Solving complex maintenance planning optimization problems using stochastic simulation and multi-criteria fuzzy decision making

Sahar Tahvili, Jonas Österberg, Sergei Silvestrov, and Jonas Biteus

Citation: [AIP Conference Proceedings](#) **1637**, 766 (2014); doi: 10.1063/1.4904649

View online: <http://dx.doi.org/10.1063/1.4904649>

View Table of Contents: <http://scitation.aip.org/content/aip/proceeding/aipcp/1637?ver=pdfcov>

Published by the [AIP Publishing](#)

Articles you may be interested in

[Determination of criteria weights in solving multi-criteria problems](#)

AIP Conf. Proc. **1635**, 3 (2014); 10.1063/1.4903555

[FSAW for REIT selection in multi-criteria decision making](#)

AIP Conf. Proc. **1605**, 1098 (2014); 10.1063/1.4887744

[Solving decision making problems using fuzzy numbers with area dominance approach](#)

AIP Conf. Proc. **1522**, 229 (2013); 10.1063/1.4801128

[SUEJ61: MultiCriteria Optimization for IGRT Decision Processes](#)

Med. Phys. **39**, 3666 (2012); 10.1118/1.4734896

[TUABRB01: Convex Optimization Techniques for Handling Delivery Time and Plan Complexity in MultiCriteria VMAT Optimization](#)

Med. Phys. **38**, 3738 (2011); 10.1118/1.3613066

Solving complex maintenance planning optimization problems using stochastic simulation and multi-criteria fuzzy decision making

Sahar Tahvili*, Jonas Österberg[†], Sergei Silvestrov[†] and Jonas Biteus**

**Mälardalen University, Sweden*

[†]Division of Applied Mathematics, Mälardalen University, Sweden

***Scania CV, Sweden*

Abstract. One of the most important factors in the operations of many cooperations today is to maximize profit and one important tool to that effect is the optimization of maintenance activities. Maintenance activities is at the largest level divided into two major areas, corrective maintenance (CM) and preventive maintenance (PM). When optimizing maintenance activities, by a maintenance plan or policy, we seek to find the best activities to perform at each point in time, be it PM or CM. We explore the use of stochastic simulation, genetic algorithms and other tools for solving complex maintenance planning optimization problems in terms of a suggested framework model based on discrete event simulation.

Keywords: Maintenance optimization, Decision theory, Genetic algorithm, Discrete event simulation, Intelligent agent.

INTRODUCTION

One of the most important factors in the operations of many cooperations today is to maximize profit and one important tool to that effect is the optimization of maintenance activities. Maintenance activities is at the largest level divided into two major areas, corrective maintenance activities (CM) and preventive maintenance activities (PM). Corrective maintenance is, per definition, performed as a response to a system failure while preventive maintenance is performed when the system is operational and to avoid future system failure. When optimizing maintenance activities, by a maintenance plan or policy, we seek to find the best activities to perform at each point in time, be it PM or CM. The optimization of these activities is in large affected by their financial implications for a specific cooperation, where given two equivalent systems (mechanical or otherwise) under similar operations may require two quite different maintenance policies for two different cooperations.

A concise review and analysis of different maintenance optimization models can be found in [1]. In the article the authors describe several models for analytical optimization of PM policies and mention computer simulation as a good tool whenever simplifications of systems, to make them analytically tractable, would lead to unrealistic results. In light of this we have focused our efforts towards a simulation approach to maintenance optimization with the benefit of a capability to optimize more complex systems.

SIMULATION FRAMEWORK MODEL

In this section we introduce a framework model for simulation of a stochastic system, the reader may think of it in terms of a mechanical system operating under some cooperate environment and subject to corrective and preventive maintenance activities.

Consider a discretization of time into time-steps Δt and a description of a system in which all events fill up a whole number of such time-steps. By making Δt sufficiently small, such a model can describe the system with arbitrary precision. We consider a time horizon of T such discrete time units. At each point in time the system is described in all important aspects by a state vector $S \in \mathbb{R}^M$, the current state includes the system time (multiple of Δt) and any variables describing the components of the system.

