

DECISION MAKING UNDER UNCERTAINTY

- PROJECT -

The University of Edinburgh
Examination number: 5870335
December 2007

1. FORMULATING THE PROBLEM

1.1 INTRODUCTION

Howard Couriers operates a large fleet of identical vans. On the anniversary of the purchase of a van, the company decides whether or not it should be replaced. It constrains this decision by limiting the maximum age that a van can have to four years and stating that a replacement van can never be older than three years.

The aim of this project is to minimise its expected discounted cost of owning and operating such a fleet over an infinite horizon. Howard Couriers discounts costs and revenues by a factor of β . We will formulate the problem as a Markov decision process.

The ideas of minimising a cost and maximising a reward are effectively the same. For this reason, we can discuss all our data in terms of rewards by simply making the costs negative.

1.2 DEFINITIONS

Definition 1.2.1: We know money devalues with time because of interest rates. An expected discounted reward is an expected reward discounted at a rate β so as to take this into consideration.

Expected discounted rewards can be planned over finite or infinite horizons. We will deal with the latter, where the most important rewards are the earlier ones.

Definition 1.2.2: A Markov decision process is a dynamic system whose future probabilistic behaviour depends on the present state of the problem and the decision taken. Its basic elements are state description, action description, immediate rewards and state transitions (which depend on the action taken) [1].

1.3 NOTATION

Before we go on to formulate the problem, it is necessary to give the notation for some key parameters so as to make its discussion simpler. So,

- let C_i be the purchase cost of a van aged i ;
- let W_i be the trade-in value of a van aged i ;
- let M_i be the expected cost of maintaining and running a van aged i for a year;
- let Q_i be the probability that a van aged i gets involved in a major accident or breakdown (and hence its state becomes U);
- and let X be the average scrap value of a van.

1.4 FORMULATION

From definition 1.2.2 we know that a Markov decision process has four basic elements. To formulate the problem, we will discuss each of these.

The states of the problem are the possible conditions in which a van can find itself on the anniversary of its purchase. These will be denoted in the following way:

- 0 if the van is new;
- 1 if the van is one year old;
- 2 if the van is two years old;
- 3 if the van is three years old;
- 4 if the van is four years old;

U if the van has been involved in a major accident or breakdown and has consequently become unusable.

The decision or action to be taken on such anniversaries will depend on its state and can be to either keep the van or replace with a new one or one whose age is one, two or three. These will be named throughout the report $ke, re0, re1, re2$ and $re3$ respectively.

The immediate reward, $r_{i,k}$, is the reward received when a process is in state i and the action k is chosen. Here, these can be the following:

- for $i = 0, 1, 2, 3$ and $k = ke$: $r_{i,k} = -M_i$;
- for $i = 0, 1, 2, 3, 4$ and $k = rej$ (where rej is the action of replacing with a van of age j): $r_{i,k} = -C_j + W_i - M_j$;
- and for $i = U$ and $k = rej$ (where rej is as before): $r_{i,k} = -C_j + X - M_j$.

Before we can finally write the optimality equation for the problem we will acknowledge the state transitions. Table 1 shows the probabilities ($p_{i,k,j}$) of the process making a transition to state j when the process is currently in state $i = 0$ and the decision k has been made. Observe how the probabilities are negligible unless state j is U , or it is the state related to rej (where rej is as before). This is because on its anniversary, a van will either become unusable after it has been involved in a major accident or breakdown (with probability Q_i), or it will simply age (with probability $1 - Q_i$). The probabilities are calculated in the same way for other states i (noting that when $i = 4$ or U , the action ke is not available).

k	p_{0,k,0}	p_{0,k,1}	p_{0,k,2}	p_{0,k,3}	p_{0,k,4}	p_{0,k,U}
ke	0	$1 - Q_0$	0	0	0	Q_0
re0	0	$1 - Q_0$	0	0	0	Q_0
re1	0	0	$1 - Q_1$	0	0	Q_1
re2	0	0	0	$1 - Q_2$	0	Q_2
re3	0	0	0	0	$1 - Q_3$	Q_3

Table 1: probabilities for state 0 to transition to other states.

