# Stochastic scheduling of single forest fire-fighting processor

<table>
<thead>
<tr>
<th>Journal:</th>
<th>Canadian Journal of Forest Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manuscript ID</td>
<td>cjfr-2015-0294.R1</td>
</tr>
<tr>
<td>Manuscript Type:</td>
<td>Article</td>
</tr>
<tr>
<td>Date Submitted by the Author:</td>
<td>18-Nov-2015</td>
</tr>
<tr>
<td>Complete List of Authors:</td>
<td>Kali, Abdesselem; University Djilali Bounaama</td>
</tr>
<tr>
<td>Keyword:</td>
<td>Forest fires management, Scheduling, Single forest fire-fighting processor, Markovian model, Dynamic Allocation Index rule</td>
</tr>
</tbody>
</table>
Full Title: Stochastic scheduling of single forest fire-fighting processor

Author: Abdesselem Kali

Institution: University Djilali Bounaama, Khemis-Miliana, Ain-Defla, Algeria

E-mail: abdesselem.kali@gmail.com

Telephone number: 00213 555435606
Abstract.

In forest fire-fighting, the longer the fires wait the larger they grow and the longer they take to control. This study concerns the optimal deployment of single forest suppression processor of initial attack in the case of fires ignited simultaneously. The aim is to minimize the total damage caused by the fires to the burnt areas when all fires are suppressed. In "Scheduling fire-fighting tasks using the concept of deteriorating jobs. Can. J. For. Res. 36: 652-658, (2006)", Rachaniotis and Pappis use the concept of start-time dependent job processing-times for modelling the time needed for fire suppression. The model is intricate but interesting in the sense that is based on theoretical and empirical research in the field of forest fire-fighting. As a continuation, a stochastic formulation which includes unpredicted parameters in the proposed model is considered, and "Dynamic Allocation Index" rule is used to solve the problem. The optimality of this rule and its effectiveness are proven. Experimental results depict the framework, inside it the forest suppression processor achieves greater efficiency.

Keywords. Scheduling. Single forest fire-fighting processor. Markovian model. Dynamic Allocation Index rule.
Résumé.

Dans la lutte contre les feux de forêt, plus en tard plus les feux propagent et prennent du temps pour leur confinement. Cette étude concerne le déploiement optimal d’un processeur de suppression de la première intervention, dans le cas des incendies enflammés simultanément. L'objectif est de minimiser les dommages dans les zones brûlées lorsque tous les feux sont supprimés. Dans "Scheduling fire-fighting tasks using the concept of deteriorating jobs. Can. J. For. Res. 36: 652-658, (2006)", Rachaniotis et Pappis montrent que la durée opératoire du processeur dans l'extinction des incendies de forêt est dépendante du temps (détériorante). Le modèle est difficile à utiliser mais intéressant dans le sens qu’il est basé sur la recherche théorique et empirique dans le domaine de la lutte contre les feux de forêt. Dans cet article, une formulation stochastique qui inclut les paramètres imprévus dans le modèle est considérée, et la règle "d'Indice d'Allocation Dynamique" est utilisée pour résoudre le problème. L'optimalité de cette règle et son efficacité ont fait leurs preuves. Les résultats expérimentaux déterminent le cadre, à l'intérieur le processeur atteint une plus grande efficacité.

Mots-clés. Ordonnancement. Un seul processeur de la lutte contre les feux de forêt· Modèle Markovien· Règle d'Indice d'Allocation Dynamique.
1. Introduction

Fire is a natural component of many forest ecosystems but it also poses significant threats to people and forest resources. The effectiveness of forest fire-fighting depends, in large part, on the early detection of fire's ignition, on the accurate prediction of fire behaviour, and on the rapid commitment of resources for intervention. The approaches to modelling forest fire have so far involved mathematical analysis of physico-chemical phenomena, laboratory tests, and observation of real cases of forest fire (Sullivan 2009a; Sullivan 2009b; Sullivan 2009c; Taylor et al. 2013).

Scheduling, a field of Operational Research, supports decisions for preventive mobilization of resources to contain fires in real time. A review of operations research contributions made in forest fire management can be found in (Minas et al. 2012; Weintraub et al. 2007).

The problem of scheduling exists whenever there is a choice as to the order in which a number of tasks can be performed (Conway et al. 1967). For our particular interest, the problem description starts with a forest fire agency (a shop), and a set of fires (jobs). The forest fire agency is completely described by giving the number and different types of fire-fighting resources of initial attack (machines), namely: fire duty officers, trucks, fixed wing air tankers, helicopters, etc.

Fires can ignite either simultaneously or at different times. We can limit our attention to the problem of simultaneously igniting fires which appears quite often in Mediterranean countries (Vorisis 1999).

The relevant attributes of a fire that are given as part of the problem description are: processing-time (the amount of time required to control the fire), release-time (the time at which fire
suppression can be started) and deadline (the time at which fire can take a potentially destructive
behaviour and merit aggressive attack).

There are scheduling models in which single-machine is considered. From a practical point of
view, direct applications of this exception in forest fire-fighting are more frequent. Some fire-
fighting processors must operate separately or they can be viewed as a single processor if they must
operate commonly.

Undeniably, the variability of fire processing-time as other attributes is inextricably involved.

There are various ways of regarding this variability. One can consider the processing-time of a
fire as a random variable or a fuzzy number unidentified in advance.

Based on the characteristics of the area where fire is ignited and on the characteristics of the
processor of initial attack, Rachaniotis and Pappis (2006) propose a model in which the
processing-time of a fire increases if the beginning of fire containment effort is delayed. This
fact is known in scheduling field as "deteriorating jobs".