Furthermore, consider three types of events, random events which happen stochastically depending on the evolution of the system state, plannable events that may happen by choice depending on the current system state and a default event that happen whenever neither a random nor a plannable event occur. Let $\mathbf{P} = \{P_i\}$ be the set of plannable events,

$\mathbf{R} = \{R_i\}$ be the set of random events and D be the default event. All events are considered as functions that does nothing else than change the current state, $P_i, R_i, D : \mathbb{R}^M \rightarrow \mathbb{R}^M$. Let r_i be functions $r_i : \mathbb{R}^M \rightarrow [0, 1]$ corresponding to each random event R_i such that $r_i(S_t)$ is the probability that the event R_i was triggered before time t . Let p_i be functions (called deciders) $p_i : \mathbb{R}^M \rightarrow \{True, False\}$ corresponding to each plannable event P_i such that if $p_i(S_t) = True$ then the plannable event P_i is triggered at time t and if $p_i(S_t) = False$ the event is not triggered at time t .

For the rationale of this simulation system we also assume that if several plannable and random events compete to run at a specific time, the order in which they are executed has limited effect on the results of the simulation. We also assume that the plannable events, when executed, has no significant increasing effect on the probability of the random events to be triggered in the next time-step.

THE SIMULATION ALGORITHM

Consider Figure 1, describing the major structure of our simulation algorithm.

FIGURE 1. Schematic view of our simulation algorithm.

We begin by setting the current state to the initial state and in each iteration of the algorithm we first execute all plannable events P_i for which $p_i(S) = True$, we then randomize the level $\bar{r}_i$ at which the random events will trigger in the future, execute the default event D , execute each random event R_i for which the current level exceeds the trigger level ($r_i \geq \bar{r}_i$), and we iterate this until our termination-time T is reached. A more detailed description can be found in the following algorithm.

```

1:  $S \leftarrow S_{initial}$ 
2: while  $t \leq T$  do
3:   for all  $i$  where  $p_i(S) = True$  do
4:      $S \leftarrow P_i(S)$ 
5:   end for
6:   for all  $i$  do
7:      $\bar{r}_i \leftarrow$  sample from uniform distribution on  $[r_i(S), 1]$ 
8:   end for
9:    $S \leftarrow D(S)$ 
10:  for all  $i$  where  $r_i(S) \geq \bar{r}_i$  do
11:     $S \leftarrow R_i(S)$ 
12:  end for
13: end while

```

Note that the state S may change for subsequent i in the for-loops at lines 3 and 10 and our implementation iterates through i in increasing order, of course the values taken of i depends on the number of plannable and random events respectively.

Normally, since we have randomly triggered events, we would need to run this simulation a number of times, perhaps 10 to 1000 repetitions, and collect statistics for the state variables in S as they develop over time. In our implementation we gather the first two moments of all state variables for each point in time, that is, their mean value and their mean square value.

EXAMPLE PROBLEM CLASS

The framework model and simulation algorithm in the previous sections allows for easy modeling of many real-world systems. In a specific model one of the most important decisions to make is the time discretization Δt . Making it too small will increase the run-time of any simulation and making it too large will introduce greater error in the simulation output and subsequent decisions based on the output. Another key factor is choosing specific ways in which to encode the deciders p_i . One of the simplest deciders, which we have used in this paper and specifically in our example problem class, is the *linear decider*.

A linear decider p_i in our framework model is a real vector $p_i \in \mathbb{R}^M$ such that the outputted decision value, True or False, is equal to the truth value of the statement $p_i \cdot S > 1$. This allows the decision to perform a plannable event to depend to different degrees on different elements in the state vector S . In many cases then, when modeling a maintenance system, all aspects of the system is constructed to describe the real world system in sufficient detail, while the deciders are the desired output to be chosen to give optimal performance for the maintenance policy.

In this section we provide a method for constructing members from a class of problems, using linear deciders, such that these problems in many aspects could be considered to describe real world maintenance activities on systems with one or more components. Specifically, the single most important outputted value of the simulations is the expected profit at the end of the time horizon T . We also allow for the components to have evolving efficiency measures that affect the profit development over time.