So, to conclude the formulation, the required optimality equation which will give us the optimal policy is:

$$v_i = \max(r_{i,k} + \beta * (p_{i,k,0} \cdot v_0 + p_{i,k,1} \cdot v_1 + \dots + p_{i,k,U} \cdot v_U))$$

for $i = 0, 1, 2, 3, 4, U$ and $k = ke, re0, re1, re2, re3$.

2. SOLVING THE PROBLEM

2.1 DEFINITIONS

Definition 2.1.1: For a Markov decision process with present state i , a policy that delivers the maximal value is called an optimal policy [2]. We will denote its value by v_i as in §1.4. It can be calculated via a value-iteration algorithm.

Definition 2.1.2: An ε -optimal policy is one that has a stopping criterion for the algorithm at a prespecified tolerance number ε [3].

2.2 METHOD

2.2.1 The set up

We are to use value iteration to find a £0.001-optimal policy for the Markov decision process described in §1. For this, the design of a spreadsheet that will carry out this calculation in a matter of seconds is necessary. An explanation of the layout of the spreadsheet and the formulae used for the computations will precede the optimal policy found.

Firstly, let us state that, throughout the report, time is to be considered discretely in years and money in GBP (United Kingdom Pounds).

The problem data is recorded in table 2 such that:

- $C_i = 18000 - 6000\sqrt{i}$;
- $W_i = 85\% C_i$;
- $M_i = 200 + 50i^2$;
- and $Q_i = 0.01 + 0.01(\sqrt{i})^3$.

i	0	1	2	3	4
C_i	18000	12000	9514.71863	7607.69515	6000
W_i	15300	10200	8087.51083	6466.54088	5100
M_i	200	250	400	650	1000
Q_i	0.01	0.02	0.03828427	0.06196152	0.09

Table 2: Problem data.

Two more pieces of information are registered: $X = 1200$ and $\beta = 0.96$ (per year).

A second table is produced on the spreadsheet containing the data from Table 1 for every state and an additional column including the immediate rewards for each state and possible action (recall from §1.4 that this is written as $r_{i,k}$). The entries in the iteration table will be referenced to these.

Now we must examine the iteration table itself.

2.2.2 The iteration table

The first column holds the number n so that n -steps have taken place in the iteration. Values for n start with zero.

The following six columns comprise the values for the optimal rewards attained by observing the reward that is obtained depending on each possible decision (this is denoted by $v_{n,i}$ with varying i). These are the maximum of the $v_{n,i,k}$'s (with fixed i 's) columns' entries, which are also in this table (occupying twenty-eight columns which will be discussed further in a few lines). This six columns start when $n = 0$, but the entries are zero for this first n . This is needed because value iteration (also known as backward induction) needs a starting point to the calculation.

All further columns will start having entries when $n = 1$.

In between the two types of columns already discussed, the columns headlined $d_{n,i}$ (with varying i) are set. These will contain the decision to be made for each state so that the reward is maximum. They depend on the value of $v_{n,i}$, for this is the value of the maximum reward for each state. For this, the *IF* statement in *Microsoft Excel* is used. The arguments in an *IF* statement are the logical test (what are we asking?), the value if true (what do we want displayed if the answer to our question is "true"?) and the value if false (what do we want displayed if the answer is "false"?) [4]. Our problem is slightly more complicated and we have to deal with four consecutive *IF* statements. Take for example the cell that will have the value for $d_{1,0}$ (this is column $d_{n,0}$ and row such that $n = 1$). We say:

1. if $v_{2,0} = v_{2,0,ke}$, return "ke"; otherwise
2. if $v_{2,0} = v_{2,0,re0}$, return "re0"; otherwise
3. if $v_{2,0} = v_{2,0,re1}$, return "re1"; otherwise
4. if $v_{2,0} = v_{2,0,re2}$, return "re2"; otherwise
5. return "re3".

