



---

## OPTIMAL FUND ALLOCATION FROM CERTAIN INVESTMENT PORTFOLIO USING BACKWARD DYNAMIC PROGRAMMING RECURSIONS WITH ELECTRONIC IMPLEMENTATIONS

<sup>1</sup>Ukwu, Chukwunye; <sup>2</sup>Manjel, Danladi & <sup>3</sup>Kutchin, Stephen

1,2,3Department of Mathematics, University of Jos, P.M.B 2084, Jos, Plateau State, Nigeria.

### ABSTRACT

*This research article investigated the problem of fund allocations from certain investment portfolio and obtained the optimal investment strategies using backward dynamic programming recursive approach. In the sequel, the article conceptualized, formulated and designed Excel solution templates and corresponding algorithm for the optimal allocation of funds from the investment portfolio. The work also provided illustrative examples which demonstrate the efficiency, utility and processing power of the solutions templates. The deployment of the solution templates circumvents the inherent tedious, and prohibitive manual computations associated with dynamic programming formulations and recursions and may be optimally appropriated for sensitivity analyses on such models.*

**Keywords:** Dynamic programming, Optimal, Algorithm, Solution templates, Investment portfolio, Fund Allocation.

### 1. INTRODUCTION

One of the challenges faced by investor's today in search of comfortable ways to earn higher returns on their investments above the current certificate of deposit (CD) and interest rates is the efficient allocation of funds, which is one of the most important functions of the financial management in modern times, Gupta and Hira [1].

The optimal investment policy has been investigated for a portfolio of two banks in a certain class of investment problems using deterministic dynamic programming recursions Taha [2]. Unfortunately the related issue of computational feasibility is yet to be addressed. Dynamic programming iterations are computationally intractable and doomed to failure for practical purposes, especially for large scale applications. The need for electronic implementation of optimal investment policies is imperative. The main contributions of this work will be as follows: it will extend the optimal investment policy to an arbitrary number of banks. It will also automate the computations of the optimal investment policies, thereby circumventing the prohibitive manual computations associated with dynamic programming approach and paving the way for sensitivity analyses on the model, the latter of which could hardly be contemplated in manual computations.

The main purpose of this finding is to obtain the optimal policy prescriptions for fund allocation from certain investment problems involving an arbitrary number of banks. This study will also formulate and design corresponding Excel automated solution templates using deterministic dynamic programming recursions. The solution templates will be motivated by the earlier works of Ukwu [3,4,5,6] and the way these were deployed to eliminate computational tedium and other constraints, as well as pave the way for sensitivity analyses with guarantee for instant generation of optimal investment strategies, as soon as the pertinent data were keyed in. These activities could hardly be contemplated in manual computations.

The template outputs of this paper will reflect and demonstrate consistency with the base results. The findings of this study will be of great benefits to the financial sector of the economy especially to investors and bank managers. In particular, the deployment of the solution templates will circumvent the computational drudgery of the key indicators as well as facilitate the sensitivity analyses on the model parameters thereby addressing the issue of computational feasibility and timely recommendation of optional investment strategies.

## **2. THEORETICAL ANALYSIS**

### **2.1 Compound Interest Computations**

The Total Accumulated Value (TAV), is the principal sum plus compounded interest and is given by the formula:

$$\text{TAV} = P \left( 1 + \frac{i}{n} \right)^{nt},$$

where  $P$  is the principal sum;  $i$  is the nominal interest rate;  $n$  is the annual compounding frequency;  $t$  is the planning horizon length the interest is applied.

Using the backward dynamic recursions

$$f_i(x_i) = \max_{0 \leq I_{i,j} \leq x_i} \{s_i + f_{i+1}(x_{i+1})\}, i = 1, 2, \dots, n-1; f_{n+1}(x_{n+1}) = 0$$

the following problem of obtaining the optimal investment policies for a given portfolio of banks with prescribed annual interest rates and end-of-year bonuses was solved in Ukwu et al [7]:

## 2.2 Problem Statement

$$\text{Maximize } z = \sum_{i=1}^n s_i$$

$$\text{s.t. } s_i = \sum_{j=1}^k I_{i,j} \alpha_{i,j}^{n+1-i}, i \in \{1, 2, \dots, n-1\}$$

$$s_n = \sum_{j=1}^k I_{n,j} \alpha_{n,j} + \sum_{j=1}^k q_{n,j} I_{n,j} = \sum_{j=1}^k I_{n,j} [\alpha_{n,j} + q_{n,j}],$$

where

$k$  : Number of banks to be invested in

$P_i$  : Fixed component of the amount for investment in all  $k$  banks at the start of year  $i$ ,  
 $i \in \{1, 2, \dots, n\}$

$r_{i,j}$  : Nominal annual interest rate offered by bank  $j$  in year  $i$ ,  $j \in \{1, 2, \dots, k\}$

$q_{i,1}, q_{i,2}, \dots, q_{i,k}$  : Bonus percentages paid at the end of year  $i$  in which investment is made in all  $k$  banks.