With the objective to maximize the total value of the burnt areas remaining after the completion
of the containment operation, a branch and bound algorithm was presented in (Pappis and
Rachaniotis 2009). In the case where the objective is to minimize the total weighted tardiness, a
lower bound for the problem is given and a heuristic algorithm is used for the extraction of an
upper bound of the solution (Rachaniotis and Pappis 2011). Unfortunately, the proposed branch
and bound algorithms can yield optimal results for a moderate size of fires. The intricate
formulation of the model is among their basic deficiencies. In contrast, the deteriorating model
may help to give an insight and stimulation for future, more realistic approximations of real
forest management problems that are expected to be encountered (Rachaniotis and Pappis 2011).
A stochastic approach that captures uncertainty and enables a more accurate description of the decision making process may be adopted. Markov decision processes are a widely used tool to model decision making under uncertainty. In the paper, a Markovian decision process for fire processing-times evaluation based on the deteriorating model given in (Rachaniotis and Pappis 2006) is considered. The objective is the maximization of the total expected discounted value of the burnt areas when all fires are suppressed. In an Markov decision process, the uncertainty in transition probabilities may not be known precisely. In our case, this can happen because quantities in the model of Pappis and Rachaniotis (2006) may have an inherent variability. The probabilities might have been obtained via an estimation process. From the well-known "central limit theorem" (Ross 1973), it is natural to consider the normal distribution for representing the uncertainty in the transition probabilities. Dynamic Allocation Index rule (Gittins and Jones 1974) plays an important role in finding an optimal strategy for the Markovian process. Bellman equations appear naturally in computing this priority rule. There are a number of methods to solve the Bellman equation (See for example (Sonin 2006)). In our Markovian model, the particular form of the transition matrix allows us to solve the corresponding Bellman equation efficiently and analytically.

The remainder of this paper is composed of seven sections. We begin by giving a description of the deterministic forest fire-fighting problem which can be modelled by deteriorating jobs. Section 3 discusses considerations for stochastic approach. In Section 4, a Markovian model is given for the stochastic scheduling problem when time proceeds in discrete steps. Dynamic Allocation Index technique is used for determining the optimal non pre-emptive containment policy. This result is given in Section 5. Section 6 provides the considered dynamic priority
determining. Section 7 reports on our numerical tests. In Section 8, we conclude our work and provide some recommendations for future development.

2. A deteriorating model for scheduling some forest fire-fighting tasks

In the scenario under consideration, $N$ forest fires are ignited simultaneously, detected and reported, and one processor of initial attack is assigned for their containment.

All the $N$ fires evolve independently and no constraint is assumed which impose the order in which fires must be suppressed.

The affected areas may require greater protection because of their characteristics: proximity of population, social, economic, military, historic, etc.

As time elapses, the fire expands if there is a delay in the fire-fighting efforts.

The surviving value of the area at time $t$ where fire $F_i$ is ignited can be estimated by assigning a discounted value $\alpha t V_i$, where $V_i$ is a positive bounded constant and $\alpha$ lies strictly between 0 and 1. This estimate is done by taking into account all characteristics concerned the area.

The objective is to minimize the total damage caused by the fires to the burnt areas when all fires are suppressed, that is the total discounted value of the surviving areas is maximized.

The processor is of limited transportation capacity $Q$ (in $L$) and application rate $\rho$ (in $L h^{-1}$), use water as the means of fire suppression.

The processor is not required to extinguish a fire while busy suppressing another one, which interpret that no pre-emption is allowed.
During the containment of selected fire $F_i$, the processor can interrupt its operation as many times as necessary, $\tau_i$ (in h) in total, for going to the nearest depository to refill and to come back.

In general, square brackets $[\ ]$ will be used to denote position in sequence in a permutation schedule. The symbol $F_{[i]}$ means the fire which is in the $i$th position in suppression sequence.

Let $T_{0,[1]}$ (in h) be the time needed for the processor to travel from the base where it is stationed to the spot where the first prioritized fire might be attacked.

$T_{[i],[i+1]}$ (in h), $i = 1, 2, ..., N-1$, be the time required for the processor to travel from the front of suppressed fire $F_{[i]}$ to the depository to refill and move to the front of fire $F_{[i+1]}$.

The times needed for the processor to travel for suppressing a prescribed fire or interruption for refilling were considered as set-up times (as they would be in a typical manufacturing environment).

For various forest fire-fighting considerations, the suppression of fire $F_i$, $0 \leq i \leq N$, might be started at time $r_i$. The release-time $r_i$, $r_i > 0$, can simply represent the fire's initial inaccessibility period. It represents, for example, the preparation time that precede the containment process.

The total forest area being burnt by fire $F_{[i]}$ at time $t$ at which fire's suppression effort starts, $E_{[i]}(t)$ (in $m^2$), as well as the relative amount of water needed to suppress the fire $F_{[i]}$, $\omega_{[i]}(t)$ (in $L \times m^2$), deteriorates (increases) at a rate that depend upon the forest land that the fire burns, the forest vegetation, and the weather conditions.
\( p_{[i]}(t) (\text{in } h) \) is the processing-time of fire \( F_{[i]} \) if the "decision" of fire containment is taken at time \( t \). \( p_{[i]}(t) \) deteriorates (increases) with \( t \). The following expression holds (Rachaniotis and Pappis 2006):

\[
p_{[i]}(t) = \frac{E_{[i]}(t) \times \omega_{[i]}(t)}{\rho} + \text{int} \left( \frac{E_{[i]}(t) \times \omega_{[i]}(t)}{Q} \right) \times \tau_{[i]}
\]

where "\text{int}" is the integer part.

\( E_{[i]}(t) \) is given in (Rachaniotis and Pappis 2006) by:

\[
E_{[i]}(t) = \frac{\pi}{\psi_{[i]}} \left( \frac{A_{[i]} \times c_{[i]}(t) \times t}{B_{[i]} + 1} \right)^2
\]

And \( \omega_{[i]}(t) \) in (Katsanos 1978) is of the form:

\[
\omega_{[i]}(t) = 1.2 \sqrt{A_{[i]} \times B_{[i]} \times t^{B_{[i]}-1} + c_{[i]}(t) \times t}
\]

where \( A_{[i]} \) and \( B_{[i]} \) are constant coefficients depending on the varying environmental factors of the fire in each area (wind speed, type of fuel and fuel loading) and/or the equilibrium rate of spread (McAlpine and Wakimoto 1991).

\( c_{[i]}(t) \) is a nonnegative coefficient at time \( t \), defined as the difference between the forest fire's rate of spread based on the fire behaviour prediction system and the fire rate of spread during the fire's acceleration phase (McAlpine and Xanthopoulos 1998).

