily occurs at integer time points. The deterioration phenomenon means that an increase in processing requirements experienced by jobs will not necessarily an integer.

If $t$ is a decision time, the next decision for an uncompleted job chosen for processing at $t$ will be $t + 1$ or the time of j’s completion , whichever occurs sooner.

4) State and state transitions. We define a state of job $i$ at decision time $t$ by :

$$X_i(t) = \begin{cases} C \text{ ( completion )} & \text{if i has been completed by t} \\ \{ A_i(t), B_i(t) \} & \text{otherwise} \end{cases}$$
\[ B_i(t) = \max_{s \in D_i} A_i(t) = \max_{s \in D_i} t \frac{p_i(t)}{m_i} \] where \( D_i = \{ 0, 1, ..., t \} \).

The state of the process at \( t \) is \( X(t) = \{ X_1(t), ..., X_N(t) \} \).

5) Policies: The preemptive policy OPBM will be used for choosing an uncompleted job for processing at each decision epoch. We break the tie with Round Robin rule.

6) Cost structures: The criterion assumes a discount factor \( \alpha, \alpha \in [0,1] \) and attempts to minimize the expected total discounted cost. For any trajectory \( \{ x_t, t \in \mathbb{N} \} \) and for any policy \( \pi \), we define:

\[ V_\pi(i) = E_{\pi} \left[ \sum_{t=0}^{\infty} \alpha^t c(x_t, a_t) / x_0 = i \right], \quad i \geq 0. \]

\( E_{\pi} \) represents the conditional expectation given that the policy \( \pi \) is employed.

We assume that \( c(x_t, a_t) \) are bounded and \( \alpha < 1 \). \( V_\pi(i) \) represents the expected total discounted cost incurred when the policy \( \pi \) is employed and the initial state is \( i \).

3.2 OPBM: A single-machine preemptive scheduling problem of stochastic jobs subject to delay.

From the definition of OPBM,

\[ t_{w}^{i}(t) = \frac{t}{\Delta_i} \left[ P_i(t) - M_i \right] \] and \( t_{p}^{i}(t) = \frac{t}{\Delta_i} \left[ M_i - P_i(t) \right] \) are respectively the time spent waiting for processing and the time spent processing job \( i \) during \([0,t]\).

Each job \( i \) has a processing requirement \( P_i \) at time 0 which is independent of the \( P_j, j \neq i \).

Let us define a process \( Y(t) = (Y_1(t), ..., Y_N(t)) \),

with \( Y_i(t) = (X_i(t), D_i(t), Z_i(t)) \), where \( X_i(t) = t_{p}^{i}(t) / N, D_i(t) = t_{w}^{i}(t) / N \) and
\( Z_i(t) \) is an indicator variable, which takes the value 1 if job \( i \) has completed prior to \( t \) and is otherwise 0. Also we have \( X_i(0) = D_i(0) = Z_i(0) = 0, 1 \leq i \leq N \).

The stochastic process \( \{ D_i(t) = t_i^w(t) / N, t \in \mathbb{N} \} \) is called the delay process.

If job \( i \) is given a single unit of processing at time \( t \), having received \( x \) units of processing prior to \( t \), then no further processing can take place until time \( t + 1 + D_i(x + 1) - D_i(x) \).

\[
D_i(x + 1) - D_i(x) = \frac{1}{N} \left[ t_i^w(x + 1) - t_i^w(x) \right] = \\
= \frac{1}{N} \left\{ \frac{x_i^w(x + 1)}{\Delta_i} \left[ P_i(x + 1) - P_i(x) \right] + \frac{1}{\Delta_i} \left[ P_i(x + 1) - m_i \right] \right\}
\]

Hence \( D_i(x + 1) - D_i(x) \) is interpreted as the delay caused by the \( (x + 1) \) st unit of processing received by job \( i \) or the time to repair a breakdown. This scheduling problem is a Markov breakdown process.

Optimal policies are determined by Gittins index when discounted rewards are earned at job completion times. The same cost structure is defined as above.

Our goal is to prove that OPBM maximizes the total reward over all policies. It is equivalent to show that it is a Gittins index rule.