$x_i$  : Variable amount available for investment in all  $k$  banks in year  $i$  (includes  $P_i$ )

$I_{i,j}$  : Actual amount invested in bank  $j$  at the beginning of year  $i$

$s_i$  : Accumulated sum at the end of year  $i$  given that  $I_{i,j}$  is the investment made in bank  $j$  at the beginning of year  $i$ ,  $j \in \{1, 2, \dots, k\}; i \in \{1, 2, \dots, n\}$

$f_i(x_i)$  = optimal value of investments for years  $i, i+1, \dots$ , and  $n$  given  $x_i$

$$s_i = f_i(x_i) - f_{i+1}(x_{i+1}), i \in \{1, 2, \dots, n\}.$$

The electronic computational process for the optimal investment strategies and corresponding returns are detailed as follows:

### 3. RESULTS AND ANALYSIS

#### 3.1 Excel Solution Templates and Electronic Computations for the Optimal Policy Prescriptions and Returns

##### 3.1.1 An Algorithm for the Implementation of Theorem on Available and Accumulated Funds

$$f_i(x_i) = (\text{max coeff. of } x_i \text{ in stage } i) x_i + (1 - \delta_{i_n}) \sum_{\lambda=\min\{i+1, n\}}^n P_\lambda (\text{max coeff of } x_\lambda \text{ in } f_\lambda(x_\lambda)) I_{ij^*} x_i + (1 - \delta_{i_n}) c_i$$

##### Step 1: Design of Excel Solution Template

**Table I: Excel Spreadsheet Layout, Documentation, Data and Fixed Value Storage, Stage Numbering, Policy Prescriptions and Reward Automation.**

	A	B	C	D	E	...	L	M	N
1			$n =$	$n\_val$		$n$			
2									
3			<b>Given Data</b>						
4									
5	$P_n =$	Val	Stage	$=[1]$					
6									
7	$j \rightarrow$		1	2	3	4	5	...	10

8	$r_{n,j} \rightarrow$		$r_{n,1}$	$r_{n,2}$	$r_{n,3}$	$r_{n,4}$	$r_{n,5}$	...	$r_{n,12}$
9	$\alpha_j \rightarrow$		= [4] $\rightarrow$						
10	$P_n; q_{n,j} =$	$P_n$	$q_{n,1}$						
11	$x_{n+1}$ component coeff.	= [6]							
12	$x_n$ , coeff. of $I_{n,j} = \alpha_j + q_{n,j}$		= [8]						
13	Max coeff. of $x_n$ in $f_n(x_n)$		= [9]						
14	$j^*$		= [10]						
15	$c_n$							= [13]	= [12]
16	$x_n \rightarrow$		= [11]						
17									
18	$P_{n-1}$		Stage =	[2]					
19									
20	$j \rightarrow$								
21	$r_j \rightarrow$								
22	$\alpha_j \rightarrow$		= [5]						
23	$x_n$ comp. coeff. in stage $(n-1)$ : $P_n; q_{n-1,j}$								
24	(max coeff. of $x_n$ in $f_n(x_n)$ ) * ( $x_n$ comp. coeff. in stage $(n-1)$ )	= [7]							
25	Return func. coeff (RFC): $\alpha_j$ + preced. contiguous row	= [14]							
26	Max coeff. of $x_{n-1}$ in $f_{n-1}(x_{n-1})$	= [15]							
27	$j^*$	= [16]							
28	$c_{n-1}$	= [17]						= [20]	= [19]
29	$x_{n-1} \rightarrow$	= [18]							
30									
31	$P_{n-2}$		Stage =	[3]					

32									
33	$j \rightarrow$								
34	$r_j \rightarrow$								
35	$\alpha_j \rightarrow$								
36	$x_{n-1}$ comp.coeff.in stage $(n-2)$ : $P_{n-1}; q_{n-2,j}$								
37	$(\max \text{coeff. of } x_{n-1} \text{ in } f_{n-1}(x_{n-1}))$ $\ast (x_{n-1} \text{ comp.coeff.in stage } (n-2))$								
38	Return func.coeff(RFC): $\alpha_j + \text{pred.contiguous row}$								
39	Max coeff. of $x_{n-2}$ in $f_{n-2}(x_{n-2})$								
40	$j^*$								
41	$c_i$								
42	$x_i \rightarrow$								
43									
44	$P_{n-3}$								
45									
46	$j \rightarrow$								
47	$r_j \rightarrow$								
48	$\alpha_j \rightarrow$								
49	$x_{n-2}$ comp.coeff.in stage $(n-3)$ : $P_{n-2}; q_{n-3,j}$								
50	$(\max \text{coeff. of } x_{n-2} \text{ in } f_{n-2}(x_{n-2}))$ $\ast (x_{n-2} \text{ comp.coeff.in stage } (n-3))$								
51	Return func.coeff(RFC): $\alpha_j + \text{pred.contiguous row}$								
52	$x_{n-3}$ in $f_{n-3}(x_{n-3})$								
53	$j^*$								
54	$c_i$								
55	$x_i \rightarrow$								

Note that  $i \in \{n, n-1, \dots, 1\} \Rightarrow \sum_{\lambda=\min\{i+1, n\}}^n (.) = \sum_{\lambda=i+1}^n (.)$ , since  $\sum_{\lambda=n+1}^n (.) = 0$ , and  $\min\{i+1, n\} = i+1$ ,

for  $i \in \{n-1, n-2, \dots, 1\}$ .

Use Excel column A and other indicated cell references for identifiers and documentation, as shown above. Save the nominal interest rate data in stages  $n, n-1$ , and  $n-2$ , in Excel rows  $8 + 13(n-i), i \in \{n, n-1, n-2\}$ , in contiguous cell locations, beginning from column B; save the maintenance cost data in Excel row 7, in contiguous cell locations, beginning from column C; save the fresh money and bonus data in Excel row 10, in contiguous cell locations, beginning from column B; save the identifiers in the remaining stages  $n-1$  to 1 in the above table, using the Copy and Paste functionality. Consecutive stages should be separated by a blank row.