And \( \psi_{[i]} \) is a constant depending on the wind speed \( U_{[i]} \) (in \( Km \times h^{-1} \)).

\( \psi_{[i]} \) is given in (Rachaniotis and Pappis 2006) by the formula:

\[
\psi_{[i]} = \frac{1}{U_{[i]}}
\]
\[ \psi[i] = 0.936 \times e^{-0.07127 \Delta[i]} + 0.461 \times e^{-0.043 \Delta[i]} . \]

Substituting Equations (2) and (3) into Equation (1), it follows that:

\[ p[i](t) = \frac{1.2 \times \pi}{\psi[i] \times \rho} \left( \frac{A[i]}{B[i]} + 1 \right) t^{B[i]+1} + c[i](t) t \right)^2 \times \frac{A[i] \times B[i] \times t^{B[i]-1} + c[i](t)}{A[i] \times B[i] + 1} \times \right] \tau[i] \right] \]

Let \( C[i] \) (in \( h \)) be the completion-time of \( F[i] \) that is the time at which \( F[i] \)'s suppression effort stops. It is:

\[ C[i] = \sum_{j=1}^{i} p[j](t[j]) + \sum_{j=1}^{i} (T[j-1][j] + \eta[j]) \]

where \( t[j] \) is the decision-date for the containment of \( F[j] \), with \( t[1] = 0 \) and \( t[j] = C[j-1] \) for \( j = 2, 3, \ldots, N \), and \( T[0][1] = T_0[1] \).

A fire \( F[i] \) has escaped initial attack if its containment time \( C[i] \) exceeds a specific "time limit" \( d[i] \) (in \( h \)), which can be considered as fire deadline. Section d'équation (suivante)

3. Considerations for stochastic approach

In the expression of \( p[i](t) \) given by Equation (4), the exponent \( B[i] \) of \( t \) is different in each fire case. This fact makes the problem more complex and difficult to solve. In addition, whatever the reliability of this model, it is important to realize that there is some, though very small, probability that the unwanted and unexpected event will happen. This consideration is justified.
by the risks involved in a potential fire containment failure. Despite the research undertaken, the statistical properties of a forest-fire are not fully known.

Fire safety design aims at providing solutions with risk levels that our society can tolerate. For a diverse variety of types of insurance, normal distributions are extremely important in natural sciences for real-valued random variables whose distributions are not known. It can be used to maximize the chance of success for an acceptable level of risk or to minimize the risk for a desired level of success.

Formally, the associated normal density \( f_{i}(t) \) to fire \( F_{i} \) is expressed as:

\[
f_{i}(t) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(t - \mu)^2}{2\sigma^2} \right),
\]

where \( \mu \) and \( \sigma \) are, respectively, the mean and standard deviation. \( t \) is the completion-time of \( F_{i} \) given by (5).

The event: "The fire \( F_{i} \) is controlled by time \( d_{i} \)" must have a considerable probability of occurrence. Consequently the mean \( \mu \) might be \( d_{i} \). The date \( d_{i} \) cannot be considered as "hard deadline" since wildfire often exceed this limit. A safety zone \( [d_{i}, d_{i} + \varepsilon] \) must be considered in which the probability of success in the suppression of \( F_{i} \) has higher value. \( \varepsilon \) depends naturally on \( s_{i} \) the starting-time of fire \( F_{i} \) containment.

The value of the standard deviation \( \sigma \) is related to the level of risk. If we take, for example, 5 % is the accepted risk then 2 standard deviations represent \( d_{i} - r_{i} \). When the containment of fire failure by time \( d_{i} + \varepsilon \), the probability that the intended processor can suppress the fire decrease, which is interpreted that the fire takes rage and merit additional fire-fighting efforts (See Figure).
1. The normal density $f_i(t)$ is used to estimate the probability of controlling the fire $F_i$ at any time $t \geq s_i$. In Figure 1, the area $P_i$ under $f_i(t)$ between $s_i$ and $t_1$ represents the probability that $F_i$ is controlled by time $t_1$ before $d_i$. $P_i$ increases with $t_1$ and when $s_i$ approaches $d_i$ the probability to contain $F_i$ in $d_i$ decreases and safety zone $[d_i, d_i+\varepsilon]$ becomes wide. The probability that fire $F_i$ is controlled in the safety zone must have higher values to that for $t_1 = d_i$. This probability decreases over time and coincides at time $t_2 = d_i + \varepsilon$ with that at $t_1 = d_i$. Exceeding $d_i + \varepsilon$, the fire is uncontrollable and the probability to contain the fire decreases rapidly. These aforementioned natural datum are justified by the expression taken for $P_2$. The value of $\varepsilon$ is then the solution of the equation:

$$
\varepsilon = \frac{1}{2} \int_{s_i}^{d_i} f_i(t) \, dt - \frac{1}{2} \int_{s_i}^{d_i} f_i(t) \, dt.
$$

Under the assumption of stationary probability transition matrix and total expected discounted reward maximization criteria, the problem is a Markov decision problem (Bellmann 1956). In theory, the problem can be solved by dynamic programming technique. However, the combinatorial explosion of states puts severe limitations on finding optimal solutions. We solved this problem optimally and efficiently by computing at each decision epoch a "priority" for each unsuppressed fire, and selecting the fire which has the larger one.

4. A Markovian model for the stochastic problem

We denote by $S = \{(\Omega_i, \Gamma_i, P_i, V_i, \alpha), 1 \leq i \leq N\}$ the associated cost-discounted Markov decision process of the stochastic forest fire-fighting problem with the special features:

$\Omega_i$ is the state space of $F_i$, which may be in our case discrete.

Let $X_i(t)$ be the state of $F_i$ at time $t$, then the state of $S$ at time $t$ is $X(t) = (X_1(t), X_2(t), ..., X_N(t))$. 

https://mc06.manuscriptcentral.com/cjfr-pubs
Fire $F_i$ enters some state of $\Omega_i$ denoted by the symbol $*_i$ if $F_i$ is completed. $*_i$ is the stationary state of $F_i$ in which no transition to another state can be made.

At time $t$, the state $X_i(t)$ is then $*_i$ or it takes a nonnegative integer $\theta_i$, interpreted as the number of time units already spent on $F_i$.