The delayed-jobs model is a SFABP with the following features:

**States:** the state of the system at time \( t \) is \( Y(t) = (Y_1(t), ..., Y_N(t)) \),

with \( Y_i(t) = (X_i(t), D_i(t), Z_i(t)) \), where \( X_i(t) = t_i^p(t) / N, D_i(t) = t_i^w(t) / N \) and \( Z_i(t) \) is an indicator variable, which takes the value 1 if job \( i \) has completed prior to \( t \) and is otherwise 0. Also we have \( X_i(0) = D_i(0) = Z_i(0) = 0, 1 \leq i \leq N \).

**Decisions:** At each decision epoch, one of the uncompleted jobs is chosen for processing.

**State transitions:** The state of all jobs other than \( i \) remain unchanged through a transition.

If job \( i \) is continued at time \( t \), with state \( Y_i(t) = (X_i(t), D_i(t), Z_i(t)) = (x, d, 0) \), a state transition will be observed where \( X_i'(t) = x + 1 \) w.p.1;

\[
p(D_i' = d') = p(D_i(x + 1) - D_i(x) = d' - d / D_i(x) = d)
\]
\[
p(Z'_i = 1) = p(P_i = x + 1 / P_i > x) = \rho_i(x), \text{ where } \rho_i(x) \text{ is the completion rate (hazard rate) of job } i\text{'s processing requirement. The next decision epoch will occur at time}
\]
\[
t + 1 + (D'_i - d).
\]
\[
\text{Note that } \sum_{i=1}^{N} \left[ X_i(t) + D_i(t) \right] = t.
\]

**Reward**: If \( t \) is a decision epoch which leads to \( i \)'s completion, this completion is deemed to be at \( t + 1 + (D'_i - d) \). A reward of \( r_i \) is received then, discounted at rate \( \alpha \), that is a reward of
\[
r_i \alpha^{t + 1 + D'_i - d}.
\]

**Policies**: the rule for choosing an uncompleted job is OPBM. We break a tie with Round Robin rule.

Optimal policies are determined by a collection of indices called Gittins indices.

We give below two propositions which describe the behavior of our priority.

The proofs are not presented in this paper.

**Proposition 1**: If job \( j \) is not complete at decision epoch \( t \)

\[
\text{then its priority } p_j(s) < \frac{m_j}{s} p_j, \quad \forall \ s \in [0, t].
\]

Suppose now that job \( j \) is chosen for processing at \( t \) and that the next decision epoch occurs at \( t + \sigma \), where \( \sigma \leq 1 \). As a consequence of this processing, uncompleted job \( k \neq j \) will deteriorate, its processing requirement growing linearly at rate \( \alpha_k \).

**Proposition 2**: An uncompleted job \( k \), not being continue at \( t \) if the next decision epoch occurs at \( t + \sigma \), \( \sigma \leq 1 \) then :

\[
p_k(t + \sigma) = \frac{t}{t+\sigma} p_k(t) \cdot \alpha_k \cdot m_k \cdot \frac{1}{t+\sigma}.
\]

For job \( j \) undergoing processing at \( t \) and cannot be completed with an additional unit of processing then

\[
p_j(t + 1) = \frac{t}{t+1} p_j(t) + \frac{m_j}{t+1}.
\]

We assert that OPBM is an index rule.
**Theorem 3**: OPBM is an index rule or a Gittins index rule and \( P_i(t) = M_i - \Delta_i \frac{t_i^p(t)}{t} \) is a dynamic allocation index denoted DAI.

We can notice that all the results established in the theory of Bandit Processes and Gittins’s indices still hold.

### 4. IMPLEMENTATIONS

We have implemented these priorities. It provides an asymptotic equitability in the service.

Let \( E = \{ \tau_i / 1 \leq i \leq n \} \) a set of \( n \) jobs.

The occupation rate of the processor (or the occupancy of the processor) by the \( n \) processes of \( (E) \) is the real number \( \rho \) defined by:

\[
\forall t > 0 \quad \rho(t) = \frac{\sum_{i=1}^{n} t_i^p(t)}{t}.
\]

The occupation rate of process by any process \( \tau_i \) of \( (E) \) is also the real number \( \rho_i(t) \) defined by:

\[
\forall i, 1 \leq i \leq n \quad \forall t > 0 \quad \rho_i(t) = \frac{t_i^p(t)}{t}.
\]

It is the ratio of \( t_i^p(t) \) over \( t > 0 \), At epoch \( t \), \( \tau_i \) is carried out during \( t_i^p(t) \).