Note that there is no need to ask the computer if $v_{2,0} = v_{2,0,re3}$ since, by then, this is the only possibility.

Let us go back to the $v_{n,i,k}$'s columns. Here, all combinations of i and k are considered, giving us twenty-eight columns as we said before. These entries are directly linked with our optimality equation from the end of §1.4. Take for instance the cell $v_{2,1,re3}$ (this is column $v_{n,1,re3}$ and row such that $n = 2$). It is calculated in the following way:

$$\begin{aligned} v_{2,1,re3} &= r_{1,re3} + \beta(p_{1,re3,0} \cdot v_{1,0} + p_{1,re3,1} \cdot v_{1,1} + p_{1,re3,2} \cdot v_{1,2} + \\ &\quad + p_{1,re3,3} \cdot v_{1,3} + p_{1,re3,4} \cdot v_{1,4} + p_{1,re3,U} \cdot v_{1,U}) \\ &= 1942.30485 + 0.96(0 \cdot 7042.30485 + 0 \cdot 1942.30485 + 0 \cdot (-170.184323) + \\ &\quad + 0 \cdot (-650) + 0.93803848 \cdot (-3157.69515) + 0.06195152 \cdot (-7057.69515)) \\ &= -3736.93031 \end{aligned}$$

To finish off the iteration table, there are columns calculating the error of each optimal policy. These are obtained using the following equality:

$$\frac{\beta}{1-\beta} \max_{i \in S} |v_{n,i} - v_{(n-1),i}| = E^n$$

where S are the states of the problem and E^n is the error associated to the n th iteration.

Since we are looking for a £0.001-optimal policy, we stop the table when the error reaches 0.001. This is when $n = 445$.

Tables 3 and 4 show a few iterations from the beginning and the end of the calculation respectively.

Table 3: Iterations at the beginning of the calculation.

n	v_n,0	v_n,1	v_n,2	v_n,3	v_n,4
0	0	0	0	0	0
1	7042.30485	1942.30485	-170.184323	-650	-3157.69515
2	4525.77999	-545.617158	-1259.50138	-3913.3713	-5674.22001
3	1681.23608	-1618.76392	-4364.88905	-6329.23515	-8518.76392
4	-914.592218	-4594.92789	-6699.87359	-9059.99731	-11114.5922
5	-3531.16468	-6841.52125	-9316.44605	-11551.9925	-13731.1647

v_n,U	d_n,0	d_n,1	d_n,2	d_n,3	d_n,4
0					
-7057.69515	re3	re3	re3	ke	re3
-9574.22001	re2	ke	ke	ke	re2
-12418.7639	re1	ke	ke	ke	re1
-15014.5922	re2	ke	ke	ke	re2
-17631.1647	re2	ke	ke	ke	re2

d_n,U	v_n,0,ke	v_n,0,re0	v_n,0,re1	v_n,0,re2	v_n,0,re3
re3	-200	-2900	3050	5385.28137	7042.30485
re2	1578.21265	-1121.78735	2754.38284	4525.77999	3778.93355
re1	-810.467059	-3510.46706	1681.23608	1420.39232	1363.06969
re2	-1857.69337	-4557.69337	-1294.92789	-914.592218	-1367.69247
re2	-4711.15955	-7411.15955	-3541.52125	-3531.16468	-3859.68763

v_n,1,ke	v_n,1,re0	v_n,1,re1	v_n,1,re2	v_n,1,re3	v_n,2,ke
-250	-8000	-2050	285.281374	1942.30485	-400
-545.617158	-6221.78735	-2345.61716	-574.220008	-1321.06645	-1259.50138
-1618.76392	-8610.46706	-3418.76392	-3679.60768	-3736.93031	-4364.88905
-4594.92789	-9657.69337	-6394.92789	-6014.59222	-6467.69247	-6699.87359
-6841.52125	-12511.1595	-8641.52125	-8631.16468	-8959.68763	-9316.44605

v_n,2,re0	v_n,2,re1	v_n,2,re2	v_n,2,re3	v_n,3,ke	v_n,3,re0
-10112.4892	-4162.48917	-1827.20779	-170.184323	-650	-11733.4591
-8334.27652	-4458.10633	-2686.70918	-3433.55562	-3913.3713	-9955.24647
-10722.9562	-5531.25309	-5792.09685	-5849.41948	-6329.23515	-12343.9262
-11770.1825	-8507.41706	-8127.08139	-8580.18164	-9059.99731	-13391.1525
-14623.6487	-10754.0104	-10743.6538	-11072.1768	-11551.9925	-16244.6187