The action space $\Gamma_i$ consists of just two elements $a_i$ and $\bar{a}_i$. Action $a_i$ will be referred to as "select $F_i$ for containment" or "maintain the suppression of $F_i$ if its treatment is started and is not completed". In contrast, $\bar{a}_i$ will be referred to as "stop the suppression process of $F_i$" off-course if $F_i$ is extinguished.

If action $a_i$ is taken at time $t$, then the suppression of fire $F_i$ makes progress or completes during the interval $[t, t+1]$, and the state of all fires not in receipt of processor remain fixed to zero, i.e., either $X_i(t+1) = *_i$ or $X_i(t+1) \geq X_i(t)$,

and $\forall j, j \neq i$ either $X_j(t+1) = *_j$ or $X_j(t+1) = X_j(t) = 0$.

Further, we shall only consider Markovian transitions. The transition probability from state $X_i(t)$ to state $X_i(t+1)$ is:

\begin{equation}
    P\left[ X_i(t+1) = *_i \mid X_i(t) = *_i \right] = 1,
\end{equation}

\begin{equation}
    P\left[ X_i(t+1) = *_i \mid X_i(t) = \theta, a_i \right] = \begin{cases} 
        \int_{s_i}^{\theta+1} f_i(t)dt & \text{if } s_i \leq \theta \leq d_i \\
        1 - \int_{s_i}^{\theta+1} f_i(t)dt & \text{if } \theta > d_i 
    \end{cases},
\end{equation}

(See Figure 1) and
Draft

(9) \[ P[X_i(t+1) = \theta+1 | X_i(t) = \theta, a_i] = 1 - P[X_i(t+1) = * | X_i(t) = \theta, a_i], \]

with \( X_i(s_i) = 0 \) and \( f_i \) is the normal distribution given beforehand in (6).

In decision processes theory, a policy is defined as any rule for choosing at each decision epoch that fire to be suppressed.

Remember that in Section 2, the optimal policy (if it exists) must to be non pre-emptive.

Decision times are then the initial time, and the fire completion-times.

Should action \( a_i \) is taken at time \( t \), an incremental rewards \( \alpha \) \( V_i \) is earned. The current reward \( R(X(t)) \) of the process \( S \) is then \( R(X(t)) = \alpha V_i \).

The objective function to be maximized is the total expected discounted reward function over an infinite horizon:

\[
E \left\{ \sum_{t=0}^{+\infty} R(X(t)) | X(0) = 0 \right\},
\]

which can be written as:

\[
(10) \quad E \left\{ \sum_{i=1}^{N} \alpha^i V_i \right\}
\]

where \( t_i \) is the first decision-time at which \( F_i \) is selected for containment.

Note that (10) is well defined because rewards \( V_i \) are bounded and \( \alpha \in ]0, 1[ \).

The general theory of discounted Markov decision process (Ross 1973) asserts the existence of an optimal policy for the process \( S \) which maximizes the total expected reward (10), which is
deterministic, stationary, of Markov, and satisfies the optimality equations of dynamic programming.

5. Dynamic Allocation Index rule for optimal processing policy

In this section, we denote by $k$ the last suppressed fire, and by $t$ its completion-time $C_k$. If no one exists then $k = 0$ and $t = 0$.

If action $a_i$ is taken first at time $t$, and action $a_j$ is taken immediately after it, namely at time $t + T_{k,i} + r_i + p_i(t)$, then the sum of discounted rewards obtained from the application of these two actions is

$$\alpha^t V_i + \alpha^{t + T_{k,i} + r_i + p_i(t)} V_j.$$  

If the action order is interchanged, we obtain

$$\alpha^t V_j + \alpha^{t + T_{k,i} + r_i + p_i(t)} V_i.$$  

A little algebra shows that the first ordering has the greater reward if

$$\frac{V_i}{1 - \alpha^{T_{k,i} + r_i + p_i(t)}} > \frac{V_j}{1 - \alpha^{T_{k,i} + r_j + p_j(t)}}.$$  

Using this idea, it is not hard to see that the total expected discounted reward obtained from the $N$ fires is maximized by allocating at each decision epoch $t$ the processor to whichever unsuppressed fire has the largest Dynamic Allocation Index $\gamma_i(t)$ computed as

$$\gamma_i(t) = \frac{V_i}{1 - \alpha^{T_{k,i} + r_i + p_i(t)}}.$$
Let $D(t)$ denotes the set of delayed fires up to decision-time $t$. The following is the main result of this paper.

**Theorem 1.** At each decision epoch $t$ it is optimal to choose action $a_i$ if and only if,

$$
\gamma_i(X_i(t)) = \max_{j \in D(t)} \gamma_j(X_j(t)) = \max_{j \in D(t)} \frac{V_j}{1 - \alpha^{T_{x_{i}}, r_{i}}} E\{\alpha^{p_j(t)}\},
$$

6. Computing the Dynamic Allocation Index

In this study, determining $\gamma(X_i(t))$ passes by the calculation of

$$
E\{\alpha^{p_j(t)}\}, j \in D(t).
$$

The simple form of the transition matrix of probabilities is the main virtue in computing (11).

Suppose that $(F_{[1]}, F_{[2]}, \ldots, F_{[i-1]})$ is the current permutation of suppressed fires.

Then the current decision-time is $t = C_{[i-1]}$.

