Transactions of the Nigerian Association of Mathematical Physics
Volume 1, (November, 2015), **pp 255 – 264**
© Trans. of NAMP

Excel Solution Implementation Templates for Machine Replacement Problems with Age and Decision Period Variable Data

Ukwu Chukwunenye

Department of Mathematics, University of Jos, P.M.B. 2084, Jos, Plateau State, Nigeria.

Abstract

This paper deployed dynamic programming recursions to design solution implementation templates for the determination of the optimal replacement policies in machine replacement problems, with pertinent data given as functions of machines ages and the decision periods of the planning horizon. These templates circumvent the inherent tedious and prohibitive manual computations associated with dynamic programming formulations and may be optimally appropriated for sensitivity analyses on such models.

Keywords: Ages, Data, Excel, Equipment, Feasible, Dynamic, Optimal, Policy, Programming, Recursions, Replacement, Solution, Template, Variable.

1.0 Introduction

Consider the problem of researching an optimal Equipment Replacement policy over an $n-$period planning horizon. At the start of each year a decision is made whether to keep the equipment in service an extra year or to replace it with a new one at some salvage value. As remarked by Taha [1]"the determination of the feasible values for the age of the machine at each stage is somewhat tricky". The determination process for the optimal replacement age is normally initiated with network diagrams with machine ages on the vertical axis and decision years on the horizontal axis. Then dynamic recursions are formulated as functions of the decision years, the corresponding feasible equipment ages, the problem data and the cash-flow profile. For the problem to be well-posed, it is pertinent that the cash-flow profile be provided for $\max\{m,n\}$ years, where m is the mandatory equipment replacement age. Unfortunately network diagrams are unwieldy, cumbersome and prone to errors, especially for large problem instances; consequently the integrity of the desired optimal policies may be compromised. Fortunately, Ukwu [2] developed computational formulas for the feasible states corresponding to each decision year in a certain class of equipment Replacement problems, thereby eliminating the drudgery and errors associated with the drawing of network for such determination. [2]went further to design a prototypical solution template for optimal solutions to such problems, complete with an exposition on the interface and solution process. Finally [2] solved eight illustrative examples, pointing out that manual solutions to these eight problems, starting with the determination of the feasible ages for each decision year via network diagrams would, in the best case scenario, consume no less than 5 hours. These would contrast sharply with the Excel implementations that took at most thirty minutes, demonstrating the efficiency, power and utility of the solution template prototype. In general, the template could be deployed to solve each ERP problem in less than 10 percent of the time required for the manual generation of the alternate optima.

However, the scope of the template in [2] is limited to prescribed pertinent data that depend only on the age of the machine with stationary values throughout the planning horizon and beyond. The challenge then is to develop solution templates that will circumvent the complexity imposed by variable data in general. This paper addresses this issue by modifying the dynamic recursions, manipulating and presenting pertinent variable data in the form appropriate for the template prototype in [2], with the data table now given as functions of the decision periods. Evidently the value $V_i(0) = r_i(0) - I_i - c_i(0)$ will need to be stored in Excel in relative referenced cells as contrasted with the stationary value $V(0) = r(0) - I - c(0)$ being stored in the fixed reference $F$4 in [2]. These are the major contributions of this paper.

Corresponding author: Ukwu Chukwunenye, E-mail: chukwu@hotmail.com, Tel.: +2347052765751

2.0 Theoretical Analysis

In this section, the problem data, working definitions, elements of the DP model and the dynamic programming (DP) recursions are laid out as follows:

Equipment Starting age $= t_1$

Equipment Replacement age $= m$

S_i = The set of feasible equipment ages (states) in decision period i (say year i), $i \in \{1, 2, \ldots, n\}$

$r_i(t)$ = annual revenue from a $t-$ year old equipment during the decision year i.

$c_i(t)$ = annual operating cost of a $t-$ year old equipment during the decision year i.

$s_i(t)$ = salvage value of a $t-$ year old equipment during the decision year i; $t = 0, 1, \ldots, m$

I_i = cost of acquiring a new equipment during the decision year i.

The elements of the DP are the following:

1. Stage i, represented by year $i, i \in \{1, 2, \ldots, n\}$
2. The alternatives at stage (year) i. These call for keeping or replacing the equipment at the beginning of year i
3. The state at stage (year) i, represented by the age of the equipment at the beginning of year i.