Table 3 (continued): Iterations at the beginning of the calculation.

v_n,3,re1	v_n,3,re2	v_n,3,re3	v_n,4,re0	v_n,4,re1	v_n,4,re2
-5783.45912	-3448.17774	-1791.15427	-13100	-7150	-4814.71863
-6079.07628	-4307.67913	-5054.52557	-11321.7873	-7445.61716	-5674.22001
-7152.22304	-7413.0668	-7470.38943	-13710.4671	-8518.76392	-8779.60768
-10128.387	-9748.05134	-10201.1516	-14757.6934	-11494.9279	-11114.5922
-12374.9804	-12364.6238	-12693.1467	-17611.1595	-13741.5212	-13731.1647

v_n,4,re3	v_n,U,re0	v_n,U,re1	v_n,U,re2	v_n,U,re3	i=0
-3157.69515	-17000	-11050	-8714.71863	-7057.69515	7042.30485
-6421.06645	-15221.7873	-11345.6172	-9574.22001	-10321.0664	2516.52485
-8836.93031	-17610.4671	-12418.7639	-12679.6077	-12736.9303	2844.54392
-11567.6925	-18657.6934	-15394.9279	-15014.5922	-15467.6925	2595.82829
-14059.6876	-21511.1595	-17641.5212	-17631.1647	-17959.6876	2616.57246

i=1	i=2	i=3	i=4	i=U	E_n
1942.30485	170.184323	650	3157.69515	7057.69515	169384.684
2487.922	1089.31706	3263.3713	2516.52485	2516.52485	78320.9111
1073.14677	3105.38767	2415.86386	2844.54392	2844.54392	74529.3041
2976.16396	2334.98454	2730.76216	2595.82829	2595.82829	71427.9351
2246.59336	2616.57246	2491.99516	2616.57246	2616.57246	62797.7391

Table 4: Iterations at the end of the calculation.

n	v_n,0	v_n,1	v_n,2	v_n,3	v_n,4
443	-64833.8318	-68203.9912	-70619.1132	-72914.4625	-75033.8318
444	-64833.8319	-68203.9913	-70619.1132	-72914.4625	-75033.8319
445	-64833.8319	-68203.9913	-70619.1133	-72914.4625	-75033.8319

v_n,U	d_n,0	d_n,1	d_n,2	d_n,3	d_n,4
-78933.8318	re2	ke	ke	ke	re2
-78933.8319	re2	ke	ke	ke	re2
-78933.8319	re2	ke	ke	ke	re2

d_n,U	v_n,0,ke	v_n,0,re0	v_n,0,re1	v_n,0,re2	v_n,0,re3
re2	-65778.838	-68478.838	-64903.9912	-64833.8318	-65222.1576
re2	-65778.838	-68478.838	-64903.9913	-64833.8319	-65222.1577
re2	-65778.8381	-68478.8381	-64903.9913	-64833.8319	-65222.1577

Table 4 (continued): Iterations at the end of the calculation.