Store the planning horizon length  $n$  in the fixed (absolute) cell references \$D\$1.

To automate the stage numbering, perform the following actions:

= [1]: Store the last stage number  $n$  under the relative cell reference \$D5.

= [2]: Secure the stage number  $n-1$  under the relative cell reference \$D18.

= [3]: Secure the stage number  $n-2$  under the relative cell reference \$D31.

## Step 2: Automation of other computations in stages $n, n-1$ , and $n-2$ .

From this step onward, horizontal crosshair dragging is associated with each code indicated by

= [integer h]: , starting from = [ 4h]:

**(a) Automation of  $\alpha_{ij} = \prod_{\lambda=i}^n (1 + r_{\lambda,j})$**

= [4]: Type the code for  $\alpha_{n1}$ , in C9, followed by the keyboard code execution <Enter>.

Click back on cell C9 and position the cursor at the right edge of the cell until a crosshair appears. Then drag the crosshair across to the last the cell N9 to secure  $\alpha_{n,j}, j \in \{2, \dots, k\}$  with trailing 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.

= [5]: Store the code for  $\alpha_{n-1,1}$ , in C22.

Perform the horizontal clerical duty across to the last cell location N22, to secure  $\alpha_{n-1,j}, j \in \{2, \dots, k\}$

and trailing blanks. Now copy C22:L22 and paste it successively onto the cell references

$C[5+13(n-i)]:L[5+13(n-i)]$ , for  $i \in \{n-2, \dots, 2, 1\}$ , to secure  $\alpha_{ij} = \prod_{\lambda=i}^n (1+r_{\lambda,j}), i \in \{n-2, \dots, 1\}$

**(b)Automation of  $x_{n+1}$  components**

= [6]: Store the code for the first  $x_{n+1}$  component, in B11. Then perform the horizontal clerical duty across to the last cell location L11 to secure the remaining components and trailing blanks.

= [7]: Store the code to secure the first value of  $(\max \text{coeff. of } x_n \text{ in } f_n(x_n)) * (x_n \text{ comp. coeff. in stage } (n-1))$ , in B24.

Then perform the horizontal clerical duty across to the last cell location L24 to secure the remaining values and trailing blanks. Copy the operation in [7] to B37:L37 in stage  $(n-2)$ .

**(c) Automation of stage  $n$  outputs:  $\alpha_{n,j} + q_{n,j}$ , Max coeff of  $x_n$  in  $f_n(x_n)$ ,  $j^*$ ,  $x_n$ ,  $f_n(x_n)$ ,  $s_n$  :**

= [8]: Store the code for  $\alpha_{n,1} + q_{n,1}$ , in C12; then perform the horizontal clerical duty across to the last cell location L12, to obtain  $\alpha_{n,j} + q_{n,j}, j \in \{2, \dots, k\}$  and trailing blanks.

= [9] Store the code for the maximum coefficient of  $x_n$  in  $f_n(x_n)$ , in [C13:L13].

= [10] Store the code for the arg max,  $j^*$  of  $f_n(x_n)$ , in [C14: L14].

= [11]: Store the code for the available fund,  $x_n$ , for investment at the beginning of year  $n$ , in [C16:L16].

= [12] Store the code for  $f_n(x_n)$ , in N15.

= [13] Store the code for  $s_n$ , in M15.

**(d) Automation of Return Function Coefficients in stage  $(n-1)$  :**

$\alpha_{n-1,j} + (\max \text{coeff. of } x_n \text{ in } f_n(x_n)) * (x_n \text{ comp. coeff. in stage } (n-1))$ .

= [14]: Store the code for



$$\alpha_{n-1,0} + (\max \text{coeff. of } x_n \text{ in } f_n(x_n)) * (x_n \text{ comp. coeff. in stage } (n-1)); \alpha_{n-1,0} = 0,$$

in B25; then perform the horizontal clerical duty across to the last cell location L25 to obtain

$$\alpha_{n-1,j} + (\max \text{coeff. of } x_n \text{ in } f_n(x_n)) * (x_n \text{ comp. coeff. in stage } (n-1)); j \in \{1, \dots, k\} \text{ and trailing blanks.}$$

**Automation of Max coeff. of  $x_{n-1}$  in  $f_{n-1}(x_{n-1})$**

=**[15]**:Store the code for Max coeff. of  $x_{n-1}$  in  $f_{n-1}(x_{n-1})$ , in[C26:L26].

**Automation of arg max,  $j^*$  in stage  $n-1$**

=**[16]**:Store the code for  $j^*$ , in [C27:L27].

**Automation of  $c_{n-1}$**

=**[17]**:Store the code for  $c_{n-1}$ , in B28.

**Automation of  $x_{n-1}$**

=**[18]**:Store the code for  $x_{n-1}$ , in [C29:L29].

**Automation of  $f_{n-1}(x_{n-1})$**

=**[19]**:Store the code for  $f_{n-1}(x_{n-1})$ , in N28.

**Automation of  $s_{n-1}$**

=**[20]**:Store the code for  $s_{n-1}$ , in M28

**Step 3: Automation of other computations in stages  $n-2, n-3, \dots, 1$**

(a) Copy the contiguous region \$A18:N29 of stage  $n-1$  into the contiguous region \$A31:N42 of stage  $n-2$ ; Modify the code in D31 to ‘=\$D18-1’ to secure stage  $(n-2)$  computational values

**(b) Stage  $i$  Implementations,  $i \in \{n-3, \dots, 2, 1\}$ , in One Fell Swoop**

This is a crucial step involving a single Copy and  $n-3$  Paste Operations, using the contiguous region

\$A31:N42 of stage  $(n-2)$ . Simply use the Copy and Paste functionality to copy and paste the contiguous region \$A31:L42 successively into stages  $(n-3)$  to 1 regions.