Let $f_i(t)$ be the maximum net income for years $i, i+1, \ldots, n-1, n$ given that the equipment is t years old at the beginning of year i.

Note: The definition of $f_i(t)$ starting from year i to year n implies that backward recursion will be used. Forward recursion would start from year 1 to year i.

3.0 Results and Discussion

The following theorem is applicable to the backward recursive procedure:

3.1 Theorem 1

For $i \in \{1, 2, \cdots, n-1\}$,

$$f_i(t) = \max \begin{cases} r_i(t) - c_i(t) + f_{i+1}(t+1); \text{ IF KEEP} \\ r_i(0) + s_i(t) - I_i - c_i(0) + f_{i+1}(1); \text{ REPLACE} \end{cases}$$

$$f_{n+1}(x) = s_{n+1}(x), \quad i = 0, 1, \ldots, n-1, \quad x = \text{age of machine at the start of period } n+1$$

Proof

Decision: KEEP

$r_i(t) - c_i(t) =$ net revenue (income) from a t-year old machine during the decision year i. Then the equipment age advances to $t+1$ years and hence $f_{i+1}(t+1) =$ maximum income for years $i+1, \ldots, n$ given that the equipment is $t+1$ years old at the start of year $i+1$.

Decision: REPLACE

$r_i(0) = 1-$ year revenue from a new equipment (age 0) during the decision year i.

$c_i(0)$ = cost of operating a new equipment for 1 year (from the start of year i to the end of year i)

I_i = cost of a new equipment during the decision year i.

$s_i(t)$ = salvage cost for a $t-$ year old equipment during the decision year i.

$$\text{Net income} = \underbrace{r_i(0) - c_i(0)}_{\substack{\text{net income for operating} \\ \text{the new equipment from the} \\ \text{beginning of year } i \text{ to the} \\ \text{end of year } i\text{. The age of the} \\ \text{equipment then advances to 1 year}}} + s_i(t) - I_i + \underbrace{f_{i+1}(1)}_{\substack{\text{max net income for} \\ \text{years } i+1,\ldots,n \text{ given} \\ \text{that the machine is 1-year} \\ \text{old at the start of year } i+1}}$$

$f_{n+1}(x) = s_{n+1}(x)$ or $f_{n+1}(.) = s_{n+1}(.) \Rightarrow$ sell off the equipment at the end of the planning horizon at price $s_{n+1}(.)$, regardless of its age, with no further income realized from the beginning of year $n+1$, since the planning horizon length is n years. Therefore the recursive equation is correct. This completes the proof.

3.2 Pertinent Remarks on the DP Recursions

For $i \in \{1, 2, \cdots, n\}$, $f_i(t)$ may be identified as $f_i(t) = \max\limits_{\{K,R\}} \{f_i^K(t), f_i^R(t)\}$, where

$f_i^K(t) = r_i(t) - c_i(t) + f_{i+1}(t+1)$ and $f_i^R(t) = r_i(0) + s_i(t) - I_i - c_i(0) + f_{i+1}(1)$

For $i \in \{1, 2, \cdots, n\}$ and $t \in S_i$ the optimal decision may be identified as $D_i(t)$, where

$$D_i(t) = \underset{\{K,R\}}{\text{argmax}}\, g_i(t,K,R);\ g_i(t,K,R) = \begin{cases} f_i^K(t), \text{ if Decision is KEEP} \\ f_i^R(t), \text{ if Decision is REPLACE} \end{cases}$$

Define

$$x_i = \begin{cases} 1, \text{ if decision is REPLACE in stage } i \text{ (start of decision year } i) \\ 0, \text{ if decision is KEEP in stage } i \text{ (start of decision year } i) \end{cases}$$

Then

$$g_i(t,K,R) = f_i(t) = (1 - x_i) f_i^K(t) + x_i f_i^R(t),\ i \in \{1, 2, \cdots, n\}$$

If the revenue profile is not given, then $r_i(t)$ may be set identically equal to zero, in which case

$-f_i(t) =$ minimum cost associated with operating the equipment from the start of decision year i to the end of decision year n.

If the variable cost profile is not given then $c_i(t)$ may be set identically equal to zero, in which case

$f_i(t) =$ maximum net revenue from the start of decision year i to the end of decision year n.

