ACTUARIAL RESEARCH CLEARING HOUSE
1993 VOL. 3

Speeding Up Universal Life Calculations in BASIC

by Dennis Radliff

Oftentimes Universal Life calculations, since they are normally monthly, require a lot of "grinding out" in a BASIC program. Some programmers even do an annual approximation when calculating guideline premiums or other lengthy calculations. A couple of things can be done to speed these up. First of all, you can compress the monthly calculations into an annual calculation that does not require 12 passes through a loop; and second, you can calculate a guideline premium by one pass through a loop from maturity back to issue. Since the formulas are not approximations, the results are the same as if you had used a monthly calculation or had used a trial-and-error iterative process.

First, I will start with a typical UL formula, and show how the calculation of yearend fund value can be done in one pass instead of 12. The UL formula to work with is as follows:

(1) $CVEM=\{ICV+(1-PC)P-OE-(DB/GIF-[ICV+(1-PC)P-OE])COIR\}(1+i)^{1/12}$

where:

ICV = initial policy fund value, possibly rollover
PC = percent of premium expense
P = gross payment
OE = other monthly expense (deducted beginning of month)
DB = specified amount
GIF = guaranteed interest factor = $1.045^{1/12}$ if guaranteed rate is 4.5%
COIR = monthly COI rate per $1 (NOT per $1,000)
i = current interest rate as a decimal
CVEM = fund value at the end of month

The above formula is basically taking an initial value, adding in a net payment, subtracting off any beginning-of-the-month expenses, subtracting a mortality charge based on the net amount at risk, and going forward with interest one month. The net amount at risk is obtained by adjusting the specified amount by the guaranteed interest factor, and subtracting the beginning fund value after the net payment has been added and expenses subtracted. If there is a waiver of monthly deduction, this would take place after the mortality deduction had been determined. This cycle is repeated 12 times to get a fund value at the end of the year.

Equation 1 can be rearranged as follows:

(2) $CVEM=[ICV+(1-PC)P-OE](1+COIR)(1+i)^{1/12}-(DB/GIF*COIR)(1+i)^{1/12}$

Now let $(1+i)^{1/12}$ = 1+mi (monthly interest) and let f =

(1+COIR)(1+mi). Thus f is a monthly interest and mortality factor.

If premiums are monthly, then by yearend, we have:

(3) $CVEY = (ICV)f^{12}$

$+[(1-PC)P - OE][f^{12}+f^{11}+...+f]$

$-(DB/GIF*COIR)(1+mi)(f^{11}+f^{10}+...+f+1)$, where CVEY is the fund at the end of the year.

Using the fact that the mortality and interest factors are geometric progressions, we define

$se = f^{12}+f^{11}+...+f = [(f^{12}-1)/(f-1)]f$ and

$sc = [(f^{12}-1)/(f-1)][1+mi]$.

I will also define an additional intermediate value, since it is used in the program illustrations below:

$xsc = (f^{12}-1)/(f-1)$

We must do a separate calculation for premiums, since they may vary by mode, and are not just monthly like the mortality and expense charges. In this formula, md=12 if annual mode, 6 if semiannual, 3 if quarterly, and 1 if monthly. Thus we define:

$fp = [(1+COIR)^{md} (1+i)^{(md/12)}]$

This represents a modal interest and mortality factor. We further define:

$sp = [(f^{12}-1)/(fp-1)]fp$

You can think of se, sc, and sp as a way of accumulating monthly payments and charges to the end of the year: se for expenses, sc for COI charges, and sp for modal payments.

Now we can express equation 3 as:

(4) $CVEY=[(ICV)f^{12}]+[(1-PC)P*sp]-[OE*se]-[DB/GIF*COIR*sc]$

I will not go through the whole analysis, but if there is a loan on the policy which is earning less than <u>i</u>, we can define <u>di</u> as the difference between the monthly rate earned on unloaned funds and the rate earned on loaned funds.