**Note:** Consecutive stages should be separated by a blank row. In other words, for  $i \in \{n-3, n-4, \dots, 1\}$  the Copy and Paste functionality can be used to copy and paste the contiguous region \$A31:L42 successively into stages  $(n-3)$  to 1 regions:

$$A\$[5+13(n-i)]:N\$[16+13(n-i)].$$

### 3.2 Remarks on the Use of the Templates for Large Problem Sizes

It is clear that the crosshair horizontal-dragging routine must be extended beyond column N, as appropriate, if  $k \geq 11$ . This can be optimally done before the Copy and Paste operations from stage  $n-1$ . Hence the template can be adequately appropriated for sensitivity analyses on this class of Investment problems in just a matter of minutes, as contrasted with manual investigations that would at best consume hours or days with increasing values of  $k$  and/or  $n$ , not to talk of the dire consequences of committing just one error in any stage computations.

The template can now be deployed to obtain the optimal investment strategies and corresponding returns, for a portfolio of nine banks, with a horizon length of eight years:

### 3.3 Problem 1: 9-Bank- 8 - Year Horizon-Length Problem

Given the amount \$5, 000,000 to be invested now and \$3, 000,000, \$7, 000,000, \$5, 000,000, \$2,000,000, \$10, 000,000, \$8, 000,000 and \$13, 000,000 at the start of years 2, 3, 4, 5, 6, 7, and 8. The annual interest rates offered by bank 1 for years 1, 2, 3, 4, 5, 6, 7, and 8 are given by 1.3%, 0.8%, 2.1%, 0.7%, 1.65%, 2.11%, 0.75% and 1.27% compounded annually, and the bonuses over the next 8 years are 0.25%, 0.9%, 1.66%, 1.7%, 1.55%, 0.74%, 2.2% and 1.48% respectively. The annual interest rates offered by bank 2 are 0.4%, 0.1%, 0.3%, 1.7%, 1.22%, 0.99%, 1.32%, and 1.47% and the bonuses over the next 8 years are 1.14%, 1.2%, 1.3%, 1.25%, 1.65%, 1.29%, 2%, 1.72% respectively. The annual interest rates offered by bank 3 are lower by 0.2% than that of bank 1 but its bonuses are higher by 0.4%.

The annual interest rates offered by bank 4 are 2.6%, 2.63%, 1.72%, 1.37%, 1.8%, 2.5%, 1.48%, 1.56% and the bonuses are 0.7%, 1.52%, 1.7%, 2.6%, 0.9%, 0.7%, 0.83% and 1.58% respectively.

The annual interest rates offered by bank 5 are higher by 0.3% than that of bank 2, but its bonuses are lower by 0.25%.

The annual interest rates offered by bank 6 are 1.27%, 2%, 1%, 1.5%, 1.23%, 1.72%, 0.6%, 1.2% and its bonuses are 2.1%, 1.3%, 0.2%, 0.1%, 0.72%, 0.75%, 1.7% and 2.6%.

The annual interest rates offered by bank 7 are lower by 0.5% than that of bank 4 but its bonuses are higher by 0.08%.

The annual interest rates offered by bank 8 are 2.16%, 2.1%, 2.6%, 1.92%, 1.11%, 1.3%, 1.52%, 1.6% and the bonuses are 1.68%, 1.8%, 1.62%, 1.23%, 1.1%, 1.73%, 0.7%, and 2.4% respectively.

The annual interest rates offered by bank 9 are lower by 0.3% than that of bank 6 but its bonuses are higher by 0.27%. The objective is to maximize the accumulated capital at the end of 8 years.

**Table 2: Pertinent Data for Problem 3.3, with Legend**

		Bank									Bank								
		1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9
Year	P	Nominal Annual Interest Rates									End-of-Year Bonuses								
1	5000000	0.01300	0.00400	0.01297	0.02600	0.00401	0.01270	0.02587	0.02160	0.01266	0.00250	0.01140	0.00251	0.00700	0.01137	0.02100	0.00701	0.01680	0.02106
2	3000000	0.00800	0.00100	0.00798	0.02630	0.00100	0.02000	0.02617	0.02100	0.01994	0.00900	0.01200	0.00904	0.01520	0.01197	0.01300	0.01521	0.01800	0.01304
3	7000000	0.02100	0.00300	0.02096	0.01720	0.00301	0.01000	0.01711	0.02600	0.00997	0.01660	0.01300	0.01667	0.01700	0.01297	0.00200	0.01701	0.01620	0.00201
4	5000000	0.00700	0.01700	0.00699	0.01370	0.01705	0.01500	0.01363	0.01920	0.01496	0.01700	0.01250	0.01707	0.02600	0.01247	0.00100	0.02602	0.01230	0.00100
5	2000000	0.01650	0.01220	0.01647	0.01800	0.01224	0.01230	0.01791	0.01110	0.01226	0.01550	0.01650	0.01556	0.02630	0.01646	0.00720	0.02632	0.01110	0.00722
6	10000000	0.02110	0.00990	0.02106	0.02500	0.00993	0.01720	0.02488	0.01300	0.01715	0.00740	0.01290	0.00743	0.01720	0.01287	0.00750	0.01721	0.01730	0.00752
7	8000000	0.00750	0.01320	0.00749	0.01480	0.01324	0.00600	0.01473	0.01520	0.00598	0.02200	0.02000	0.02209	0.00600	0.01995	0.01700	0.00600	0.00700	0.01705
8	13000000	0.01270	0.01470	0.01267	0.01560	0.01474	0.01200	0.01552	0.01600	0.01196	0.01480	0.01720	0.01486	0.01200	0.01716	0.02600	0.01201	0.02400	0.02607
		Interest Relations				$i \in \{1, 2, \dots, 8\}$				Bonus Relations									
		$r_{i3}$	=	0.9980	$r_{i1}$						$q_{i3}$	=	1.0040	$q_{i1}$					
		$r_{i5}$	=	1.0030	$r_{i2}$						$q_{i5}$	=	0.9975	$q_{i2}$					
		$r_{i7}$	=	0.9950	$r_{i4}$						$q_{i7}$	=	1.0008	$q_{i4}$					
		$r_{i9}$	=	0.9970	$r_{i6}$						$q_{i9}$	=	1.0027	$q_{i6}$					