Denote by $M_{[i]}^{(0)}$ the probability generating function of $p_{[i]}(t)$ evaluated at $\alpha$ conditional upon initial state "$X_{[i]}(s_{[i]}) = 0$", where $s_{[i]} = C_{[i-1]} + T_{[i-1], [i]} + r_{[i]}$, i.e.

$$
M_{[i]}^{(0)} = E\{\alpha^{p_{[i]}(t)} | X_{[i]}(s_{[i]}) = 0\}.
$$

Using the geometric sum, it is easy to show that Equation (12) can be expressed as

$$
M_{[i]}^{(0)} = 1 - (1 - \alpha) \sum_{h=0}^{p_{[i]}(t)-1} \alpha^h | X_{[i]}(s_{[i]}) = 0\}.
$$
Let \( H_{[i]}^{(0)} \) denote in Equation (13) the conditional expectation

\[
\mathbb{E}
\left\{
\sum_{h=0}^{p_{[i]}(r)-1} \alpha^h \mid X_{[i]}(s_{[i]}) = 0
\right\}
\]

\( H_{[i]}^{(0)} \) is obtained by solving the following Bellman equation (Ross 1973)

\[
H_{[i]}^{(y)} = 1 + \alpha \sum_{h=y+1, \ldots, q-1} P_{[i]}^{(y,h)} H_{[i]}^{(h)}, \quad y = 0,1, \ldots, q-1
\]

where in (14):

\[
q = p_{[i]}(r),
\]

\[
H_{[i]}^{(q)} = 0,
\]

\[
P_{[i]}^{(y,0)} = 0, \ h > y + 1, \ h \neq \ast_{[i]},
\]

\[
P_{[i]}^{(y,y+1)} = P \left[ X_{[i]} \left( s_{[i]} + y + 1 \right) \mid X_{[i]} \left( s_{[i]} + y \right) = y, a_{[i]} \right], y = 0,1, \ldots, q-1,
\]

\[
P_{[i]}^{(y,\ast_{[i]})} = P \left[ X_{[i]} \left( s_{[i]} + y + 1 \right) = \ast_{[i]} \mid X_{[i]} \left( s_{[i]} + y \right) = y, a_{[i]} \right], y = 0,1, \ldots, q-1,
\]

\( P \) is the transition probability defined in (7), (8) and (9).

Consequently,

\[
H_{[i]}^{(0)} = 1 + \alpha P_{[i]}^{(0,1)} H_{[i]}^{(1)} + \alpha P_{[i]}^{(0,\ast_{[i]})} H_{[i]}^{(\ast_{[i]})}
\]

\[
H_{[i]}^{(1)} = 1 + \alpha P_{[i]}^{(1,2)} H_{[i]}^{(2)} + \alpha P_{[i]}^{(1,\ast_{[i]})} H_{[i]}^{(\ast_{[i]})}
\]

\[
\ldots
\]

\[
H_{[i]}^{(q-1)} = 1 + \alpha P_{[i]}^{(q-1,q)} H_{[i]}^{(q)} + \alpha P_{[i]}^{(q-1,\ast_{[i]})} H_{[i]}^{(\ast_{[i]})}
\]

\[
H_{[i]}^{(\ast_{[i]})} = 1 + \alpha P_{[i]}^{(\ast_{[i]},\ast_{[i]})} H_{[i]}^{(\ast_{[i]})}
\]

The structure of system (15) provided us to solve it directly in ascent fashion.
Namely, 

\[ H^{(q-1)}_{i} = 1 + \frac{\alpha}{1-\alpha} P_{i}^{(q-1)}, \]

\[ H^{(0)}_{i} = 1 + \frac{\alpha}{1-\alpha} P_{i}^{(0)} + \sum_{j=1}^{q-1} \alpha^j \left( \prod_{i=0}^{j-1} P_{i}^{(i+1)} \right) \left( 1 + \frac{\alpha}{1-\alpha} P_{i}^{(j)} \right). \]  

Consequently, the final form of the Dynamic Allocation Index of fire \( F_{i} \) is then 

\[ \gamma_{i}(t) = \frac{V_{i}}{1 - \alpha^i \prod_{i=0}^{\eta} (1 - (1 - \alpha)) H^{(0)}_{i}}, \]

where \( H^{(0)}_{i} \) is given in (16), and \( T_{[0,1]} = T_{0,1} \).

7. Numerical tests

The reported Dynamic Allocation Index, given by Equation (17), was coded and run in Scilab 5.5.2. All Tests are made on a personal computer with Intel(R) Core(TM) i5-2450M CPU@ 2.50 GHz and 4 Go of RAM Under Windows 7 (64 bits). All parameters are randomly generated according to the uniform law in specific intervals esteemed using experimental data from (McAlpine and Wakimoto 1991) (see (Rachaniotis and Pappis 2006)):

- \( A_i \in [0, 14] \) and \( B_i \in [1, 2] \).
- In general \( c_i(t) \in [0, 11] \) for \( U_i \in \{0, 1, ..., 8\} \).
- \( \alpha \in [10^{-6}, 10^{-4}] \) and \( V_i \in [10^5, 2 \times 10^5] \).
- \( \rho = 6.48 \times 10^6 \) (in \( \text{Lh}^{-1} \)) and \( Q = 3 \times 10^4 \) (\( \text{L} \)) (estimations).
In this experimentation, it is assumed that $\tau_i = 0.2$ (in h), $T_{0[i]}$ and $T_{[i][i+1]}$ are in $]0, 1[$ (in h).

The standard deviation $\sigma_i$ is taken randomly according to the uniform law, as envisaged for the other parameters in this experimentation, in the set $\left\{ \frac{d_i-r_i}{4}, \frac{d_i-r_i}{3}, \frac{d_i-r_i}{2}, d_i-r_i \right\}$, since almost all of the area under the bell curve lies within 4 standard deviations from the mean. Thus, outliers more than 4 standard deviations from the mean will be extremely rare.

In our tests, we mainly interested in estimating the deadline $d$ of fires containment and, in respect of this delay, the number of fires that can be surmounted by the processor. For this purpose, $d$ taken integer was raised by "one" from 1 to 50 (in h). For each $d$ value, the algorithm has been applied on a variety of instances (1000 instances in total, each of which contains 100 cases of fires).

To get accuracy in the calculation of the Dynamic Allocation Indices, we set the number of fire's states to 100 states.

For each value of $d$ and for the 1000 instances generated for this delay, we tabulate the average of numbers of fires suppressed by the processor (denoted $\tilde{N}$), the average of the containment times for the first fires selected by the index rule and escaped the efforts of initial attack (denoted $\tilde{C}$), and the average of CPU time consumption (in sec) (denoted $\tilde{T}$). These results are reported in Table 1. It appears that only 07 fires (on average) can be confined within the period of 11 to 50 first hours (on average). With $d$ between 11 and 50, the time required to control the $8th$ fire deteriorates and exceeds all barriers feasible in practice. The calculation time for each instance is about 1.6 second.