Let N_c be the number of components in or problem system, this value is a free parameter in the class and can for example be chosen randomly. Each such component is granted a number of independent state variables s_{ij} , a part of the total state vector S .

For each component we allow for any number of randomly triggered events R_i . Each such event has a probability distribution r_{ij} which is a Weibull distribution with a randomly chosen scale (λ) and shape (k). To compute the probability levels as functions of the state $r(S)$ we introduce a linear dependency on the independent state variables associated with the component ($a_i \cdot S$). That is, $r(S) = r(a_i \cdot S)$.

For each component we also allow for a number of dependent state variable in a similar fashion, e_{ij} . These state variables are computed from the current independent state variables associated with the component by a function $f(S) = f(v_i \cdot S)$, where f is a randomly chosen Weibull distributions and v_i is suitable randomly chosen dependency vector. We interpret these dependent variables as a measure of efficiency for some function of the component which soon will be seen to affect simulated profit.

What remains in our example class is to define, actually parameterize, the actual event functions P_i, R_i, D .

We generate the plannable and the random events P_i, R_i in the same fashion: $S \leftarrow MS + w$, where M is a matrix of suitable dimensions and w a vector, both chosen randomly such that the new values for the dependent state values associated with the corresponding component are free to change, but only dependent on the old values for these state variables. The only other state variables allowed to change is the time t and the current profit, time must be advanced by an integer amount and the profit must be increased by a constant term.

The default event D is somewhat different. It may add a constant term to any dependent state for any component δs , the current profit must increase by addition of a term δp and may also decrease linearly by a factor c_e multiplied by any dependent state variable (efficiency measure) and the time must increase by 1 (Δt).

A NUMERICAL EXAMPLE

We selected the following example system, randomized from our stated problem class. The number of components is $N_c = 2$. Horizon time is $T = 504$. The first component has two random events (failures) and one independent state variable. The second component has only one random event and also one independent state variable. Both components have two efficiency measures.

TABLE 1. Parameters defining random events.

Component	Event	λ	k	a	M	w	δt	δp
1	1	43.385	1.6374	0.33321	0	0	4	-7.8147
1	2	66.813	1.5975	0.38519	0	0	4	-8.8816
2	1	66.487	1.1860	0.30687	0	0	2	-9.8577

TABLE 2. Parameters defining plannable events.

Component	Event	M	w	δt	δp
1	1	0	0	1	-1
1	2	0	0	1	-1
2	1	0	0	1	-1

TABLE 3. Parameters defining efficiency measures.

Component	Efficiency	λ	k	v
1	1	504	0.55770	0.36798
1	2	504	1.2823	0.70517
2	1	504	0.85805	0.071498
2	2	504	0.51121	0.34842

TABLE 4. Parameters defining the default event.

δp	c_e	δs
1	$(-0.0096735, -0.69229, -0.77131, -0.71269)$	$(0.21124, 0.78309)$

FIGURE 2. Results from optimization of linear deciders in the example problem.

In Figure 2 the expected value for the relevant state variables from 2000 iterations of a simulation of the example problem is shown. The linear deciders of the system (plannable events) was optimized using a genetic algorithm with the expected profit at the time horizon (T) as fitness function. All 6 graphs are rescaled to lie between 0 and 1 to allow placement in the same figure. The two green curves shows the two independent state variables (one for each component). The red spikes represent a high probability of having a plannable event at those points in time. Where there is an increased probability of the plannable events to occur there is also a visible drop in the corresponding state variable because M and w are both 0 for all plannable events, efficiently resetting the states (component age) to zero. The fitness can be seen to increase approximately linearly, which is a common feature of maintenance planning simulations. As can be seen, the red spikes, denoting a high probability for plannable events, decrease in height at later times. This is likely a result from the increased number of random events that has occurred on average at later times, causing a decreased determinability of the system.