**Table 3: Template Outputs for Optimal Policies and Returns: Stages 8 to 5**

	$n =$	8	$k =$	9							
$f_i(x_i) = (\max \text{coeff. of } x_i \text{ in stage } i)x_i$ $+ (1 - \delta_{i,n}) \sum_{\lambda=i+1}^n P_{\lambda} (\max \text{coeff of } x_{\lambda} \text{ in } f_{\lambda}(x_{\lambda}) = f_{i,n}x_i + (1 - \delta_{i,n})c_i)$											
$p_n =$	13000000	Stage	8							$s_i$	$f_i(x_i) \downarrow$
$j \rightarrow$		1	2	3	4	5	6	7	8	9	
$r_j \rightarrow$		0.01270	0.01470	0.01267	0.01560	0.01474	0.01200	0.01552	0.01600	0.01196	
$\alpha_j \rightarrow$		1.0127	1.0147	1.0126746	1.0156	1.014744	1.012	1.01552	1.016	1.01196	
$p_n, q_{n,j} =$	13000000	0.01480	0.01720	0.01486	0.01200	0.01716	0.02600	0.01201	0.02400	0.02607	
$x_{n+1}$ component coeff.	0	0	0	0	0	0	0	0	0	0	
$x_n$ Coeff of $l_{nj} = a_j + q_{nj}$		1.0275	1.0319	1.0275338	1.0276	1.031901	1.038	1.02753	1.04	1.03803	
Max coeff of $x_n$ in $f_n(x_n)$									1.04		
$j^*$									8		
$c_i$										13689572.68	13689572.68
$x_i \rightarrow$						13163051					
$p_{n-1} =$	8000000	Stage	7	Computations							
$j \rightarrow$		1	2	3	4	5	6	7	8	9	
$r_j \rightarrow$		0.00750	0.01320	0.00749	0.01480	0.01324	0.00600	0.01473	0.01520	0.00598	
$\alpha_j \rightarrow$		1.0203	1.028094	1.020254469	1.030631	1.028179	1.01807	1.03048	1.03144	1.01802	
$x_n$ comp. coeff. in stage $(n-1): p_n, q_{n-1,j}$	8000000	0.02200	0.02000	0.02209	0.00600	0.01995	0.01700	0.00600	0.00700	0.01705	
Max coeff. of $x_n$ in $f_n(x_n)$ (x comp. coeff. in stage $n-1$ )	8320000	0.02288	0.0208	0.02297152	0.00624	0.020748	0.01768	0.00624	0.00728	0.01773	
Return function coeff (RFC): $\alpha_j$ + preced. contiguous row	8320000	1.04318	1.048894	1.043225989	1.036871	1.048927	1.03575	1.03672	1.03872	1.03575	
Max coeff of $x_{n-1}$ in $f_{n-1}(x_{n-1})$						1.048927					
$j^*$						5					
$c_i$	13520000									8403270.15	22092842.82
$x_i \rightarrow$					8172965						
$p_{n-2} =$	10000000	Stage	6	Computations							
$j \rightarrow$		1	2	3	4	5	6	7	8	9	
$r_j \rightarrow$		0.02110	0.00990	0.02106	0.02500	0.00993	0.01720	0.02488	0.01300	0.01715	
$\alpha_j \rightarrow$		1.04182	1.038272	1.041738784	1.056397	1.038388	1.03558	1.05611	1.04485	1.03547	
$x_{n-1}$ comp. coeff. in stage $(n-2): P_{n-1}, q_{n-2,j}$	10000000	0.00740	0.01290	0.00743	0.01720	0.01287	0.00750	0.01721	0.01730	0.00752	
Max coeff. of $x_{n-1}$ in $f_{n-1}(x_{n-1})$ (x comp. coeff. in stage $n-2$ )	10489269	0.00776	0.013531	0.007793107	0.018042	0.013497	0.00787	0.01806	0.01815	0.00789	
Return function coeff (RFC): $\alpha_j$ + preced contiguous row	10489269	1.04959	1.051803	1.049531891	1.074438	1.051886	1.04345	1.07417	1.063	1.04336	
Max coeff of $x_{n-2}$ in $f_{n-2}(x_{n-2})$					1.074438						
$j^*$					4						
$c_i$	21911415									10623231.3	32716074.15
$x_i \rightarrow$					10056101						
$P_{n-3}$	2000000	Stage	5	Computations							
$j \rightarrow$		1	2	3	4	5	6	7	8	9	
$r_j \rightarrow$		0.01650	0.01220	0.01647	0.01800	0.01224	0.01230	0.01791	0.01110	0.01226	
$\alpha_j \rightarrow$		1.05901	1.050939	1.058893097	1.075412	1.051095	1.04832	1.07502	1.05645	1.04817	
$P_{n-3}, q_{n-3,j}$	2000000	0.01550	0.01650	0.01556	0.02630	0.01646	0.00720	0.02632	0.01110	0.00722	
$x_2$ component coeff.	2148876	0.01665	0.017728	0.016720407	0.028258	0.017684	0.00774	0.02828	0.01193	0.00776	
Coeff of $l_{ij} = a_j + q_{ij} = \text{RFC}$	2148876	1.07567	1.068667	1.075613504	1.10367	1.068779	1.05606	1.1033	1.06838	1.05593	
Max coeff of $x_{n-3}$ in $f_{n-3}(x_{n-3})$					1.10367						
$j^*$					4						
$c_i$	32655797									2293976.37	35010050.52
$x_i \rightarrow$					2133114						