Scheduling is equitable for the set of processes \( E = \{ \tau_i / 1 \leq i \leq n \} \) when

\[
\exists \lambda, 0 \leq \lambda \leq 1 : \forall i, 1 \leq i \leq n \quad \forall t > 0 \quad \rho_i(t) = \lambda / n
\]

Scheduling is asymptotically equitable for \( E = \{ \tau_i / 1 \leq i \leq n \} \) if:

\[
\exists \beta, 0 \leq \beta \leq 1 : \forall i, 1 \leq i \leq n : \lim_{t \to \infty} \rho_i(t) = \frac{\beta}{n}.
\]

During a period \( T_s \), the processor carries out one job of the set \( E \). Whenever a tie for the highest priority occurs, the Round-Robin strategy is adopted. Each of the jobs with the highest priority receive a process quantum \( Q \), though \( « m » \) is a lower bound and \( « M » \) is an upper bound for the priority.

This implementation takes account the whole past history of the process to compute again its priority.

The computing is made each \( T_s \) time units by the formulae:

\[
\forall i, 1 \leq i \leq n \quad \forall j > 0 : P_i(j.T_s) = M - \frac{\Delta P_{i,j}}{jT_s} t e_i(j - 1)
\]

The processing times at epoch \( (j.T_s) \) of each job are computed by:
∀ i ,1 ≤ i ≤ n and ∀ j >0 :

\[ t_{ei}(j.Ts) = t_{ei}(j-1).Ts + Ts \], only if job i has the highest priority

\[ t_{ei}(j.Ts) = t_{ei}(j-1).Ts + Q \], if job i is one of the jobs with the highest priority and is in tie.

\[ t_{ei}(j.Ts) = t_{ei}(j-1).Ts \], Else.

For each job, the occupation rate of the processor is given by:

∀ i , 1 ≤ i ≤ n and ∀ j >0 :

\[ \rho_{i}(j.Ts) = \frac{t_{ei}(j.Ts)}{j.Ts} \]

Example: (Rahal et al [1996]) Within the system, it is assumed that the number of jobs is two. The duration of the period Ts is one (1), the quantum value is one. The priorities may vary in the interval [1, 10] (i.e. m = 1; M = 10). The initial priorities are \( p_{1}(0) = 1 \) and \( p_{2}(0) = 10 \).

We schedule the jobs during 100 sec. We have plotted the behavior of the priorities and the occupation rate (occupancy of the processor). See figure 1 and 2.

![Figure 1: Job’s Priorities evolution](image)

It can be shown graphically that priorities converge to a common value \( P/ n \) where
P = \sum_{i=1}^{n} p_i(t) \quad \text{and} \quad n \quad \text{the number of processes in the system.}

Figure 2: Processor Occupation Rate’s Evolution

From this graphic, we see that the occupation rate also converges to a constant value given by \lambda/n.

5. SUMMARY

We considered a scheduling rule, called OPBM, which prioritizes jobs by an index computed for each job as a weighted sum of the proportion of time it has been processed and the proportion of time it has been waiting for processing. We show that jobs are deteriorating and delaying, in the sense that processing times increase while they are waiting for processing. An expression of this dynamic priority is interpreted as a net-reduction in the processing of a job and another one as a delay caused or a time to repair a breakdown. As far as the behavior of this extended new scheduling scheme, we stated two results. We proved that OPBM is equivalent to a Gittins index scheduling rule or an index rule. This paper answer a question raised in Browne and Yechiali[1990]. They were inquiring whether any deterioration functions yield an index policy.

ACKNOWLEDGMENT : I am very grateful to Professor Christian Proust of the computer science Laboratory of E3I of the University of Tours in France, where a part of this research was conducted,
for his interest and guidance in conducting the research and for Dr Christophe Haro for his help and from which this material is drawn. Thanks are due to Dr Zouleikha Zaidi for her help.

REFERENCES:


GITTINS, J.C and JONES, D.M(1972). A dynamic index for the sequential design of experiments, 9, European meeting of statisticians, BUDAPEST( Hungary), pp.241-266.


Cambridge University.