$v_{n,1,ke}$	$v_{n,1,re0}$	$v_{n,1,re1}$	$v_{n,1,re2}$	$v_{n,1,re3}$	$v_{n,2,ke}$
-68203.9912	-73578.838	-70003.9912	-69933.8318	-70322.1576	-70619.1132
-68203.9913	-73578.838	-70003.9913	-69933.8319	-70322.1577	-70619.1132
-68203.9913	-73578.8381	-70003.9913	-69933.8319	-70322.1577	-70619.1133

$v_{n,2,re0}$	$v_{n,2,re1}$	$v_{n,2,re2}$	$v_{n,2,re3}$	$v_{n,3,ke}$	$v_{n,3,re0}$
-75691.3272	-72116.4804	-72046.321	-72434.6468	-72914.4625	-77312.2971
-75691.3272	-72116.4804	-72046.321	-72434.6468	-72914.4625	-77312.2972
-75691.3273	-72116.4805	-72046.3211	-72434.6469	-72914.4625	-77312.2972

$v_{n,3,re1}$	$v_{n,3,re2}$	$v_{n,3,re3}$	$v_{n,4,re0}$	$v_{n,4,re1}$	$v_{n,4,re2}$
-73737.4503	-73667.2909	-74055.6167	-78678.838	-75103.9912	-75033.8318
-73737.4504	-73667.291	-74055.6168	-78678.838	-75103.9913	-75033.8319
-73737.4504	-73667.291	-74055.6168	-78678.8381	-75103.9913	-75033.8319

$v_{n,4,re3}$	$v_{n,U,re0}$	$v_{n,U,re1}$	$v_{n,U,re2}$	$v_{n,U,re3}$	$i=0$
-75422.1576	-82578.838	-79003.9912	-78933.8318	-79322.1576	4.3881E-05
-75422.1577	-82578.838	-79003.9913	-78933.8319	-79322.1577	4.2126E-05
-75422.1577	-82578.8381	-79003.9913	-78933.8319	-79322.1577	4.0441E-05

$i=1$	$i=2$	$i=3$	$i=4$	$i=U$	E_n
4.3881E-05	4.3881E-05	4.3881E-05	4.3881E-05	4.3881E-05	0.00105314
4.2126E-05	4.2126E-05	4.2126E-05	4.2126E-05	4.2126E-05	0.00101102
4.0441E-05	4.0441E-05	4.0441E-05	4.0441E-05	4.0441E-05	0.00097058

2.2.3 Performing sensitivity analyses

To investigate how sensitive the optimal policy is to changes in the problem parameters, we create sensitivity tables for each state i and parameters C_i , W_i , M_i and Q_i .

Table 5 and table 6 show examples of the sensitivity analyses for Q_0 and C_3 .

Q_0		0	0.25	0.5	0.75	1
d_445_0	re2	re2	re2	re2	re2	re2
d_445_1	ke	ke	ke	ke	ke	Ke
d_445_2	ke	ke	ke	ke	ke	Ke
d_445_3	ke	ke	ke	ke	ke	Ke
d_445_4	re2	re2	re2	re2	re2	re2
d_445_U	re2	re2	re2	re2	re2	re2
E_445	0.001	0.001	0.001	0.001	0.001	0.001

Table 5: Sensitivity analysis for Q_0 .

C_3		5608	6608	7608	8608	9608
d_445_0	re2	re3	re3	re2	re2	re2
d_445_1	ke	re3	ke	ke	ke	Ke
d_445_2	ke	re3	ke	ke	ke	Ke
d_445_3	ke	ke	ke	ke	re2	re2
d_445_4	re2	re3	re3	re2	re2	re2
d_445_U	re2	re3	re3	re2	re2	re2
E_445	0.001	0.001	0.001	0.001	0.001	0.001

Table 6: Sensitivity analysis for C_3 .

2.3 SOLUTIONS

2.3.1 £0.001-optimal policy

The £0.001-optimal policy for the Markov decision process discussed is to keep the vans that are in states 1, 2 and 3, and to replace with a two-year-old van the ones in states 0, 4 and U . This can be seen from the columns for $d_{n,1}$, $d_{n,2}$, $d_{n,3}$, $d_{n,0}$, $d_{n,4}$, $d_{n,U}$, for $n = 445$ in table 4 respectively.