Figure 3 shows the convergence of the expected profit at the end of horizon T for the genetic algorithm used. We begin with an expected profit of 377 and after 272 generations of the optimization we have reached a fairly long plateau beginning at a profit of 421 and we terminate the algorithm after 512 generations with a profit of 423.

FIGURE 3. Convergence of expected profit at the end of horizon T for the genetic algorithm used to optimize the linear deciders in the example problem.

MULTI-CRITERIA FUZZY DECISION MAKING

The health of a system depends on the health of all the components that make up the system. We consider the health of a system to be between 0 and 1 in this paper. By health equal to 0 we mean that the system fails and a health equal to 1 indicates a fully healthy system. By this consideration, we are able to formulate our decision making problem in a fuzzy environment.

In this section we provide a method by which we can find the best maintenance activity for each component in a mechanical system. To reach this goal we need to identify all criteria that affect the health of the components and also to define different maintenance activities, such as preventive, corrective, inspective and imperfect maintenance. In the present paper, we use Intuitionistic Fuzzy Sets (IFS) which was introduced by K. Atanassov in 1983 to define inclusion and non-inclusion degrees [2].

Fuzzy set and Membership function

A membership function indicates the degree of truth as an extension of evaluation. This concept was introduced by Zadeh in 1965. Fuzzy truth represents membership in vaguely defined sets. Some basic definitions of fuzzy sets, membership function and intuitionistic fuzzy sets are reviewed by Yun Shi [3], Yang[4] and KERRE [5].

Definition 1. A fuzzy set is a pair (A, m) where A is a set and $m : A \rightarrow [0, 1]$. For each $x \in A$, $m(x)$ is called the grade of membership of x in (A, m) . For a finite set $A = \{x_1, \dots, x_n\}$, the fuzzy set (A, m) is often denoted by $\{m(x_1)/x_1, \dots, m(x_n)/x_n\}$. Let $x \in A$. Then x is called fully included in the fuzzy set (A, m) if $m(x) = 1$ and is called not included if $m(x) = 0$. The set $\{x \in A | m(x) > 0\}$ is called the support of (A, m) and the set is called a kernel. x is a fuzzy member if $0 < m(x) < 1$, [4].

Definition 2. For any set X a membership function on X is any function from X to the real unit interval $[0, 1]$, the membership function which represents a fuzzy set A is denoted by μ_A . For an element x of X , the value $\mu_A(x)$ is called the membership degree of x in the fuzzy set A , [6].

According to [7] we are able to model unknown information by using an additional degree and Intuitionistic fuzzy sets (IFS)

IFS generalize fuzzy sets

Definition 3. An Intuitionistic Fuzzy Set A on a universe U is defined as an object of the following form:

$A = \{(u, \mu_A(u), \nu_A(u)) \mid u \in U\}$, where the functions $\mu_A : U \rightarrow [0, 1]$ and $\nu_A : U \rightarrow [0, 1]$ define the degree of membership and the degree of non-membership of the element $u \in U$ in A , respectively, and for every $u \in U$ we have $0 \leq \mu_A(u) + \nu_A(u) \leq 1$, [8].

According to [7] a fuzzy set can be written as:

$$\{(u, \mu_A(u), 1 - \mu_A(u)) \mid u \in U\} \quad (1)$$

IFS distribute fuzzy sets for every membership function μ and non-membership functions ν where $\nu = 1 - \mu$.

Fuzzy Implication Operators

The following table summarizes the classical binary implication:

TABLE 5. Binary implication

a	b	$a \rightarrow b$
0	0	1
0	1	1
1	0	0
1	1	1

Definition 4. A mapping $I : [0, 1]^2 \rightarrow [0, 1]$ is a fuzzy implication if it satisfies the boundary conditions: $I(0, 0) = I(0, 1) = I(1, 1) = 1$ and $I(1, 0) = 0$, [3].

A fuzzy implication can be generated by using three different approaches, R-implications, S-implications and QL-implications. In the present paper we use R-implications.

Inclusion degree function of IFS

Assume U is a finite universe and R is an implication. I_{IFS} is an inclusion degree function of IFS if R satisfies the following conditions [3]:

- $\forall a, b \in [0, 1]$ and $a \leq b \Rightarrow R(a, b) = 1$
- $R(a, b)$ is non-decreasing with respect to b and non-increasing with respect to a .