If the cost profile is not given then $c_i(t)$ and I_i may be set identically equal to zero, in which case

$f_i(t) =$ maximum accrueable revenue from the start of decision year i to the end of decision year n.

3.3 Application Problems

A company reviews the status of heavy equipment at the end of each year, and a decision is made to either to keep the equipment an extra year or to replace it. However, equipment that has been in service for 3 years must be replaced. The company wishes to develop a replacement policy for its fleet over the next ten years. The following table provides the pertinent data. The equipment is new at the start of year 1.

Table 1: Pertinent Data for Optimal Policy and Reward Determination

		Maintenance cost ($) for given age (yr.)			Salvage value ($) for given age (yr.)		
Purchase yr.	Purchase price	0	1	2	1	2	3
1	10,000	200	500	600	9,000	7,000	5,000
2	12,000	250	600	680	11,000	9,500	8,000
3	13,000	280	550	600	12,000	11,000	10,000
4	13,500	320	650	700	12,000	11,500	11,000
5	13,800	350	590	630	12,000	11,800	11,200
6	14,200	390	620	700	12,500	12,000	11,200
7	14,800	410	600	620	13,500	12,900	11,900
8	15,200	430	670	700	14,000	13,200	12,000
9	15,500	450	700	730	15,500	14,500	13,800
10	16,000	500	710	720	15,800	15,000	14,500

Devise Solution Templates and consequently anOptimal Replacement Policy for the Equipment Fleet.

3.3.1 Problem

Solve the above problem for $t_1 \in \{0, 1, \cdots\}$, using dynamic programming recursions.

Solution

The given data must be restructured to depend only on the age of the machine for a given decision year, starting from the terminal year. Note that the revenue profile is not given; so this is a minimization problem. Set $r_i(t) = 0$, uniformly in $i \in \{1, 2, \cdots, 10\}$ and $t \in \{1, 2, 3\}$ and organize the data in three – dimension format, as follows:

Table 2: Pertinent Data for Decision Year 10

Year 10: Purchase price = $16,000			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	500	-
1	0	710	15800
2	0	720	15000
3	0	0	14500

Table 3: Pertinent Data for Decision Year 9

Year 9: Purchase price = $15,500			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	450	-
1	0	700	15500
2	0	730	14500
3	0	0	13800

Table 4: Pertinent Data for Decision Year 8

Year 8: Purchase price = $15,200			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	430	-
1	0	670	14000
2	0	700	13200
3	0	0	12000

Table 5: Pertinent Data for Decision Year 7

Year 7: Purchase price = $14,800			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	410	-
1	0	600	13500
2	0	620	12900
3	0	0	11900

Table 6: Pertinent Data for Decision Year 6

Year 6: Purchase price = $14,200			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	390	-
1	0	620	12500
2	0	700	12000
3	0	0	11200

Table 7: Pertinent Data for Decision Year 5

Year 5: Purchase price = $13,800			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	350	-
1	0	590	12000
2	0	630	11800
3	0	0	11200

Table 8: Pertinent Data for Decision Year 4

Year 4: Purchase price = $13,500			
Age: *t* yrs.	Revenue: *r*(*t*) ($)	Maintenance cost: *c*(*t*) ($)	Salvage value: *s*(*t*) ($)
0	0	320	-
1	0	650	12000
2	0	700	11500
3	0	0	11000

Table 9: Pertinent Data for Decision Year 3

Year 3: Purchase price = $13,000			
Age: t yrs.	Revenue: $r(t)$ ($)	Maintenance cost: $c(t)$ ($)	Salvage value:$s(t)$ ($)
0	0	280	-
1	0	550	12000
2	0	600	11000
3	0	0	10000

Table 10: Pertinent Data for Decision Year 2

Year 2: Purchase price = $12,000			
Age: t yrs.	Revenue: $r(t)$ ($)	Maintenance cost: $c(t)$ ($)	Salvage value: $s(t)$ ($)
0	0	250	-
1	0	600	11000
2	0	680	9500
3	0	0	8000

Table 11: Pertinent Data for Decision Year 1

Year 1: Purchase price = $10,000			
Age: t yrs.	Revenue: $r(t)$ ($)	Maintenance cost: $c(t)$ ($)	Salvage value: $s(t)$ ($)
0	0	200	-
1	0	500	9000
2	0	600	7000
3	0	0	5000

To determine S_i for $i \in \{1, 2, \cdots, 10\}$, invoke the following result from Ukwu [ ]:

3.4 Corollary 1

If $t_1 < m$, then for $i \in \{2, \ldots, n\}$,

$$S_i = \begin{cases} \left\{ \min\limits_{2 \le j \le i} \{ j-1, m \} \right\} \cup \{ i - 1 + t_1 \}, \text{ if } i \le m + 1 - t_1 \\ \left\{ \min\limits_{2 \le j \le i} \{ j-1, m \} \right\}, \text{ if } i > m + 1 - t_1 \end{cases}$$

In the given problem,

$$t_1 = 0, \ m = 3, \ n = 10, \ i \in \{1, 2, \cdots, 10\}, S_1 = \{0\} \Rightarrow m + 1 - t_1 = 4$$

$$\Rightarrow S_2 = \{1\}, S_3 = \{1, 2\}, S_i = \{1, 2, 3\}, i \in \{4, 5, \cdots, 10\}.$$

Using the second of the relations:

$\min\{ f(x) \} = -\min\{ -f(x) \}$, $\min\{ -f(x) \} = -\max\{ f(x) \}$, the template for the implementation of $f_i(t)$ in theorem 1 may be exploited to obtain the optimal solutionto the problem noting thatthe optimal objective value of the given problem is $-f_i(t)$: the minimum cost of operating each machine in the fleet.

Table 12: Template Solution of the Equipment Replacement Problem with Starting Age 0

Equipment Replacement Problem Solution Template: (New) 0 - year Starting age Implementation

Replacement Age = 3 yrs

Given Data				Stage 10 Computations in $1000				
	I =	16000		V(0) = r(0)-c(0)-I	-16.5	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f10(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)+s(t+1)-c(0)	)+s(t) + s(1)-c(0) - I			t
0	0	500	-					
1	0	710	15800	14.29	15.1	15.1	R	1
2	0	720	15000	13.78	14.3	14.3	R	2
3	0	0	14500	Must Replace	13.8	13.8	R	3

Given Data				Stage 9 Computations in $1000				
	I =	15500		V(0) = r(0)-c(0)-I	-15.95	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f9(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f10(t+1	V(0)+s(t)+f10(1)			t
0	0	450	-					
1	0	700	15500	13.6	14.65	14.65	R	1
2	0	730	14500	13.07	13.65	13.65	R	2
3	0	0	13800	Must Replace	12.95	12.95	R	3

Given Data				Stage 8 Computations in $1000				
	I =	15200		V(0) = r(0)-c(0)-I	-15.63	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f8(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f9(t+1)	V(0)+s(t)+f9(1)			t
0	0	430	-					
1	0	670	14000	12.98	13.02	13.02	R	1
2	0	700	13200	12.25	12.22	12.25	K	2
3	0	0	12000	Must Replace	11.02	11.02	R	3

Given Data				Stage 7 Computations in $1000				
	I =	14800		V(0) = r(0)-c(0)-I	-15.21	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f7(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f8(t+1)	V(0)+s(t)+f8(1)			t
0	0	410	-					
1	0	600	13500	11.65	11.31	11.65	K	1
2	0	620	12900	10.4	10.71	10.71	R	2
3	0	0	11900	Must Replace	9.71	9.71	R	3

Given Data				Stage 6 Computations in $1000				
	I =	14200		V(0) = r(0)-c(0)-I	-14.59	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f6(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f7(t+1)	V(0)+s(t)+f7(1)			t
0	0	390	-					
1	0	620	12500	10.09	9.56	10.09	K	1
2	0	700	12000	9.01	9.06	9.06	R	2
3	0	0	11200	Must Replace	8.26	8.26	R	3

Given Data				Stage 5 Computations in $1000				
	I =	13800		V(0) = r(0)-c(0)-I	-14.15	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f5(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f6(t+1)	V(0)+s(t)+f6(1)			t
0	0	350	-					
1	0	590	12000	8.47	7.94	8.47	K	1
2	0	630	11800	7.63	7.74	7.74	R	2
3	0	0	11200	Must Replace	7.14	7.14	R	3

Given Data				Stage 4 Computations in $1000				
	I =	13500		V(0) = r(0)-c(0)-I	-13.82	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f4(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f5(t+1)	V(0)+s(t)+f5(1)			t
0	0	320	-					
1	0	650	12000	7.09	6.65	7.09	K	1
2	0	700	11500	6.44	6.15	6.44	K	2
3	0	0	11000	Must Replace	5.65	5.65	R	3