So if there is a loan on the policy, formula 4 can be expressed as:

```
F  = COI.FACTOR * MTH.INT.FACTOR

'fp is MODAL mort & interest factor
FP  = (COI.FACTOR ^ (MODE.FACTOR * 12)) * MODAL.INT.FACTOR

'xsc is additional factor for calculations below
XSC = ((F ^ 12) - 1) / (F - 1)

'sc is yearly accumulation factor for monthly COI charges
SC  = XSC * MTH.INT.FACTOR

'se is yearly accumulation factor for monthly expenses
SE  = XSC * F

'sp is yearly accumulation factor for MODAL premiums
SP  = (((F ^ 12) - 1) / (FP - 1)) * FP

'Calculate accumulated values for year.
'If policy is not in corridor, the adjustment for waiver is already in
'   the COI, but in the corridor, you have to accumulate the waiver rate
'   times the monthly charges separately.

'accumulated value of the monthly COI charges to the end of the year.
IF NOT IN.CORRIDOR THEN
    ACCUM.VALUE.COI = COI * SC
ELSE
    ACCUM.VALUE.COI = WMD.RATE * MTHLY.CHG * SC
END IF

'accumulated value of the monthly expense charges to the end of the year.
ACCUM.VALUE.EXP        = MTHLY.CHG * SE

'accumulated value of the loan spread to the end of the year. DIFF.INT is
'   MTH.INT.FACTOR - GIF if the gtd rate is credited to loaned funds.
ACCUM.VALUE.LOAN.ADJ  = LOAN * DIFF.INT * XSC

'accumulated value of the modal premiums paid to the end of the year.
ACCUM.VALUE.PREM       = MODAL.PREMIUM * (1 - PREM.LOAD) * SP

'Calculate fund value for year. LAST.VALUE is fund at end of last year.
FUND.VALUE = LAST.VALUE * (F ^ 12) + ACCUM.VALUE.PREM_
           - ACCUM.VALUE.COI - ACCUM.VALUE.EXP -ACCUM.VALUE.LOAN.ADJ

RETURN   'end of YEARLY.CALC
```

The above program is simply a subroutine for doing the main part of the calculation. In the actual program, a check is done each year to see if the policy is in the corridor. For the year the policy goes into or out of the corridor, the program branches to a subroutine that recalculates the fund on a monthly basis for those 12 months only.

following years. the above factor will work.

Another difference that must be accounted for because of the corridor is waiver of monthly deduction. If a policy is not in the corridor, the COI rate is adjusted for waiver and the COI is accumulated with this adjustment. However, when a policy hits the corridor, the cost of insurance to be accumulated with the factor <u>sc</u> is zero: so the additional waiver of monthly deduction charges, i.e., the monthly expenses times the waiver rate, must be accumulated separately when a policy is in the corridor.

All this may be difficult to follow, but it is easier to see if you look at an actual BASIC subroutine. For this purpose, below is a printout of a routine I use to calculate UL fund values on a yearly basis. The syntax of the program is according to Microsoft QuickBASIC. As I said, the fund values obtained this way are identical to those you get if you loop through a monthly routine 12 times, but it is much faster.

```
' ---------------------------------------
' Calculate Fund Values on a yearly basis
' ---------------------------------------
YEARLY.CALC:
     'coi.rate is monthly cost of insurance per $1
     'first adjust the monthly COI rate for waiver of monthly deduction per $1

     COI.RATE = COI.RATE * (1 + WMD.RATE)

     'Set up COI to be used below.  GIF = (1+gtd int rate)^(1/12).
     'Rider cost is sum of all deductions for family rider, ADB, etc.
     'Face amt is specified amount for both death benefit options.
     'Mthly.chg is total monthly expense charges, including rider cost.

     MTHLY.CHG = RIDER.COST + MTHLY.EXP
     COI = (FACE.AMT / GIF) * COI.RATE + WMD.RATE * MTHLY.CHG

     'if opt 2 or in corridor, then set up different COI factor.
     'corridor factors start at 1.5 through age 40, then reduce.
     IF IN.CORRIDOR THEN                                   'in corridor
         COI.FACTOR = 1 - (CORRIDOR.FACTOR * COI.RATE)
     ELSEIF DB.OPTION = 2 THEN                            'increasing death ben
         COI.FACTOR = 1 + COI.RATE * (1 - (1 / GIF))
     ELSE                                                  ' opt 1, not in corridor
         COI.FACTOR = 1 + COI.RATE
     END IF

     'Set up accumulation factors.
     'Mth.int.factor = (1+i)^(1/12) & modal.int.factor =(1+i)^(mode.factor),
     '   where mode.factor is .25 for quarterly, .5 for semiannual, etc.

     'f is monthly mort & interest factor
```