By using this definition we can write

$$I_{IFS}(A, B) = \frac{1}{|U|} \sum_{u \in U} [\lambda R(\mu_A(u), \mu_B(u)) + (1 - \lambda) R(\nu_B(u), \nu_A(u))], \lambda \in [0, 1], \quad (2)$$

where $|U|$ is the cardinality of U which can be calculated by, [9],

$$|U| = \sum_{u \in U} \frac{1 + \mu_A(u) - \nu_A(u)}{2}. \quad (3)$$

There are different methods to calculate an R-implication which was introduced by several mathematicians. we use Lukasiewicz implication:

$$R_L(a, b) = \min(1 - a + b, 1). \quad (4)$$

TOPSIS method in multiple criteria fuzzy decision making

The Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) is a well known decision making method, developed by Hwang and Yoon. According to the definition of the TOPSIS method, we need to find a positive ideal solution and also a negative ideal solution.

Since we consider our problem to be a multi criteria decision problem in a fuzzy environment we define A^+ as a Fuzzy Positive Ideal Solution and A^- as a Fuzzy Negative Ideal Solution.

We now use the TOPSIS method to calculate the distance between A^+ and A^- . Before we begin describing the method, we need to determine all criteria and alternatives. The best alternative is an alternative which has the shortest distance from A^+ and also the farthest distance from A^- .

Assume that we have a set of criteria C and a set of alternatives M :

$$C = \{C_1, C_2, \dots, C_m\}$$

$$M = \{M_1, M_2, \dots, M_n\}$$

According to [7] we assume that the alternatives and criteria are represented (using *IFS*) as:

$$M_1 = \{(C_1, \mu_{1,1}, \nu_{1,1}), (C_2, \mu_{1,2}, \nu_{1,2}), \dots, (C_m, \mu_{1,m}, \nu_{1,m})\}$$

$$M_2 = \{(C_1, \mu_{2,1}, \nu_{2,1}), (C_2, \mu_{2,2}, \nu_{2,2}), \dots, (C_m, \mu_{2,m}, \nu_{2,m})\}$$

$$\vdots$$

$$M_n = \{(C_1, \mu_{n,1}, \nu_{n,1}), (C_2, \mu_{n,2}, \nu_{n,2}), \dots, (C_m, \mu_{n,m}, \nu_{n,m})\},$$

where $\mu_{i,j}$ indicates the degree by which the alternative M_i satisfies criterion C_j , $\nu_{i,j}$ indicates the degree by which the alternative M_i does not satisfy criterion C_j .

Definition 5. A fuzzy positive ideal solution is defined as

$$A^+ = \{(C_1, \text{Max}\{\mu_{i,1}\}, \text{min}\{\nu_{i,1}\}),$$

$$(C_2, \text{Max}\{\mu_{i,2}\}, \text{min}\{\nu_{i,2}\}),$$

$$\vdots$$

$$(C_m, \text{Max}\{\mu_{i,m}\}, \text{min}\{\nu_{i,m}\})\}.$$

Definition 6. A fuzzy negative ideal solution is defined as

$$A^- = \{(C_1, \text{min}\{\mu_{i,1}\}, \text{Max}\{\nu_{i,1}\}),$$

$$(C_2, \text{min}\{\mu_{i,2}\}, \text{Max}\{\nu_{i,2}\}),$$

$$\vdots$$

$$(C_m, \text{min}\{\mu_{i,m}\}, \text{Max}\{\nu_{i,m}\})\}.$$

To calculate the distance between alternatives A^+ and A^- we define two inclusion degrees as follows:

Definition 7. The inclusion degree $D^+(M_i)$ of the positively ideal solution in alternative M_i and the inclusion degree $D^-(M_i)$ of the negatively ideal solution in alternative M_i are respectively defined as

$$D^+(M_i) = \text{Max}(I(A^+, M_i)), \quad (5)$$

$$D^-(M_i) = \text{min}(I(M_i, A^-)), \quad (6)$$

where I denotes the inclusion degree function, see Equation (2).