Given Data				Stage 3 Computations in $1000				
	I =	13000		V(0) = r(0)-c(0)-I	-13.28	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f3(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f4(t+1)	V(0)+s(t)+f4(1)			
								t
0	0	280	-					
1	0	550	12000	5.89	5.81	5.89	K	1
2	0	600	11000	5.05	4.81	5.05	K	2
3	0	0	10000					

Given Data				Stage 2 Computations in $1000				
	I =	12000		V(0) = r(0)-c(0)-I	-12.25	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f2(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f3(t+1)	V(0)+s(t)+f3(1)			
								t
0	0	250	-					
1	0	600	11000	4.45	4.64	4.64	R	1
2	0	680	9500					
3	0	0	8000					

Given Data				Stage 1 Computations in $1000				
	I =	10000		V(0) = r(0)-c(0)-I	-10.2	Optimal Solution		
Age, t	Revenue,	Operating	Salvage	K	R	f1(t)	Decision	
(yr)	r(t) ($)	cost, c(t) ($)	value, s(t) ($)	r(t)-c(t)+f2(t+1)	V(0)+s(t)+f2(1)			
								t
0	0	200		4.44	-5.56	4.44	K	0
1	0	500	9000					
2	0	600	7000					
3	0	0	5000					

Optimal Policy: 0K1R1K2K3R1K2R1R1R1R1S or KRKKRKRRRRS
Optimal value: $4,440.00 (disregarding the initial purchasing cost of $10,000)
Actual Optimal value: $4,440.00 − $10,000 = − $5,560 (initial purchasing cost of $10,000 factored in)
This implies that the minimum cost incurred is $5,560
Note that for an n - year planning horizon problem, there should be $2(n+1)$ such decision symbols starting with t_1 and terminating with the symbol S.

3.5 Interpretation of the concatenated symbols 0K1R1K2K3R1K2R1R1R1R1S

(i) The new machine is kept in use for 1 year: from the beginning of year 1 to the end of year 1; the age then advances to 1.

(ii) The machine is replaced at the beginning of year 2; the age of the replacement machine becomes 1 after one year of operation

(iii) The machine is kept for kept further for two consecutive years, bringing the age to 3, thereupon it is replaced, having attained the mandatory replacement age. After 1 year of operation the age becomes 1.

(iv) The 1 - year machine is kept for another year, culminating in the age being 2.

(v) The 2 - year machine is replaced. Thereafter each replacement machine is replaced with a new one until the end of the planning horizonwhen the terminal 1 – year machine is sold off.

The simplest interpretation is as follows:

Replace the machines at the beginning of the decision years 2,5,7,8,9,10 and then sell off the asset.
For $t \in \{1, 2, \cdots\}$, the states corresponding to the decision years 1 to 10, the optimal policies and the optimal returns are obtained as follows:

$t_1 = 1 \Rightarrow m + 1 - t_1 = 3 \Rightarrow S_1 = \{1\}, S_2 = \{1, 2\}, S_i = \{1, 2, 3\},$ for $i \in \{3, 4, \cdots, 10\}.$

Plug each S_i into the template under column I, to automatically obtain the following results:

3.5.1 Optimal Policy: RRKKRKRRRRS or 1R1R1K2K3R1K2R1R1R1R1S

Optimal Return: \$3,440

$t_1 = 2 \Rightarrow m + 1 - t_1 = 2 \Rightarrow S_1 = \{2\}, S_2 = \{1, 3\}, S_3 = \{1, 2\}, S_i = \{1, 2, 3\},$ for $i \in \{4, 5, \cdots, 10\}.$

Plug each S_i into the template under column I, to automatically obtain the following results:

3.5.2 Optimal Policy: RRKKRKRRRRS or 1R1R1K2K3R1K2R1R1R1R1S

Optimal Return: \$1,440

$t_1 = 3 \Rightarrow m + 1 - t_1 = 1 \Rightarrow S_1 = \{3\}, S_2 = \{1\}, S_3 = \{1, 2\}, S_i = \{1, 2, 3\},$ for $i \in \{4, 5, \cdots, 10\}.$

Plug each S_i into the template under column I, to automatically obtain the following results:

3.5.3 Optimal Policy: RRKKRKRRRRS or 1R1R1K2K3R1K2R1R1R1R1S

Optimal Return: $-$**\$560**

Note the stationarity of the optimal policy with respect to the machine ages 1 to 3 and the successive constant decrement of \$2,000 in the optimal returns.