(4.1) $CVEY=[(ICV)*f^{12}]+[(1-PC)*P*sp]$
$-[OE*se]-[(DB/GIF*COIR)*sc]$
$-[LV*di*xsc]$,
where LV is the loan value.

If you examine the terms of this expression, it represents initial fund accumulated with interest and mortality to yearend, plus accumulated net premium, less accumulated monthly expenses, less accumulated monthly COI charges, less accumulated difference in loan interest due to fund being impaired.

If a waiver of monthly deduction rider is added, then COIR simply becomes the monthly COI rate per $1 multiplied by the factor (1+WMD.RATE), where WMD.RATE is the cost of waiver per $1 of monthly deduction. Thus, for equation 2 and beyond, COIR becomes:

COIR=COIR'*(1+WMD.RATE), where COIR' is the monthly COI rate per $1 for the basic policy without waiver.

If there are other riders on the policy, such as a family rider or ADB, these costs are simply added to the "other expenses" (OE in the equations above).

If you are computing values for an increasing UL policy (death benefit option 2), then the death benefit in the original equation 1 becomes:

$DB=SA+ICV+(1-PC)P-OE$

This is because the death benefit in this case becomes the specified amount plus the beginning fund value. If you work this out algebraically, the only change that needs to be made is by adjusting the cost of insurance rate in the definition of the factor f. For option 2:

$f=\{1+COIR[1-(1/GIF)]\}(1+mi)$

The rest of the calculations are the same.

If a policy goes into the DEFRA corridor and the death benefit goes up, again you just have to adjust the definition of the factor f:

$f=[1-(CORRIDOR.FACTOR)(COIR)](1+mi)$,

where CORRIDOR.FACTOR is the DEFRA factor beginning at 1.50 from ages 0 through 40, then decreasing.

For the year in which the policy actually enters the corridor, you must either do the actual monthly calculation for just those 12 months, or else do an approximation. However, in the

Now that we have developed a routine for calculating one year of values, we can expand this to maturity of the policy and directly calculate a guideline premium or target premium . Actually, a target premium can be calculated in this manner for any number of years; so it can be used in a UL proposal system to calculate the premium required to reach a certain fund at a certain point. Again, since we are using exact formulas, this procedure will give you the correct premium even if you calculate the fund values on a monthly basis.

Expanding on the notion used above, I will use a subscript to indicate the year, since the mortality charges and interest rate may change from year to year. The fund at the end of year 2 can be calculated as:

$$(5)\quad CVEY_2 = (CVEY_1)f_2^{12} + [(1-PC_2)P*sp_2]$$
$$- [OE_2*se2] - [(DB_2/GIF*COIR_2)*sc_2]$$

Notice that P in the expression above has no subscript. This is because we are going to be solving for a level premium, and P is the unknown quantity.

Now, substituting the CVEY expression in equation 4 for $CVEY_1$ in equation 5, adding subscripts of 1 where necessary, and simplifying the result we get:

$$(6)\quad CVEY_2 = (ICV)*f_1^{12}*f_2^{12}$$
$$+ P(1-PC_1)*sp_1*f_2^{12}$$
$$- OE_1*se_1*f_2^{12}$$
$$- (DB_1/GIF*COIR_1)*sc_1*f_2^{12}$$
$$+ P(1-PC_2)*sp_2$$
$$- OE_2*se_2$$
$$- (DB_2/GIF*COIR_2)*sc_2$$