8. Conclusion
Regardless of the complex structure of the model by Rachaniotis and Pappis (2006), and of inability to solve the problem perfectly, it may be also argued that uncertainty can enter the model because quantities in the model have an inherent variability. We have addressed a novel approach for the optimal solution when uncertainty occurs. A Markovian decision process is developed, and Dynamic Allocation Index technique is used to solve the problem. The optimality of this rule is proved, and its determination is also explained. By field tests, fires deadline are at most 11 hours, and only 07 fires can be surmounted by the processor.

In our Markovian process the rewards are fixed, so that the only parameters that vary are the transition probabilities. To our knowledge, the statistical properties of forest-fire are not fully known. We adopt normal distribution for representing the uncertainty in the transition probabilities. The normal distribution is most often assumed to describe the random variation that occurs in the data from many scientific disciplines with condition that their distributions should be unknown. The choice of the mean to be the deadline of fire and standard deviation governed by the risk of fire suppression failure seem natural in forest fires-fighting.

A notable direction of study is to consider the problem of scheduling a single fire-fighting processor under incomplete information about the normal distribution means as in (Katehakis and Robbins 1995), or incomplete information about the fire containment lengths as in (Burnetas and Katehakis 1993), or use the consistent estimates of the transition law and reward as defined in (Burnetas and Katehakis 1997).

In our study, fires are ignited at the same time. The generalization to the case of fires ignited at different times, as depicted in (Rachaniotis and Pappis 2006), is quite simple. Thereby, dispute the difficulty of Poissonian arrivals (Blake 1979) seems worthwhile. It is also interesting to discuss the case in which fires are disadvantaged during their containment. In this situation, and
unlike the deteriorating models, the suppression of a fire generates a supplementary processing-
time to complete the fire.

The problem of scheduling parallel identical or non-identical fire-fighting processors is treated
by Pappis and Rachaniotis (2010) as a job-scheduling with deteriorating jobs. In this problem the
pre-emption is allowed with a variable number of processors used by a fire over time. An
interesting model for fire processing-times is formulated depending on the number and type of
each processor employed to contain a fire. A stochastic continuation of this model is almost
similar to that developed in Section 4 and the dynamic allocation index \( \gamma_i \) can be generalized
without jeopardizing the optimality of the index rule. With each fire \( F_i \), which is at time \( t \) in state
\( X_i(t) \), is associated a priority
\[
\gamma_i \left( X_i(t) \right) = \sup_{\xi \leq t} \frac{V_i}{1 - \alpha^{t + \sigma} \mathbb{E}(\alpha^\xi)}
\]
where \( T_i \) is the time required to move the needed processors from other sites to the spot where
fire \( F_i \) is ignited, and \( \xi \) is the stopping-time of \( F_i \). \( \xi \) is a decision variable that depends on the fire
processing-time calculated at time \( t \). This prioritization allows the decision-maker to allocate a
minimal set of required processors to the fire with the most highly index. In a case of a tie, any
well known rule is employed to break it. An important property of Dynamic Allocation Index \( \gamma_i \)
is that the supremum is achieved by \( \xi = \min_{h \geq t} \{ h, \gamma_i(X_i(h)) = \gamma_i(X_i(t)) \} \) and by any stopping time
\( \sigma > t \) which satisfies \( \sigma \leq \xi \) and \( \gamma_i(X_i(\sigma)) = \gamma_i(X_i(t)) \). At each decision epoch, which corresponds to
an optimal stopping-time or a completion-time or new arrival of a fire, the processing-times of
uncompleted fires are recalculated for new indexation and new assignments of resources are
done. Suitable methodologies for computing \( \gamma_i \) are discussed in (Katehakis and Veinott Jr. 1987).
In Pappis and Rachaniotis (2010), "when there are more processors available than those already assigned, which are simultaneously required by the fires already ignited, a maximal possible number of processors are assigned. Otherwise, processors are shared by fires with equal ratios so that their heights decrease at the same rate. The processors are assigned according to non-ascending order of the ratios containment rate/travelling time, which depends directly on the distance from their location to the fire". Please note that the action taken at each decision epoch is always optimal with regard to the state resulting from the allocation of residual processors to fires already ignited up to time $t$.