Assuming that currently 10% of the fleet is new, 20% is one year old, 40% is two years old and 30% is three years old, the minimum expected discounted cost of owning and operating the fleet over an infinite horizon is £70246.17 N (to the nearest pence), where N is the number of vans in the fleet. Figure 1 shows the formula and solution for this calculated on *Microsoft Excel*.

min expected cost	$=-(10\%*B483)+(20\%*C483)+(40\%*D483)+(30\%*E483)$
	£70,246.17

Figure 1: Calculation of the minimum expected cost on *Microsoft Excel*.

2.3.2 Sensitivity of parameters

Recall the given values for the parameters from table 2 and, also, $X = £1200$.

Testing on purchase cost has been done in £1000-intervals. We assume $C_0 > C_1 > C_2 > C_3$ (we will not consider C_4 since we are never going to buy a four-year-old van). Changing C_0 to anything more expensive than £17000 will not alter the optimal policy. This is because our optimal policy claims we should replace all new vans with two-year-old ones since the cost is already fairly expensive. If the price of the new vans was even higher, it would not make sense for us to suddenly want to buy them. However, if $C_1 < C_0 \leq £16000$, our optimal policy would change. The lower this upper bound, the higher the chance we will want to have all our vans replaced for new ones. The range of C_1 that leaves our optimal policy unchanged is $[£12000, £14000]$. If it is strictly lower than £12000, the decision of replacing vans in states 0, 4 and U for a two-year-old van becomes *re1*. If it is strictly higher than £14000, the actions for states 1 and 0 become *ke* and *re2* respectively. In the case of C_2 , the range for which the optimal policy remains unchanged is even smaller. This is to be expected, for it has a much shorter range of possible change than any other C_i . From table 6 we can observe that decreasing C_3 will make the optimal policy tend to $\{re3, re3, re3, ke, re3, re3\}$, this is, deal only with vans of age three.

Testing on trade-in value has been done in £50-intervals. For W_0 we find ourselves in the same situation as above. The optimal policy will only change once we decrease the trade-in value of a new van (here, we have to decrease W_0 by at least £950). For W_1, W_2 and W_3 , the optimal policy starts changing shortly after we increase their values. In the case of W_4 , even though the range for which the optimal policy is unchanged is similar to the ones of W_i for $i = 1, 2$ and 3, it also affects the optimal policy when it is decreased.

Testing on maintenance and running costs has been done in £50-intervals. Since these costs increase with the age of the vehicle, $M_0 < M_1 < M_2 < M_3 < M_4$. Given these constraints, sensitivity tables for M_0, M_1 and M_4 show that changes in maintenance and running costs have no effect on our optimal policy. On the other hand, when $£550 \leq M_2 < £650$ or $£800 \leq M_3 < £1000$, the optimal policy becomes $\{re1, ke, ke, ke, re1, re1\}$.

Testing on probability of a van getting involved in a major accident or breakdown has been done to the nearest 0.1%. The risk of one of Howard Couriers' van getting involved in a major accident or breakdown increases with its age, so $Q_0 < Q_1 < Q_2 < Q_3 < Q_4$. Changing the probabilities according to these constraints leaves the optimal policy unchanged except for vans aged one. Table 7 shows the sensitivity analysis for such a case.

Finally, it is worth mentioning that the scrap value of a van is an average and therefore is independent of the van's age. This implies that the optimal policy is unaffected when the value of X is changed.

Q_1		0.010	0.011	0.012	0.13
d_445_0	re2	re1	re1	re2	re2
d_445_1	ke	ke	ke	ke	ke
d_445_2	ke	ke	ke	ke	ke
d_445_3	ke	ke	ke	ke	ke
d_445_4	re2	re1	re1	re2	re2
d_445_U	re2	re1	re1	re2	re2
E_445	0.001	0.001	0.001	0.001	0.001

Table 7: Sensitivity analysis for Q_1 .

3. STRENGTHS AND WEAKNESSES OF THE MODEL

The simplicity of the formulated model is both its main strength and weakness. We shall discuss a few of the assumptions and data used in it to see why this is so.