Now considering a more general case where we want the fund value at the end of year 5 and we want to pay a level premium for 3 years. To solve for this premium, we first set up the following equation, containing expressions for the values for each of the 5 years:

$$(7)\quad CVEY_5 = [ICV*f_1^{12}*f_2^{12}*\ldots*f_5^{12}]$$

$$+ [P(1-PC_1)*sp_1*f_2^{12}*\ldots*f_5^{12}]$$

$$- [(DB_1/GIF*COIR_1)*sc_1*f_2^{12}*\ldots*f_5^{12}]$$

$$- [OE_1*se_1*f_2^{12}*\ldots*f_5^{12}]$$

$$+ [P(1-PC_2)*sp_2*f_3^{12}*f_4^{12}*f_5^{12}]$$

$$- [(DB_2/GIF*COIR_2)*sc_2*f_3^{12}*f_4^{12}*f_5^{12}]$$

$$- [OE_2*se_2*f_3^{12}*f_4^{12}*f_5^{12}]$$

$$+ [P(1-PC_3)*sp_3*f_4^{12}*f_5^{12}]$$

$$- [(DB_3/GIF*COIR3)*sc_3*f_4^{12}*f_5^{12}]$$

$$- [OE_3*se_3*f_4^{12}*f_5^{12}]$$

$$- [(DB_4/GIF*COIR_4)*sc_4*f_5^{12}]$$

$$- [OE_4*se_4*f_5^{12}]$$

$$- [(DB_5/GIF*COIR_5)*sc_5]$$

$$- [(OE_5*se_5]$$

The way the above formulas can be described is as follows:
The initial fund at issue is accumulated to the end of year 5 by multiplying successively by each of the "f" factors for years 1 through 5. The net premium is accumulated to the end of year 1 using the "sp" factor for year 1. This value is then accumulated from the end of year 1 to the end of year 5 using the "f" factors for years 2 through 5. The COI charges and other expenses are accumulated in like manner and subtracted. This process is continued for years 2 through 5, accumulating values from the end of each year to the end of year 5. Notice that there is no premium expression for the fourth and fifth year, because we are calculating a 3-year level premium.

For this case, we define

$$fcv_t = f_{t+1}^{12} * f_{t+2}^{12} * \ldots * f_5^{12}$$

$$fcv_5 = 1$$

fcv can be considered an accumulation factor for the fund from the end of the year t to the end of the year 5.

Now we can rewrite equation 7 as:

$$(8)\quad CVEY_5 = [ICV*fcv_0]$$
$$+ [P(1-PC_1)*sp_1*fcv_1]$$
$$+ [P(1-PC_2)*sp_2*fcv_2]$$
$$+\ldots+ [P(1-PC_5)*sp_5*fcv_5]$$
$$- [(DB_1/GIF*COIR_1)*sc_1*fcv_1]$$
$$-\ldots- [(DB_5/GIF*COIR_5)*sc_5*fcv_5]$$
$$- [OE_1*se_1*fcv_1]$$
$$- [OE_5*se_5*fcv_5]$$

Note: P is 0 for years 4 and 5 above, so only years 1-3 are accumulated above in the expressions involving P.

Now define:

$$Totalfp = \sum_{t=1}^{5}[(1-PC_t)*sp_t*fcv_t]$$

$$TotalDeduc = \sum_{t=1}^{5}\{[(DB_t/GIF*COIR_t*sc_t)+(OE_t*se_t)]*fcv_t\}$$

Totalfp represents a factor for accumulating premium over 5 years, and TotalDeduc represents total accumulated deductions over 5 years.

Thus we can write:

$$(9)\quad CVEY_5 = [P*\ Totalfp] + [ICV * fcv_0] - TotalDeduc$$

Solving for P, we get:

(10) $P = [CVEY_{s} - (ICV*fcv_{o}) + TotalDeduc]/Totalfp$

This is our target premium.