Observe also that $s_1(t) < s(t_1)$, if $t > t_1 \geq m$, leading to the following conclusion:

$$f_1(t) = f_1(t_1) - \left[s_1(t_1) - s_1(t) \right], \text{for } t \geq t_1 \geq m.$$

Furthermore all states with starting ages $t_1 \geq m$ yield the same optimal policy. Using the fact that $m = 3$, and $s_1(3) = \$5,000$, the following conclusion is immediate for any machine with starting age $t_1 > 3$:

3.5.4 Optimal Policy: RRKKRKRRRRS or 1R1R1K2K3R1K2R1R1R1R1S

Optimal Return: $-\$560 - \$\left(5,000 - s_1(t_1)\right) = \$\left(s_1(t_1) - 5,560\right).$

Set $h_1(0) = -I_1 + f_1(0)$, and note the following relationship between the actual optimal return for ages 0 and m :

$$f_1(m) = s_1(m) - I_1 + f_1(0) \;\Leftrightarrow\; f_1(m) = s_1(m) + h_1(0)$$

The above relation holds because a salvage value of $s_1(m)$ is received for an m - year old machine at the beginning of decision period 1 and a new machine at purchase price I_1 is immediately deployed, whereas no such money is received for a new machine.

3.6 An Exposition on the Solution Template Code

Use the Excel cell references A1:I2 for documentation. Save the problem data in the indicated cells using the Copy and Paste functionality. Under the decision R, save the decision year - dependent fixed value $V_i(0) = r_i(0) - c_i(0) - I_i$ under the fixed cell reference \$F\$(4+ (m+8) i). Note that for each feasible i and fixed m, \$F\$(4+ (m+8) i) is the fixed row reference for the fixed F column. This value must be typed in without the parentheses. Similar remarks apply throughout the exposition.

3.6.1 Stage *n* Computations

For $t = 1$, under REPLACE, type the following code in the cell reference F10:

=If(\$I10 = “”,””,\$F\$4+ 0.001*(\$D\$10+D10)) <ENTER> to secure $f_n^R(1)$.

Click back on cell F10, position the cursor at the right edge of the cell until a crosshair appears. Then drag the crosshair vertically one down to the last required F(m+9) corresponding to $t = m$, to secure the remaining $f_n^R(t_n)$, where $t_n \in S_n$, as well as the blank spaces. Henceforth, the act of clicking back on a specified cell, positioning the cursor at the right edge of the cell until a crosshair appears and the crosshair-dragging routine will be referred to as clerical routine/duty.

For $t = 1$, under KEEP, type the following code in the cell reference E10:

=If(\$I10 =\$D\$2,”Must Replace”, if(\$I10= “”,””,0.001*(B10 +D11- C10))) <ENTER> to secure

$f_n^K(1)$. Perform the clerical duty to secure the remaining $f_n^K(t_n)$, where $t_n \in S_n$, as well as the blanks.

To secure $f_n(t)$, for $t \in S_n$, type the following code in the cell reference G10:

=If($I10 = " "," ",if (E10 = "Must Replace", F10, max(E10,F10)))<ENTER> to secure $f_n(1)$.

Then perform the clerical routine to secure $f_n(t_n), t_n \in S_n, t_n \neq 1$, as well as the blank spaces.

3.6.2 Remarks

In Excel, the max and min functions return values for only numeric expressions, ignoring string constants; for example if the number 5 is saved in B2 and the string constant "**Must**" is saved in C2, then in D2, the code: = max(B2, C2) <Enter> returns 5. In E2, the code: = max(B2,C2) <Enter> alsoreturns 5. Therefore the code segment involving "if (E10 = "Must Replace", F10"need not appearin the template.

To obtain the optimal decision for each of the stage n states $t = t_n \in S_n$, type the following code in the cell reference H10:

=If($I10 = " "," ",if(A10 = $D$2, "R", if(F10 > E10,"K", "K/R")))) <ENTER> to secure $D_n(1)$.

Then perform the clerical routine to secure $D_n(t_n), 1 \neq t_n \in S_n$, as well as the blanks.