**Table 4: Template Outputs for Optimal Policies and Returns: Stages 4 to 1**

$P_{n-4}$	5000000	Stage	4	Computations							$s_i$	$f_i(x_i) \downarrow$
$j \rightarrow$		1	2	3	4	5	6	7	8	9		
$r_j \rightarrow$		0.00700	0.01700	0.00699	0.01370	0.01705	0.01500	0.01363	0.01920	0.01496		
$\alpha_i \rightarrow$		1.06643	1.068805	1.066290524	1.090145	1.069017	1.06405	1.08968	1.07673	1.06385		
$P_{n-4}, q_{n-4,j}$	5000000	0.01700	0.01250	0.01707	0.02600	0.01247	0.00100	0.02602	0.01230	0.00100		
$x_2$ component coeff.	5518348	0.01876	0.013796	0.018837431	0.028695	0.013761	0.0011	0.02872	0.01358	0.00111		
Coeff of $I_{ij} = a_i + b_{ij} = RPC$	5518348	1.08519	1.082601	1.085127955	1.11884	1.082778	1.06515	1.1184	1.09031	1.06496		
Max coeff of $x_{n-4}$ in $f_{n-4}(x_{n-4})$					1.11884							
$j^*$					4							
$c_i$	34863136										5581306.85	40591357.38
$x_i \rightarrow$					5119784							
$P_{n-5}$	7000000	Stage	3	Computations								
$j \rightarrow$		1	2	3	4	5	6	7	8	9		
$r_j \rightarrow$		0.02100	0.00300	0.02096	0.01720	0.00301	0.01000	0.01711	0.02600	0.00997		
$\alpha_i \rightarrow$		1.08882	1.072011	1.08863784	1.108895	1.072234	1.07469	1.10833	1.10473	1.07446		
$P_{n-5}, q_{n-5,j}$	7000000	0.01660	0.01300	0.01667	0.01700	0.01297	0.00200	0.01701	0.01620	0.00201		
$x_2$ component coeff.	7831882	0.01857	0.014545	0.018647041	0.01902	0.014509	0.00224	0.01904	0.01813	0.00224		
Coeff of $I_{ij} = a_i + b_{ij} = RPC$	7831882	1.10739	1.086556	1.107284881	1.127916	1.086742	1.07692	1.12736	1.12285	1.0767		
Max coeff of $x_{n-5}$ in $f_{n-5}(x_n)$					1.127916							
$j^*$					4							
$c_i$	40457338										7813423.55	48404780.92
$x_i \rightarrow$					7046132							
$P_{n-6}$	3000000	Stage	2	Computations								
$j \rightarrow$		1	2	3	4	5	6	7	8	9		
$r_j \rightarrow$		0.00800	0.00100	0.00798	0.02630	0.00100	0.02000	0.02617	0.02100	0.01994		
$\alpha_i \rightarrow$		1.09753	1.073083	1.097329525	1.138059	1.073309	1.09618	1.13733	1.12793	1.09588		
$P_{n-6}, q_{n-6,j}$	3000000	0.00900	0.01200	0.00904	0.01520	0.01197	0.01300	0.01521	0.01800	0.01304		
$x_2$ component coeff.	3383747	0.01015	0.013535	0.010191846	0.017144	0.013501	0.01466	0.01716	0.0203	0.0147		
Coeff of $I_{ij} = a_i + b_{ij} = RPC$	3383747	1.10768	1.086618	1.107521371	1.155204	1.08681	1.11084	1.15449	1.14823	1.11058		
Max coeff of $x_{n-6}$ in $f_{n-6}(x_{n-6})$					1.155204							
$j^*$					4							
$c_i$	48352748										3454010.21	51858791.13
$x_i \rightarrow$					3035000							
$P_{n-7}$	5000000	Stage	1	Computations								
$j \rightarrow$		1	2	3	4	5	6	7	8	9		
$r_j \rightarrow$		0.01300	0.00400	0.01297	0.02600	0.00401	0.01270	0.02587	0.02160	0.01266		
$\alpha_i \rightarrow$		1.1118	1.077376	1.111566278	1.167649	1.077615	1.1101	1.16675	1.15229	1.10976		
$P_{n-7}, q_{n-7,j}$	5000000	0.00250	0.01140	0.00251	0.00700	0.01137	0.02100	0.00701	0.01680	0.02106		
$x_2$ component coeff.	5776018	0.00289	0.013169	0.002899561	0.008086	0.013136	0.02426	0.00809	0.01941	0.02432		
Coeff of $I_{ij} = a_i + b_{ij} = RPC$	5776018	1.11469	1.090545	1.11446584	1.175735	1.090752	1.13436	1.17485	1.1717	1.13408		
Max coeff of $x_{n-7}$ in $f_{n-7}(x_{n-7})$					1.175735							
$j^*$					4							
$c_i$	51818359										5838244.6	\$57,697,036
$x_i \rightarrow$					5000000							