References


Table 1. Experimentation results

<table>
<thead>
<tr>
<th></th>
<th>d</th>
<th>N</th>
<th>Ĉ</th>
<th>Ĥ</th>
<th></th>
<th></th>
<th>d</th>
<th>N</th>
<th>Ĉ</th>
<th>Ĥ</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.22</td>
<td>1.37</td>
<td>1.03</td>
<td>18</td>
<td>7.14</td>
<td>537.2</td>
<td>1.56</td>
<td>35</td>
<td>7.76</td>
<td>30272.</td>
<td>1.61</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2.54</td>
<td>2.37</td>
<td>1.18</td>
<td>19</td>
<td>7.06</td>
<td>839.6</td>
<td>1.57</td>
<td>36</td>
<td>7.6</td>
<td>12106.</td>
<td>1.60</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.86</td>
<td>3.69</td>
<td>1.31</td>
<td>20</td>
<td>7.34</td>
<td>4260.</td>
<td>1.58</td>
<td>37</td>
<td>7.14</td>
<td>41890.</td>
<td>1.59</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5.18</td>
<td>6.47</td>
<td>1.42</td>
<td>21</td>
<td>7.08</td>
<td>12791.</td>
<td>1.57</td>
<td>38</td>
<td>7.36</td>
<td>12334.</td>
<td>1.58</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5.86</td>
<td>11.40</td>
<td>1.48</td>
<td>22</td>
<td>7.96</td>
<td>2397.</td>
<td>1.6</td>
<td>39</td>
<td>7.8</td>
<td>45222.</td>
<td>1.60</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>6.24</td>
<td>22.95</td>
<td>1.49</td>
<td>23</td>
<td>7.24</td>
<td>5620.</td>
<td>1.6</td>
<td>40</td>
<td>7.62</td>
<td>108125.</td>
<td>1.59</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>6.3</td>
<td>20.96</td>
<td>1.52</td>
<td>24</td>
<td>7.1</td>
<td>5000.</td>
<td>1.59</td>
<td>41</td>
<td>7.18</td>
<td>564072.</td>
<td>1.59</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>6.2</td>
<td>66.35</td>
<td>1.51</td>
<td>25</td>
<td>7.24</td>
<td>2748.</td>
<td>1.58</td>
<td>42</td>
<td>7.36</td>
<td>12334.</td>
<td>1.58</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>6.8</td>
<td>120.53</td>
<td>1.53</td>
<td>26</td>
<td>7.7</td>
<td>14101.</td>
<td>1.59</td>
<td>43</td>
<td>7.82</td>
<td>173442.</td>
<td>1.59</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>6.72</td>
<td>196.28</td>
<td>1.53</td>
<td>27</td>
<td>7.5</td>
<td>10661.</td>
<td>1.6</td>
<td>44</td>
<td>7.66</td>
<td>108284.</td>
<td>1.62</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>7.08</td>
<td>98.76</td>
<td>1.54</td>
<td>28</td>
<td>7.1</td>
<td>10352.</td>
<td>1.59</td>
<td>45</td>
<td>7.18</td>
<td>564072.</td>
<td>1.61</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>7.14</td>
<td>156.5</td>
<td>1.54</td>
<td>29</td>
<td>7.24</td>
<td>2801.</td>
<td>1.6</td>
<td>46</td>
<td>7.88</td>
<td>186788.</td>
<td>1.62</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>7.08</td>
<td>164.48</td>
<td>1.54</td>
<td>30</td>
<td>7.7</td>
<td>14101.</td>
<td>1.59</td>
<td>47</td>
<td>7.68</td>
<td>108302.</td>
<td>1.62</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>7.06</td>
<td>403.24</td>
<td>1.55</td>
<td>31</td>
<td>7.56</td>
<td>11212.</td>
<td>1.61</td>
<td>48</td>
<td>7.2</td>
<td>582567.</td>
<td>1.662</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>7.18</td>
<td>1721.6</td>
<td>1.56</td>
<td>32</td>
<td>7.12</td>
<td>10577.</td>
<td>1.6</td>
<td>49</td>
<td>7.38</td>
<td>13026.</td>
<td>1.62</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>7.18</td>
<td>486.37</td>
<td>1.56</td>
<td>33</td>
<td>7.32</td>
<td>10197.</td>
<td>1.589</td>
<td>50</td>
<td>7.92</td>
<td>332857.</td>
<td>1.634</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>7.2</td>
<td>449.96</td>
<td>1.57</td>
<td>34</td>
<td>7.66</td>
<td>18542.</td>
<td>1.58</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Fig.1. The probability determining: $P_1$ and $P_2$ are respectively the probabilities that fire $F_i$ is controlled at time $t_1$ before time $d$ and at time $t_2$ after time $d$. 
Scilab Source Code

1. Variables and parameters definition

2. File 1: DAI.sce (Main File)

3. File 2: Data.sci (Data of the problem)

4. File 3: Schedule.sci (A schedule of selected fires)

5. File 4: processing_time.sci (Processing time of a selected fire)

6. File 5: starting_time.sci (Fire starting-time)

7. File 6: completion_time.sci (Completion time of a selected fire)

8. File 7: cdfnormal.sci (Cumulative distribution function of normal distribution, Written by Philippe.CASTAGLIOLA@univ-nantes.fr, Université de Nantes & IRCCyN UMR CNRS 6597)

9. File 8: transition_matrix.sci (Matrix of transition probabilities)

10. File 9: DAI_fire.sci (Dynamic Allocation Index of fire)
1. **Variables and parameters definition**

\[ \text{NS} : \text{Number of states} \]

\[ \text{NI} : \text{Number of instances} \]

\[ \text{NF} : \text{Number of fires} \]

\[ \text{SF} : \text{Set of fires} \]

\[ \text{A and B} : \text{Constant coefficients depending on the varying environmental factors of the fire in each area (wind speed, type of fuel and fuel loading) and/or the equilibrium rate of spread} \]

\[ \text{U} : \text{Wind speed in the area where the fire is ignited} \]

\[ \text{c} : \text{Nonnegative coefficient, defined as the difference between the forest fire's rate of spread based on the fire behaviour prediction system and the fire rate of spread during the fire's acceleration phase} \]

\[ \text{Gamma} : \text{constant depending on the wind speed U} \]

\[ \text{alpha} : \text{a fixed discount factor} \]

\[ \text{V} : \text{the associated bounded value to the area where fire is ignited} \]

\[ \text{ro} : \text{application rate of the processor} \]

\[ \text{Q} : \text{transportation capacity of the processor} \]

\[ \text{thau} : \text{the necessary time for the processor for going to the nearest depository of the area where fire is ignited to refill and to come back} \]

\[ \text{T(0,f)} : \text{the time needed for the processor to travel from the base where it is stationed to the spot where the fire f might be attacked} \]
\( T(f_1, f_2) \): is the time required for the processor to travel from the front of suppressed fire \( f_1 \) to the depository to refill and move to the front of fire \( f_2 \).

\( r \): release-date of fire

\textit{cdfnormal}: cumulative distribution function of normal distribution

\( f \): fire

\( C \): completion-time

\( s \): starting-time

\( p \): processing-time

\textbf{DAI}: Dynamic Allocation Index of a fire

\textbf{TM}: transition matrix

\textbf{step}: step

\( d \): deadline

\( D \): deviation

\( Pr \): transition probability

\textbf{LSF}: last selected fire

\textbf{CTLSF}: completion of last selected fire

\textbf{SCHEDULE}: a sequence of suppressed fires

\textbf{Tr_Mat}: is a matrix, contains all fire’s Transition Matrices of probabilities