3.6.3 Stage (*n* - 1) Computations

For $t = 1$, under REPLACE, type the following code in the cell reference F(10+(*m*+8)):
=If ($I(10 + (*m*+8)) = "","", F(4+ (*m*+8))+ 0.001*D(10+(*m*+8)) +$G$10) <ENTER>
to secure $f_{n-1}^R(1)$.

Perform the clerical duty to secure the remaining $f_{n-1}^R(t_{n-1})$, $t_{n-1} \in S_{n-1}$, as well as the blank spaces.

For $t = 1$, under KEEP, type the following code in the cell reference E(10 + (*m*+8)):
= If($I(10 + (*m*+8)) =$D$2,"Must Replace", if($I(10 + (*m*+8))=0, "","", 0.001*(B(10 + (*m*+8))-C(10 + (*m*+8))+1000*$G$11)) <ENTER>to secure $f_{n-1}^K(1)$.

Perform the clerical duty to secure the remaining $f_{n-1}^K(t_{n-1})$, $t_{n-1} \in S_{n-1}$, as well as the blank spaces.

To secure $f_{n-1}(t_{n-1})$, for $1 \neq t_{n-1} \in S_{n-1}$, type the following code in the cell reference G(10 + (*m*+8)):

=If($I(10 + (*m*+8)) = " "," ",max(E(10 + (*m*+8)),F(10 + (*m*+8)))) <ENTER> to secure $f_{n-1}(1)$.

Then perform the clerical routine to secure the remaining $f_{n-1}(t_{n-1})$, $t_{n-1} \in S_{n-1}$, as well the blank spaces.

To obtain the optimal decision for each of the stage $n-1$ states $t = t_{n-1} \in S_{n-1}$, type the following code in the cell reference H(10 + (*m*+8)):
=If($I(10 + (*m*+8)) = " "," ",if(A(10 + (*m*+8)) = $D$2, "R", if(F(10 + (*m*+8))> E(10 + (*m*+8)),"K", "K/R")))) <ENTER> to secure $D_{n-1}(1)$.

Then perform the clerical routine to secure $D_{n-1}(t_{n-1})$ and the blanks.

3.6.4 Stage $(n-j)$ Computations, $j \in \{2, \cdots, n-1\}$.

For $t = 1$, under REPLACE, type the following code in the cell reference F(10 + (*m*+8)*j*)):
=If($I(10 + (*m*+8)*j*)) = "","", F(4+ (*m*+8)*j*)+ 0.001*D(10 + (*m*+8) *j*))
+G(10 + (*m*+8)($j-1$))<ENTER> to secure $f_{n-j}^R(1)$.

Perform the clerical duty for the remaining *m* cells to secure $f_{n-j}^R(t_{n-j})$ and the blank spaces

For $t = 1$, under KEEP, type the following code in the cell reference E(10 + (*m*+8) *j*)):
=If ($I (10 + (*m*+8) *j*)) =$D$2,"Must Replace", if($I(10 + (*m*+8) *j*))= "","",
0.001*(B (10 + (*m*+8) *j*) − C(10 + (*m*+8) *j*)+1000*G(11 + (*m*+8)($j-1$)))<ENTER> to secure $f_{n-j}^K(1)$. Perform the clerical duty to secure other $f_{n-2}^K(t_{n-j})$ and the blank spaces.

To secure $f_2(t)$, for $t \in S_2 = \{1\}$, type the following code in the cell reference G(10 + (*m*+8) *j*)):

=If($I(10 + (*m*+8) *j*)) = " "," ",max(E (10 + (*m*+8) *j*)),F(10 + (*m*+8) *j*)))) <ENTER> to secure $f_{n-j}(1)$.

Then perform the clerical routine to secure other $f_{n-j}(t_{n-j})$ and the blank spaces.

To obtain the optimal decision type the following code in the cell reference H(10 + (m+8) j)):
=If($I(10 + ($m$+8) j)) = " "," ",if(A(10 + (m+8) j)) = $D$2, "R",
if(F(10 + (m+8) j))> E(10 + (m+8) j)),"K", "K/R"))))<ENTER> to secure $D_{n-j}(1)$.

Then perform the clerical routine to secure the remaining $D_{n-j}(t_{n-j})$ and the blanks.