**Table 5: Summary of the Optimal Investment Strategies and corresponding Returns for Problem 3.3**

Year $i$	Bank Invested in $j^*$	Optimal Solution $I_{i,j^*}$	Optimal Decision $I_{i,j^*} = x_i$	Accumulation $s_i$
1	4	$I_{1,4} = x_1$	Invest $x_1 = \$5,000,000$ , in Bank 4	$s_1 = \$5,838,244.60$
2	4	$I_{2,4} = x_2$	Invest $x_2 = \$3,035,000$ , in Bank 4	$s_2 = \$3,454,010.21$
3	4	$I_{3,4} = x_3$	Invest $x_3 = \$7,046,132$ , in Bank 4	$s_3 = \$7,813,423.55$
4	4	$I_{4,4} = x_4$	Invest $x_4 = \$5,119,784$ , in Bank 4	$s_4 = \$5,581,306.85$
5	4	$I_{5,4} = x_5$	Invest $x_5 = \$2,133,114$ , in Bank 4	$s_5 = \$2,293,976.37$
6	4	$I_{6,4} = x_6$	Invest $x_6 = \$10,056,101$ , in Bank 4	$s_6 = \$10,623,231.30$
7	5	$I_{7,5} = x_7$	Invest $x_7 = \$8,172,965$ , in Bank 5	$s_7 = \$8,403,270.15$
8	8	$I_{8,8} = x_8$	Invest $x_8 = \$13,163,051$ , in Bank 8	$s_8 = \$13,689,572.68$
			<b>Optimal Total Dollar Accumulation</b> = $f_1(x_1) = \$57,697,036 = \sum_{i=1}^8 s_i$	

Note that  $x_i = \sum_{j=1}^9 I_{i,j}, i \in \{1, 2, \dots, 8\}$

**3.4 Problem 2: Electronic Solution of the following problem in Ukwu et al [ ]:**

Given the amount \$15,000 to be invested now and \$ 8,000, \$ 12,000, \$ 5,000 and \$13, 000 at the start of years 2,3 4 and 5, Table 2 furnishes the investment funds, pertinent nominal annual interest rates and end-of-year bonuses offered by a portfolio of three banks for a period of five successive years:

**Table 5: Pertinent Data for Problem 3.4**

Bank						Bank			
		1	2	3	4	1	2	3	4
Year	$P$	Nominal Annual Interest Rates				End-of-Year Bonuses			
1	15,000	0.012	0.0132	0.0136	0.07	0.02	0.013	0.015	0.022
2	8,000	0.012	0.0132	0.0136	0.07	0.018	0.011	0.030	0.020
3	12,000	0.012	0.0132	0.0136	0.07	0.03	0.023	0.027	0.083
4	5,000	0.012	0.0132	0.0136	0.07	0.032	0.025	0.0175	0.050
5	13,000	0.012	0.0132	0.0136	0.07	0.015	0.08	0.042	0.037

Devise the optimal investment strategies.