Howard Couriers starts off with a fleet of vans. These are already assumed to have been bought at the same time and to be identical to each other. If this was not the case, the number of states of the problem would increase. Similarly, the fact that all vehicles can only be x years old (where x is a whole number, this is, $x = 0, 1, 2, \dots$) and not, for example, 78 days old, also restrains the number of states.

Having the age of the vans not only as a positive-integer value but as positive rational numbers too would also improve the practicality of the model, for it would allow the company to replace vans in state U just after the major accident or breakdown has occurred, and not only on the anniversary of its purchase.

The model assumes price stability throughout the company's life. If inflation and deflation were taken into consideration, the optimal policy would remain unchanged, but the minimum expected discounted cost of owning and operating the fleet (calculated in §2.3.1) would vary greatly (increasing with inflation and decreasing with deflation).

Howard Couriers makes no inversion in investigating cheaper sales, even though it is trying to minimise its costs. This could mean there is collusion between van providers. Collusion is an explicit or implicit agreement between existing firms to avoid competition [5]. In the car-dealer business, this is highly unrealistic.

All parameters in the problem are taken to be constant, making the model easier to formulate. However, it is likely that these will change with time.

It is assumed the company purchases vans from a never-ending stock. Depending on the length of time the company will be running its business, it would be acceptable to assume the vehicles are going to change over time, for upgrades are being made on a regular basis, and the original vans Howard Couriers was dealing with will stop being available.

Another assumption is that all vans are identical. This does not imply that in the long run, they will remain having identical characteristics. For this reason, maintenance and running costs and trade-in values should be different not only according to their age, but to their particular condition too. In the same way, scrap value should depend on type of problem arisen.

To improve the model we could establish the ineffectiveness of replacing a van with another of its own age. By doing this, we would not change the formulation of the problem enormously, but our iteration table would have less $v_{n,i,k}$ -columns and still leave the optimal policy unchanged.

In conclusion, the model is too simplistic but, with some minor adjustments such as the addition of states or the variation of data according to time, it could improve noticeably.

4. ADVANTAGES AND DISADVANTAGES OF ADDING INFORMATION

Howard Couriers keeps detailed records of the maintenance and repairs carried out on its fleet of vehicles. The maintenance records for an individual van include the number of unscheduled maintenance interventions, the cumulative cost of unscheduled maintenance interventions and the type of each unscheduled maintenance intervention. An unscheduled maintenance intervention is defined as any maintenance or repair action that is not part of regular servicing, for example, the replacement of a flat tyre.

The disadvantage of including these records into the model formulated and solved earlier is that it would make it more complex, for these new parameters would suppose, for instance, an addition in states and the reformulation of the immediate rewards. It would also have an effect on the probability of getting involved in a major accident or suffering a serious breakdown associated to a van, since a van that has just had its tyres changed because of a puncture will have a lower probability of transitioning to state U than one with same original state but no unscheduled repair.

More specifically, the cumulative cost of unscheduled interventions would increase the minimum expected discounted cost of owning and operating the company's fleet of vehicles. But this would have no effect on the optimal policy unless vans were aged 2 or 3.

The number of unscheduled maintenance interventions, and the types of these, would only affect the optimal policy if the states were changed so that specific conditions of vans were regarded. However, this is not how the problem is formulated in §1, so the optimal policy would again remain unchanged.

But whilst the reformulation of the problem might be seen as a disadvantage for the analyst, in the long run, it would be beneficial for the company.

To conclude, we know that with the extra information, the model becomes more realistic, so it starts to overcome its main disadvantage (discussed in §3).

REFERENCES

- [1] DMU lecture notes, 2007.
- [2] Thie, P. 1983, *Markov Decision Processes*, Comap, Inc.
- [3] Tijms, H. C. 1994, *Stochastic Models: An algorithmic approach*, John Wiley & Sons.
- [4] <http://personal-computer-tutor.com/if1.htm>
- [5] Begg, D., Fischer, S., Dornbusch, R. 2003, *Foundations of Economics*, McGraw Hill.