For stage 1, $f_1^K(0), f_1^R(0), f_1(0)$ and $D_1(0)$ must be secured for a new machine.Since the assigned value for m depends on the given problem, it makes sense to secure blank values for the infeasible $f_1^K(0), f_1^R(0), f_1(0)$ and $D_1(0)$, in stages 2 to n.

3.7 Observations on the Template, and its Implementation Scope

(1) For $i \in \{n-1\cdots,2,1\}$, the code for the determination of $\left(f_i(t), D_i(t)\right)$ and $\left(f_{i-1}(t), D_{i-1}(t)\right)$ are Excel equivalent in the sense that they are the same, except for a constant row reference difference of $(m+8)$.

The results for the pair $\left(f_i(t), D_i(t)\right)$ are automatic in stages $i \in \{n-2,\cdots,1\}$ following the implementation of the pair $\left(f_{n-1}(t), D_{n-1}(t)\right)$ in stage $n-1$ and the required copy and paste operations. It must be stressed that the Option "Keep Source Formatting" should be used in paste operations.

(2) In particular, the template automatically solves any Equipment Replacement problem of the same structure, with the same data set, but different starting ages $t_1 \in \{1,2,\cdots,m\}$. All that is required is simply to type in the appropriate S_i values in column I, as in the above examples.

(3) The template can solve any Equipment Replacement problem of the same structure, with a different data set and different starting ages $t_1 \le m$, the mandatory replacement age, and the planning horizon length n. In this case the age range in column A should be adjusted to $t \in \{0,1,\cdots,m\}$ and the appropriate S_i values assigned in column I, as in the above examples. The code for replacement should assign blank for all infeasible (blank) ages under column I.

3.8 Closing Remarks

Suppose that the following constraint is imposed: (Minimum replacement age $= p$ years).Then theorem 1 modifies to

$$f_i(t) = \begin{cases} r_i(t) - c_i(t) + f_{i+1}(t+1), & 0 \le t \le p-1 \\ \max \begin{Bmatrix} r_i(t) - c_i(t) + f_{i+1}(t+1); & \text{KEEP} \\ r_i(0) + s_i(t) - I_i - c_i(0) + f_{i+1}(1); & \text{REPLACE} \end{Bmatrix}, & p \le t \le m \end{cases}$$

$f_{n+1}(x) = s_i(x),\ i = 0,1,\ldots,n-1,\ x =$ age of machine at the start of period $n+1$

In the above case the code for the template need only make the following modifications for $t = t_i \in S_i$, for each $i \in \{1,2,\cdots,n\}$ such $0 \le t_i \le p-1$, under the decision REPLACE: Save the value $p-1$ in the fixed cell reference $I$2.

For $t = 1$, under REPLACE, type the following code in the cell reference F10:
= If (AND ($I10 >=0, $I10 <= $I$2),"Must Keep",If ($I10 = "","",$F$4+ 0.001*($D$10+D10)))
<ENTER> to secure $f_n^R(1)$. Incorporate the code segment **If (AND ($I10 >=0, $I10 <= $I$2),"Must Keep",** and the closing parenthesis **)** in each code segment under decision **REPLACE**

For $t = 1$, under KEEP, type the following code in the cell reference E10:
=If($I10 =$D$2,"Must Replace", if($I10= "","",0.001*(B10 +D11- C10))) <ENTER> to secure

4.0 Conclusion

This research article extended solution templates for optimal solutions to Machine Replacement problems (MRP) in [2], to a larger class of MRP problem, together with an exposition on the interface and solution process. The state inputs to the template exploited the already developed computational formulas for the states corresponding to each decision year by [2]. By restructuring the age and decision year variable data in three – dimensional formats and appropriately modifying the dynamic programming recursions and deploying these in the implementation template, optimal trade – in policies, as well as the corresponding optimal returns are automatically generated as soon as the relevant inputs are typed into appropriate Excel cells.
Finally the article solved four illustrative examples of the same flavour that demonstrated the efficiency, power and utility of the solution template prototype. In general, the template could be deployed to solve each equipment replacement problem in less than 10 percent of the time required for the manual generation of the alternate optima.

5.0 References

[1] Taha, H.A. Operations Research: An Introduction. Seventh Edition. Prentice-Hall of India, New Delhi. (2006). pp. 418-421.

[2] Ukwu, C. Novel State Results on Equipment Replacement Problems and Excel Solution Implementation Templates. New Journal of Nigerian Association of Mathematical Physics, International Edition. Vol. 1, November, 2015.?