\( \text{pos} \): position in the schedule
2. **File 1: DAI.sce (Main File)**

```plaintext
exec('C:\Users\pc\Desktop\...\Data.sci')
exec('C:\Users\pc\Desktop\...\Schedule.sci')
exec('C:\Users\pc\Desktop\...\processing_time.sci')
exec('C:\Users\pc\Desktop\...\starting_time.sci')
exec('C:\Users\pc\Desktop\...\completion_time.sci')
exec('C:\Users\pc\Desktop\...\cdfnormal.sci')
exec('C:\Users\pc\Desktop\...\transition_matrix.sci')
exec('C:\Users\pc\Desktop\...\DAI_fire.sci')
funcprot(0);
stacksize('max');
format(10);
timer();
NS=100;
NI=1000;
time=[];
LS=[];
CTLSF2=[];
for d=1:50
  for i=1:NI
    [r,T,T0,thau,Q,ro,V,alpha,Gamma,c,U,B,A,SF,NF]=Data(i);
    CTLSF=0;
    Tr_Mat=TM1(SF);
    [Sched,CTLSF,LSF]=schedule(NF,SF,Tr_Mat);
    CTLSF2(i)=CTLSF;
  end
end
```

https://mc06.manuscriptcentral.com/cjfr-pubs
LS(i)=length(Sched);

end

time(i)=timer();

end

average_running_time(d)=sum(time)/NI;

average_Lenght_Schedule(d)=sum(LS)/NI;

average_completion_time(d)=sum(CTLSF2)/NI;

end
3. **File 2**: Data.sci (Data of the problem)

```matlab
function [r, T, T0, tau, Q, ro, V, alpha, Gamma, c, U, B, A, SF, NF]=Data(i);
NF=grand(1,1,"uin",100,100)
SF=[1:NF];
A=grand(1,NF,"unf",0,14);
B=grand(1,NF,"unf",1,2);
U=grand(1,1,"uin",0,8);
c=grand(1,NF,"unf",0,11);
Gamma=0.936*exp(0.07127*U)+0.461*exp(-0.043*U);
alpha=grand(1,1,"unf",0.000001,0.0001);
V=grand(1,NF,"unf",100000,200000);
ro=6480000;
Q=30000;
thau=0.2;
T0=grand(1,NF,"unf",0,1);
for j=1:1:(NF-1)
    for i=j+1:1:NF
        T(j,i)=grand(1,1,"unf",0,1);
        T(i,j)=T(j,i);
    end
end
r=grand(1,NF,"unf",0,1);
endfunction
```
4. **File 3**: Schedule.sci (A schedule of selected fires)

```plaintext
function [Sched, CTLSF, LSF]=schedule(NF, SF, Tr_Mat)
   for pos=1:NF
      t=0;
      indices=[];
      for f=SF
         TM=Tr_Mat(:,1+t:NS+1+t);
         indices=[indices,DAI(TM)];
         t=NS+1+t;
      end
      max_DAI=max(indices);
      i=find(indices==max_DAI);
      LSF=SF(i);
      CTLSF=C(SF(i));
      if CTLSF>d then
         break
      else
         Sched(pos)=SF(i);
         SF(i)=[];
         Tr_Mat(:,(i-1)*(NS+1)+1:i*(NS+1))=[];
      end;
   end
endfunction
```

5. **File 4: processing_time.sci** (Processing time of a selected fire)

```plaintext
function y = p(f)
    y = ((1.2*%pi)/(Gamma*ro))*(((A(f)*(s(f)^(B(f)+1)))/(B(f)+1))+c(f)*s(f))^2)*sqrt(A(f)*B(f)*
    (s(f)^(B(f)-1)+c(f)))+int(((1.2*%pi)/(Gamma*Q))*(((A(f)*(s(f)^(B(f)+1)))/(B(f)+1))+
    c(f)*s(f))^2)*sqrt(A(f)*B(f)*s(f)^(B(f)-1)+c(f)))*thau
endfunction
```

6. **File 5: starting_time.sci** (fire starting-time)

```scilab
function y = s(f)
if CTLSF==0 then
  y = T0(f) + r(f)
else
  y = CTLSF + T(LSF,f) + r(f)
end
endfunction
```

7. **File 6: completion_time.sci** (Completion time of a selected fire)

```matlab
function y = C(f)
    y = s(f) + p(f)
endfunction
```
8. **File 7: cdfnormal.sci** (cumulative distribution function of normal distribution, Written by Philippe.CASTAGLIOLA@univ-nantes.fr Université de Nantes & IRCCyN UMR CNRS 6597)

function \( Y = \text{cdfnormal}(X, \mu, \sigma) \)

[argout,argin]=argn()

if (argin<1)|(argin>3)
    error("incorrect number of arguments")
end

if ~exists("mu","local")
    mu=0
end

if ~exists("sigma","local")
    sigma=1
end

if sigma<=0
    error("argument "sigma" must be > 0")
end

\( Y = \text{cdfnor}("PQ",X,\mu*\text{ones}(X),\sigma*\text{ones}(X)) \)

defunction

254
255
256
257
258
259
function y = TM(f)
step = linspace(s(f), C(f), NS)
step(1) = []
n = grand(1, 1, "uin", 1, 4)
D = (d - r(f)) / n
k = 1
for t = step
    if t <= d then
        Pr(k, k+1) = cdfnormal(t, d, D) - cdfnormal(s(f), d, D)
        PP(k) = 1 - Pr(k, k+1)
    else
        PP(k) = cdfnormal(t, d, D) - cdfnormal(s(f), d, D)
        Pr(k, k+1) = 1 - PP(k)
    end
    k = k + 1
end
Pr = [Pr, PP]
Pr = [Pr; zeros(Pr(1,:), :)]
Dim = size(Pr)
Pr(Dim(1), Dim(2)) = 1
y = Pr
endfunction

function Tr_Mat = TM1(SF)
Tr_Mat = [];
for f=SF
  Tr_Mat=[Tr_Mat,TM(f)];
end
endfunction
10. **File 9**: DAI_fire.sci (Dynamic Allocation Index of fire)

```matlab
function y = DAI(TM)

P1 = [];
Product = [];
for l = 1:NS-1
    P1 = [P1, TM(l,l+1)]
    Product = [Product, (alpha^l)*prod(P1)*(1+(alpha/(1-alpha))*TM(l+1,NS+1))]
end
H = 1 + ((alpha*TM(1,NS+1))/(1-alpha) + sum(Product))
if CTLSF == 0  then
    y = V(f)/(1-(alpha^(T0(f)+r(f)))*(1-(1-alpha))*H)
else
    y = V(f)/(1-(alpha^(T(LSF,f)+r(f)))*(1-(1-alpha))*H)
end
endfunction
```