Below is a routine in QuickBASIC for computing a target premium by using the calculation above, starting at the target age and going back to the issue age year by year.

```
' ---------------------------------
' Calculate Target Premium Directly
' ---------------------------------

TARGET.PREMIUM:

    ' Start thru calculation from last year forward.
    ' If calculating to maturity, target age is (maturity age - 1).
    ' This is because you are going to the END of the previous age.

    FOR AGE% = TARGET.AGE% TO ISSUE.AGE% STEP -1

        'read the COI rate per $1 for this age from the array.
        COI.RATE = COIR(AGE%)
        'adjust for waiver of monthly deduction.
        COI.RATE = COI.RATE * (1 + WMD.RATE)
        'set up the total monthly expense charges.
        MTHLY.CHG = RIDER.COST + MTHLY.EXP
        'set up cost of insurance used below.
        COI = (FACE.AMT / GIF) * COI.RATE + WMD.RATE * MTHLY.CHG

        'set up different COI factor for option 2
        IF DB.OPTION = 1 THEN
            COI.FACTOR = 1 + COI.RATE                   'option 1 factor
        ELSE
            COI.FACTOR = 1 + COI.RATE * (1 - (1 / GIF)) 'option 2 factor
        END IF

        'set up monthly interest factor
        MTH.INT.FACTOR = (1 + INT.RATE(AGE%))^(1 / 12)
        'set up modal interest factor. Mode.factor=.25 for quarterly, etc.
        MODAL.INT.FACTOR    = (1 + INT.RATE(AGE%)) ^ MODE.FACTOR

        'set up factors for yearly calculation
        F    = COI.FACTOR * MTH.INT.FACTOR
```

```
    FP  = (COI.FACTOR ^ (MODE.FACTOR * 12)) * MODAL.INT.FACTOR
    XSC = ((F ^ 12) - 1) / (F - 1)
    SC  = XSC * MTH.INT.FACTOR
    SE  = XSC * F
    SP  = ((F ^ 12) - 1) / (FP - 1) * FP

    'calculate accumulated value of monthly rider costs & expenses
    ACCUM.VALUE.EXP = MTHLY.CHG * SE

    'calculate accumulated value of COI
    ACCUM.VALUE.COI = COI * SC

    'Calculate factor to accumulate values to target age. LAST.YAF
    '    is yearly accumulation factor for mortality & interest
    '    from last year.  Now we accumulate FCV for another year.
    IF AGE% = TARGET.AGE% THEN
        FCV = 1
    ELSE
        FCV = LAST.FCV * LAST.YAF
    END IF

    'calculate premium factor to accumulate values to target age.
    PREM.FACTOR = (1 - PREM.LOAD) * SP * FCV

    'zero out premium factor for this year if beyond payment period.
    IF AGE% > LAST.PAY.AGE% THEN PREM.FACTOR = 0

    'accumulate premium factors.
    TOTAL.PREM.FACTOR = TOTAL.PREM.FACTOR + PREM.FACTOR

    'calculate accumulated deductions for another year.
    ACCUM.DEDUCT = (ACCUM.VALUE.EXP + ACCUM.VALUE.COI) * FCV

    'accumulate total deductions so far.
    TOTAL.DEDUCTION = TOTAL.DEDUCTION + ACCUM.DEDUCT

    'Store cumulative cash value factor (FCV) & yearly accumlation
    '    factor to use next year.
    LAST.FCV = FCV
    LAST.YAF = F ^ 12

NEXT AGE%

'accumulate initial fund to the target age.
ACCUM.ICV = ICV * LAST.YAF * FCV

'calculate target premium.
NUMERATOR = TARGET.FUND + TOTAL.DEDUCTION - ACCUM.ICV
TARGET.PREM  = NUMERATOR / TOTAL.PREM.FACTOR

RETURN           'end of TARGET.PREMIUM
```

The above routines can be used to calculate target premiums, guideline premiums, guaranteed maturity premiums, and 7-pay premiums. Again, since no approximations are used, you get the same results as you get by using a month-by-month calculation, but it goes much faster. For example, with these routines, it is feasible to use a BASIC program for calculating reserves according to the UL Model Regulation. You can use the target premium routine to calculate the guaranteed maturity premium directly and then use the yearly calculation routine for calculating guaranteed maturity funds and for doing the projection of death benefits beyond the valuation date. I have written such a program for our company's use.