**Table 6: Template Outputs for Optimal Policies and Returns for Problem 3.4**

		$n =$	5	$k =$	4		
$f_i(x_i) = (\text{max coeff. of } x_i \text{ in stage } i) x_i$							
$+(1-\delta_{in}) \sum_{j=1}^n P_{ij} (\text{max coeff. of } x_j \text{ in } f_j(x_j)) = f_{i,j} x_i + (1-\delta_{in}) c_i$							
$P_5 =$	13000	<b>Stage</b>	<b>5</b>			$s_i$	$f_i(x_i) \downarrow$
$j \rightarrow$		1	2	3	4		
$r_j \rightarrow$		0.01200	0.13200	0.13600	0.07000		
$\alpha_j \rightarrow$		1.012	1.132	1.136	1.07		
$P_5, q_{5,j} =$	13000	0.01500	0.08000	0.04200	0.03700		
$x_6$ component coeff.	0	0	0	0	0		
$x_5$ , Coeff of $l_{5j} = a_j + q_{5j}$		1.027	1.212	1.178	1.107		
Max coeff of $x_5$ in $f_5(x_5)$			1.212				
$j^*$			2				
$c_5$						\$15,917.52	\$15,917.52
$x_5 \rightarrow$			13133.27				
$P_4 =$	5000	<b>Stage</b>	<b>4</b>	<b>Computations</b>			
$j \rightarrow$		1	2	3	4		
$r_j \rightarrow$		0.01200	0.13200	0.13600	0.07000		
$\alpha_j \rightarrow$		1.02414	1.281424	1.290496	1.1449		
$x_5$ comp. coeff. in stage (4) : $p_{5j}, q_{4,j}$	5000	0.03200	0.02500	0.01750	0.05000		
Max coeff. of $x_4$ in $f_4(x_4)$ ( $x_5$ comp. coeff. in stage 4)	6060	0.03878	0.0303	0.02121	0.0606		
Return function coeff (RFC): $\alpha_j$ + preced. contiguous row	6060	1.06293	1.311724	1.311706	1.2055		
Max coeff of $x_4$ in $f_4(x_4)$			1.311724				
$j^*$			2				
$c_4$	15756					\$6,830.84	\$22,748.36
$x_4 \rightarrow$				5330.66225			
$P_3 =$	12000	<b>Stage</b>	<b>3</b>	<b>Computations</b>			
$j \rightarrow$		1	2	3	4		
$r_j \rightarrow$		0.01200	0.13200	0.13600	0.07000		
$\alpha_j \rightarrow$		1.03643	1.450572	1.466003456	1.225043		
$x_4$ comp. coeff. in stage 3 : $P_{4j}, q_{3,j}$	12000	0.03000	0.02300	0.02700	0.08300		
Max coeff. of $x_3$ in $f_3(x_3)$ ( $x_4$ comp. coeff. in stage 3)	15740.688	0.03935	0.03017	0.035416548	0.108873		
Return function coeff (RFC): $\alpha_j$ + preced. contiguous row	15740.688	1.07579	1.480742	1.501420004	1.333916		
Max coeff of $x_3$ in $f_3(x_3)$				1.501420004			
$j^*$				3			
$c_3$	22314.62					\$17,953.78	\$40,702.14
$x_3 \rightarrow$				12246.75			
$P_2 =$	8000	<b>Stage</b>	<b>2</b>	<b>Computations</b>			
$j \rightarrow$		1	2	3	4		
$r_j \rightarrow$		0.01200	0.13200	0.13600	0.07000		
$\alpha_j \rightarrow$		1.04887	1.642047	1.665379926	1.310796		
$P_2, q_{2,j}$	8000	0.01800	0.01100	0.03000	0.02000		
$x_3$ component coeff.	12011.36	0.02703	0.016516	0.0450426	0.030028		
Coeff of $l_{2j} = a_j + q_{2j} = \text{RFC}$	12011.36	1.0759	1.658563	1.710422526	1.340824		
Max coeff of $x_2$ in $f_2(x_2)$				1.710422526			
$j^*$				3			
$c_2$	40331.66					\$13,697.75	\$54,399.89
$x_2 \rightarrow$				8225			
$P_1$	15000	<b>Stage</b>	<b>1</b>	<b>Computations</b>			
$j \rightarrow$		1	2	3	4		
$r_j \rightarrow$		0.01200	0.13200	0.13600	0.07000		
$\alpha_j \rightarrow$		1.06146	1.858798	1.891871596	1.402552		
$P_1, q_{1,j}$	15000	0.02000	0.01300	0.01500	0.02200		
$x_2$ component coeff.	25656.338	0.03421	0.022235	0.025656338	0.037629		
Coeff of $l_{1j} = a_j + q_{1j} = \text{RFC}$	25656.338	1.09567	1.881033	1.917527934	1.440181		
Max coeff of $x_1$ in $f_1(x_1)$				1.917527934			
$j^*$				3			
$c_1$	54015.04					\$28,378.07	\$82,777.96
$x_1 \rightarrow$				15000			

The outputs automatically generated, following the keying in of pertinent data are consistent with the manual solutions in [7].

#### 4. CONCLUSION

This article designed and automated prototypical solution templates for optimal policy prescriptions for a certain dynamic class of investment problems, with an arbitrary set of pertinent data, complete with an algorithmic exposition on the interface and solution process. The optimality results were realized through the interrogation of the layout of the set of feasible data and spreadsheet locations for the various outputs at each stage, a robust investigation of the solution templates in Ukwu [8] for the optimal investment strategies and rewards for a certain dynamic class of probabilistic investment problems with specified market conditions and the transformation of the current backward dynamic recursions to special functional forms suitable for automation on Excel platform. In the sequel, the article deployed the templates to secure the optimal fund allocations with respect to two illustrative problems of different investment portfolio cardinalities, demonstrating the efficiency, power and utility of the templates and their suitability for sensitivity analyses of the problem parameters. Financial practitioners, consultants and decision makers can leverage on these templates to enhance their confidence and portfolio of competencies, as well as secure large market share and optimal returns on their investments.

#### References

- [1] Gupta, P.K. & Hira, D.S. (2012). Problems in operations research (Principles and Solutions). S. Chand & Company Ltd. New Delhi. 2012; 844-854.
- [2] Taha, H.A. (2006). Operations Research: An Introduction. Seventh Edition. Prentice-Hall of India, New Delhi. 418-421.
- [3] Ukwu C. (2016). Alternative Layout and Automation of Optimality Results for Machine Replacement Problems Based On Stationary Data and Age Transition Perspectives. *Advances in Research*. 7(1): 1-13.
- [4] Ukwu C. (2016). Design and Full Automation of Excel Solution Templates for a Time-perspective Class of Machine Replacement Problems with Pertinent Dynamic Data. *Archives of Current Research International*. 4(1): 1-15.



- [5] Ukwu C. (2016). Starting Age Zero-Based Excel Automation of Optimal Policy Prescriptions and Returns for Machine Replacement Problems with Stationary Data and Age Transition Perspectives. *Journal of Scientific Research & Reports* 10 (7): 1-11.
- [6] Ukwu C. (2016). An Algorithm for Global Optimal Strategies and Returns in One Fell Swoop, for a Class of Stationary Equipment Replacement Problems with Age Transition Perspectives, Based on Nonzero Starting Ages. *Advances in Research*. 7(4):1-20.
- [7] Ukwu C., Manjel, D. & Kutchin S. (2017). Optimal Fund Allocation from Certain Investment Portfolio Using Backward Dynamic Programming Recursions. *International Research Journal of Natural and Applied Sciences* (to appear) July 2017 Issue.
- [8] Ukwu, C. (2016). Optimal investment strategy for a certain class of probabilistic investment problems. *International Research Journal of Natural and Applied Sciences*. 3(2): 92-105.