Definition 8. The ranking index of alternative M_i is defined as

$$p_i = \frac{D^+(M_i)}{D^-(M_i) + D^+(M_i)}, \quad (7)$$

where $0 \leq p_i \leq 1$.

If there exists $i_0 \in \{1, 2, \dots, n\}$ such that $p_{i_0} = \text{Max}\{p_1, p_2, \dots, p_n\}$ then the alternative M_{i_0} is the best alternative, [7].

A numerical example

To have a better understanding of MCFDM, we analyse internal combustion engine cooling (cooling system for short) as a real world example. In this section, we try to find the best maintenance activity for the cooling system in a typical engine.

As we mentioned in the last section, we need to define different alternatives and also identify all criteria which have a direct effect on the health of a component.

To define the maintenance activities, we need to study the company policies, the customer's perspective and requirements, which depend on the company's task operating systems.

Let us assume that M_1, M_2, M_3 are three maintenance alternatives which indicates imperfect maintenance, corrective maintenance and preventive maintenance respectively.

To identify the criteria with the highest influence we need knowledge of the mechanical properties of the component. Calendar time, mileage, failure rate, humidity, temperature, quality of roads, road dust, component age (usage), fuel quality, driving styles, environment and speed are some important criteria for a mechanical system.

Let C_1, C_2, C_3 and C_4 be the criteria that represent mileage, temperature, time and humidity and assume that these have the highest impact on the cooling system's health. As a decision maker, we want to find which of the alternatives M_i that best satisfy the criteria C_1 and C_2 or just C_3 , according to the customers perspective and the company's policies. Suppose that the relationships between alternatives and criteria are:

$$\begin{aligned} M_1 &= \{(C_1, (0.5, 0.6)), (C_2, (0.5, 0.1)), (C_3, (0.2, 0.4)), (C_4, (0.1, 0.5))\}, \\ M_2 &= \{(C_1, (0.5, 0.6)), (C_2, (0.5, 0)), (C_3, (0.3, 0.6)), (C_4, (0.5, 0.2))\}, \\ M_3 &= \{(C_1, (0.6, 0.2)), (C_2, (0.4, 0.3)), (C_3, (0.2, 0.3)), (C_4, (0.4, 0.1))\}. \end{aligned}$$

To estimate the exact coefficients for these relationships we need to perform an accurate data mining with some suitable tool such as RapidMiner. Now we can construct the positive and negative ideal solutions:

$$\begin{aligned} A_1^+ &= \{(C_1, (0.6, 0.2)), (C_2, (0.5, 0))\}, \\ A_2^+ &= \{(C_3, (0.3, 0.3))\}, \\ A_1^- &= \{(C_1, (0.5, 0.6)), (C_2, (0.4, 0.3))\}, \\ A_2^- &= \{(C_3, (0.2, 0.6))\}. \end{aligned}$$

We then calculate the inclusion degree function by using Equation (2), but before that we need to calculate R_L by using Equation (4) (Lukasiewicz implication):

$$\begin{aligned} R_L(\mu_{A_1^+}, \mu_{M_1}) &= \min(1 - 0.6 + 0.5, 1) = 0.9 \times 0.5 = 0.45, \\ R_L(v_{M_1}, v_{A_1^+}) &= \min(1 - 0.6 + 0.5, 1) = 0.6 \times (1 - 0.5) = 0.3, \\ R_L(\mu_{A_2^+}, \mu_{M_2}) &= \min(1 - 0.5 + 0.5, 1) = 1 \times 0.5 = 0.5, \\ R_L(v_{M_2}, v_{A_2^+}) &= \min(1 - 0.1 + 0, 1) = 0.9 \times (1 - 0.5) = 0.45. \end{aligned}$$

Note that λ is an optimal value between 0 and 1, we determine $\lambda = 0.5$ in this example and $|U|$ is the cardinality of U which is $|U| = 2$.

By using Equation (2) we have:

$$I(A_1^+, M_1) = \frac{1}{2} \times (0.45 + 0.3 + 0.5 + 0.45) = 0.85$$

TABLE 6. The inclusion degrees of A_1^+ in M_1, M_2 and inclusion degrees of A_1^- in M_1, M_2

	M_{11}	M_{21}	M_{31}
$I(A_1^+, M_{i1})$	0.85	0.825	0.9
$I(M_{i1}, A_1^-)$	0.925	0.9	0.875

and also:

TABLE 7. The inclusion degrees of A_2^+ in M_1, M_2 and inclusion degrees of A_2^- in M_1, M_2

	M_{11}	M_{22}	M_{32}
$I(A_2^+, M_{i2})$	0.45	0.425	0.475
$I(M_{i2}, A_2^-)$	0.45	0.475	0.425

By using Equations (5) and (6) we can calculate the inclusion degrees $D^+(M_i)$ and $D^-(M_i)$:

TABLE 8. The inclusion degrees $D^+(M_i)$ and $D^-(M_i)$

$D^+(M_i)$	0.85	0.825	0.9
$D^-(M_i)$	0.45	0.475	0.425

The ranking index of alternatives (p_i) can be calculated as follows:

$$p_1 = \frac{0.85}{0.85 + 0.45} = 0.65$$

$$p_2 = \frac{0.825}{0.825 + 0.475} = 0.634$$

$$p_3 = \frac{0.9}{0.9 + 0.425} = 0.679$$

As we see $p_3 = 0.679$ is the best alternative and indicates preventive maintenance in this case.

CONCLUSIONS

We have provided some introduction to maintenance planning optimization. Simulation and suitable optimization algorithms provides freedom in describing complex systems otherwise intractable by analytical methods. The method of finding optimal deciders defined by some parameterization of the decision logic allows for prompt establishment of a fair approximation to the optimal use of resources. Although some additional layers are useful, perhaps necessary, to establish a clear decision support for human consideration. The deciders make their decision only as a function of the current state and so provides only indirect decision patterns for planning of future activities. One good way to transition from the deciders, and the statistics collected for their decisions on a given system, is the realization that decisions closer in time are more readily understandable and predictable. As a system evolves, the inherent stochastic nature makes for example precise timing of maintenance activities more and more difficult. In light of this we may consider optimizing a plan for the close future in which activities are decided and fixed, and will be performed unless unexpected events occur, in which case a new good close future plan must be found. For a one-component system such a short-time plan may be based on 'time to first intervention', where we for all possible times to first the plannable event consider enforcing the plan, and in the simulations, if unexpected behavior occur, fall back on the deciders to evaluate a near optimal 'just in time' decision policy for the future. This provides the decision maker with a nice single-value (profit) expected result for planning the first intervention on different times, and may apply additional non-modeled preferences on the decision based on these additional simulations.

ACKNOWLEDGMENTS

This research was supported in part by the Swedish Research Council (621-2007-6338), Swedish Foundation for International Cooperation in Research and Higher Education (STINT), Royal Swedish Academy of Sciences, Royal Physiographic Society in Lund, Crafoord Foundation and Scania CV.

REFERENCES

1. N. I. M. Vasili, T. S. Hong, and M. Vasili, *Proceedings of the 2011 International Conference on Industrial and Operations Management* pp. 1131–1138 (2011).
2. G. K. Atanassov, and R. Yager, *International Journal of Systems Science* pp. 66–74 (2002).
3. Y. Shi, *A Deep Study of Fuzzy Implications*, 2009.
4. J. Yang, and J. Watada, *International Journal of Innovative Computing, Information and Control* p. 5715–5724 (2012).
5. B. D. Baets, and E. E. Kerre pp. 109–120 (1993).
6. L. Zadeh, *Information and Control* p. 338–353 (1965).
7. C. YU, and Y. Luo, *Journal of Information and Computing Science* pp. 146–152 (2008).
8. K. Atanassov, *Fuzzy Sets and Systems* pp. 87–96 (1986).
9. J. Fan, and W. Xie, *Fuzzy Sets and Systems* pp. 201–209 (1999).